

Novakovic, Andrew M.; Babb, Emerson M.; Martella, David R.; Pratt, J.E.

Working Paper

A Computer Program Documentation of the Dairy Market Policy Simulator

Staff Paper, No. SP 79-04

Provided in Cooperation with:

Charles H. Dyson School of Applied Economics and Management, Cornell University

Suggested Citation: Novakovic, Andrew M.; Babb, Emerson M.; Martella, David R.; Pratt, J.E. (1979) : A Computer Program Documentation of the Dairy Market Policy Simulator, Staff Paper, No. SP 79-04, Cornell University, Charles H. Dyson School of Applied Economics and Management, Ithaca, NY, <https://doi.org/10.22004/ag.econ.183685>

This Version is available at:

<https://hdl.handle.net/10419/276555>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

A COMPUTER PROGRAM DOCUMENTATION OF THE DAIRY

MARKET POLICY SIMULATOR (MODEL A)

by

A. M. Novakovic, E. M. Babb,
D. R. Martella and J. E. Pratt

April 1979

No. 79-4

PREFACE

This paper provides a listing and brief documentation of the Dairy Market Policy Simulator, or DAMPS. DAMPS was constructed as part of Novakovic's Ph.D. thesis at Purdue University. During some part of the course of the research and model development, all of the authors were at Purdue. Dr. Andrew Novakovic is now an Assistant Professor in the Department of Agricultural Economics at Cornell University. Dr. Emerson Babb is a Professor in the Department of Agricultural Economics at Purdue University. David Martella is on the staff of the Department of Agricultural Economics and Rural Sociology of the University of Arkansas. James Pratt is a Ph.D. student in the Department of Agricultural Economics at Michigan State University.

The authors also benefited from the earlier modeling work of Dr. David Banker of the Dairy Group of the Economics, Statistics, and Cooperatives Service in the U.S. Department of Agriculture and Dr. Oscar Goldman, a graduate of Pennsylvania State University now living in Argentina.

The information contained in this paper is intended as a reference for those already familiar with DAMPS. A detailed description of the economic model embodied in DAMPS or the underlying mathematical relations is not provided. It is assumed that the reader has a basic understanding of the U.S. dairy economy and the methods used to construct DAMPS.

Requests for further information regarding DAMPS, should be directed to:

Andrew Novakovic
Department of Agricultural Economics
312 Warren Hall
Cornell University
Ithaca, New York 14853

TABLE OF CONTENTS

	<u>Page</u>
INTRODUCTION.....	1
DAMPSIN.....	2
DAMPSLV.....	33
DAMPCLC.....	76
DRWL.....	99
DRW2.....	137
LIST OF MAJOR VARIABLES.....	154
BIBLIOGRAPHY.....	208

A COMPUTER PROGRAM DOCUMENTATION OF THE DAIRY

MARKET POLICY SIMULATOR (MODEL A)

INTRODUCTION

The Dairy Market Policy Simulator or DAMPS was developed as part of a thesis research project at Purdue University (7). DAMPS evolved from a model known as the Federal Milk Marketing Order System Policy Simulator, or FMMOPS, developed by Banker (4). Details of the economic model and program documentation for FMMOPS are provided elsewhere (1, 2, 3, 5, 6). These references serve as background to DAMPS.

DAMPS is a transshipment model of the U.S. Dairy Sector. The dairy sector is split geographically into federally regulated areas, state regulated areas, unregulated Grade A milk regions, and Grade B or manufacturing milk regions.

Model components include:

- supplies of Grade A milk
- supplies of Grade B milk
- processing activities
- demands for fluid (Class I) products, soft (Class II) manufactured products, cheese, butter, nonfat dry milk, and miscellaneous hard (Class III) manufactured products
- imports of cheese, butter, nonfat dry milk and miscellaneous Class III products
- commercial stocks of cheese, butter, and nonfat dry milk
- government stocks of cheese, butter, and nonfat dry milk

DAMPS is a quarterly model, capable of simulating from one to five years of dairy sector activity. Dynamic elements of the dairy sector are represented in DAMPS by the carryover of dairy stocks between quarters and by a lagged response of production and consumption to prices.

Simulation with and the operation of DAMPS is controlled by two sources of data--base data and input form data. Base data are data for the base year (1976) from which projections of production, consumption, and so forth are extrapolated. The base data includes prices and quantities for milk and milk products, processing capacities, marketing costs, hauling distances, exogenous factors affecting production, consumption, and costs, and other restrictions or model options. Input form data permit the user to select various pricing, merger, exogenous factors, length of run, and report writer options. For a listing and description of the data see Novakovic (7, Appendix C).

Projected Grade B prices are the basic model input. The Class III price in regulated areas is equal to the Grade B price. Class I and II prices in regulated areas are based on Class I and II differentials added to the Class III price. Differentials default to values in the base year, 1976, but can be set at any level. Retail prices are based on farm level prices and marketing costs or margins. Other model data include exogenously specified import levels, desired stock levels, the levels of exogenous factors affecting supply and demand, and restrictions and pricing mechanisms used in regulated markets.

Given a matrix of prices and exogenous factors, quarterly production, consumption, beginning stocks, imports, and desired ending stocks, can be computed in each area or region. DAMPS determines the spatial allocation that minimizes marketing costs; a capacitated network algorithm solves the transshipment problem. Actually, the transshipment model solves in three stages. In stage one, the optimal allocation of raw milk in fluid milk markets is computed. In stage two, primary manufactured products (all manufactured products, excluding butter and nonfat dry milk) markets are modeled. For further details on the mathematical model the reader is referred to Novakovic (7, Chapter III). Use of DAMPS, input forms, and available output are described elsewhere by Novakovic, *et al.* (8).

The purpose of the remainder of this paper is to list the computer programs that form DAMPS and briefly discuss the purpose of the programs. A detailed documentation of the computer model is not provided. Each program contains some internal documentation. All programs are written in FORTRAN IV for the Minnesota FORTRAN compiler available with the CDC 6500 at Purdue University. A list of the major variables used in the program is also provided, along with their dimension and function.

DAMPS is made up of five computer programs named DAMPSIN, DAMPSLV, DAMPCLC, DRW1, and DRW2. Figure 1 illustrates the flow of these programs in DAMPS. Briefly, DAMPSIN reads in model input and sets up base and input data needed for the solver and the report writers. DAMPSIN will list the base data, if requested. DAMPSLV sets up the model networks and solves the networks for the number of quarters requested. Results from stage one are passed to DAMPCLC. Some results from stage one and all results from stages two and three are passed to DRW2. DAMPSLV lists the solution time, value of the objective function, and other information which may be of interest to the operator. A listing of costs and restrictions on arcs and the raw solution by arcs is optional. DAMPCLC performs calculations needed for the report writers, in particular DRW1. DRW1 writes reports for stages two and three of DAMPS and for aggregated results of all three stages.

DAMPSIN

DAMPSIN, listed on the following pages, includes one subroutine--OSET and five functions--GCLIP, TRANSE, TRANSM, TRANSD, and TRANSP. The main program reads input and base data, prints base data (if requested), and writes data as it is needed to DAMPSLV, DAMPCLC, and DRW2. OSET sets up an array used in DAMPCLC for a report. The functions compute various transportation costs.

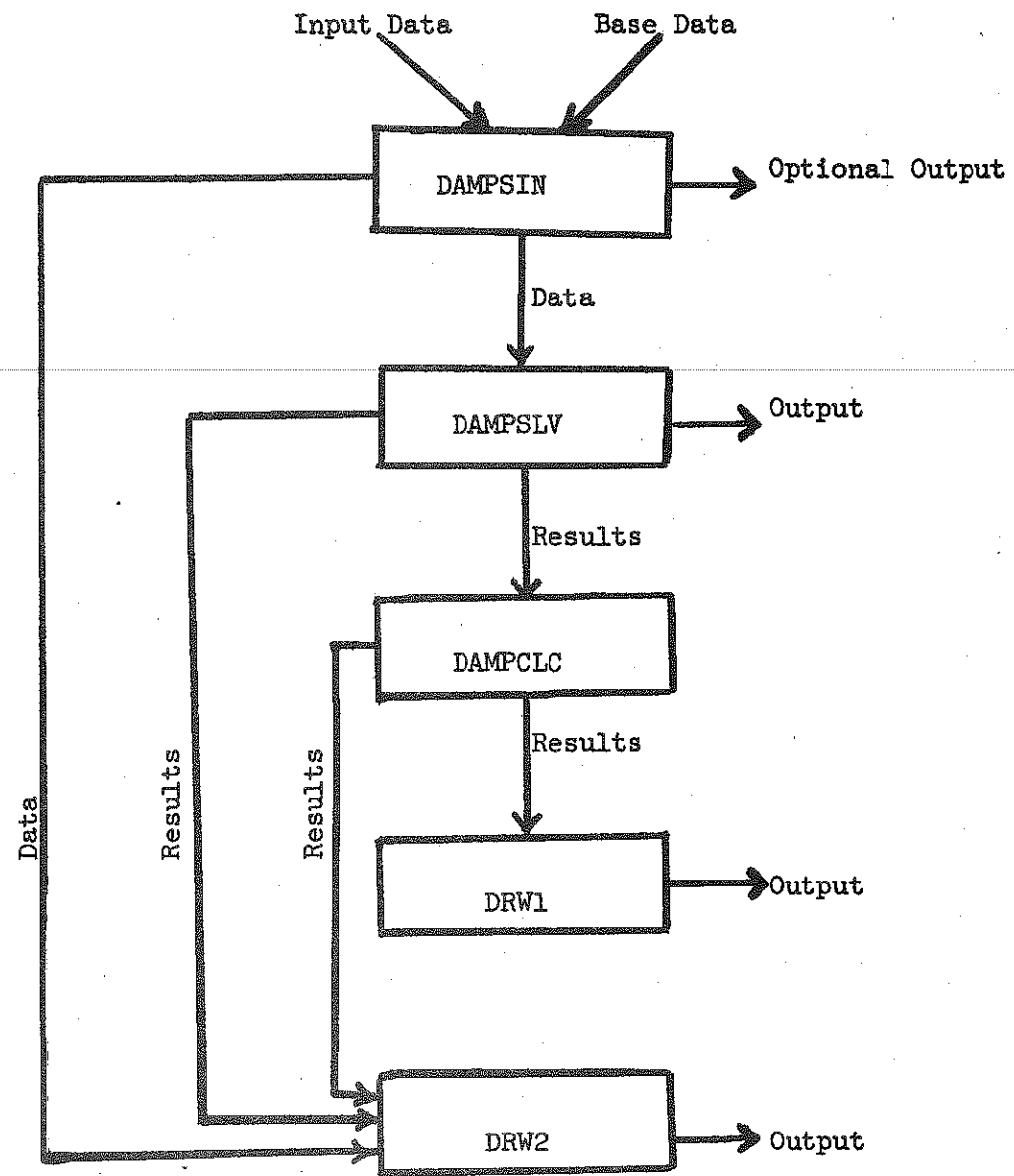


Figure 1. The Overall Flow Between Computer Programs Comprising DAMPS.

PROGRAM DAMPSIN (INPUT=130,OUTPUT=130,TAPES=OUTPUT,TAPES=INPUT,TAP
1E8=130,TAPE9=260,TAPE10=130,TAPE12=130,TAPE41=130)

.... DAMPSIN IS THE FIRST PROGRAM OF DAMPS
DAMPSIN READS IN AND PREPARES DATA FOR DAMPSLU, DAMPCLC,
AND THE REPORT WRITERS - DRW1 AND DRW2. DAMPSIN WAS
WRITTEN BY D.R. MARTELLA, J.E. PRATT, AND A.M. NOVAKOUC.

DIMENSION WORK1(14992)
DIMENSION WORK2(8025)
DIMENSION WORK4(24), WORK5(890), WORK6(45), WORK8(11)
DIMENSION WORK9(407), WORK10(852), WORK11(18)
DIMENSION WORK12(767), WORK13(10)

...WORK1 = TOTAL SIZE OF BLANK COMMON UP TO AND INCLUDING SPACE2
WORK2 = TOTAL SIZE OF COMMON IN BLK1
WORK4 = TOTAL SIZE OF COMMON IN BLK2
WORK5 = TOTAL SIZE OF COMMON IN BLK5
WORK6 = TOTAL SIZE OF COMMON IN BLK7 EXCLUDING OP4
WORK8 = TOTAL SIZE OF COMMON IN BLK9
WORK9 = TOTAL SIZE OF COMMON IN BLK10
WORK10 = TOTAL SIZE OF COMMON IN BLK11 UP TO AND INCLUDING EFRR
WORK11 = TOTAL SIZE OF COMMON IN BLK12
WORK12 = TOTAL SIZE OF COMMON IN BLK14
WORK13 = SIZE OF LAST LINE OF COMMON IN BLK11

...CURRENT SIZE OF UNLABELED COMMON IN DAMPSIN IS 22093

COMMON SPACE1(9600), ABLEND(59), ACEXP(2), ACHGC1(45), ACHGC2(45), ACHG
1C3(45), ACHGPR(45), ACHIAE(45), ACL1P(59), ACL1UT(59), ACL2D(59, 2), ACL3
2D(59, 2), AFRPS(59), AGRC1(45), AIAC1S(59, 2), AMBLND(2), AMC1P(2), AMC1U(
32), AMC2P(2), AMC3P(2), AMRTL(2), AMUCP(2), ANFP(45), ADAG1T(45), APCL1(5
43, 2), APPREC(59, 2), ARDD(45), ARPR(59), ASCL2D(2), ASCL3Q(2), ASFRPS(2)
5, ASIAC1(2), ASMFGU(2), ASNFP(2), ASDAC1(2), ASPCL1(2), ASPRR(2), ASRODC
6(2), ASTRNS(2), ATCCWT(59), ATMC(53), ATRNS(45), ATRMC(45), AUMCP(53),
7AUTCP(59), AVP(2), ANFRPS(2), BDEXP(5), BC3P(5), BLDIF(59), BMBLEN(5), BM
8CL1P(5), BMCL1U(5), BMRTL(5), BNCL2P(5), BNCL3P(5), BP(20), BPS(20), BSM
9FGU(5), BSMUCP(5), BSNFP(5), BSDAC1(5), BSTRNS(5), BSUFPR(5), BSUP(5), BS
*URDD(5), BNFRPS(5), CHGCL1(59), CHGCL0(45), CHGCL2(45), CHGCL3(45), CHGI
*AC(45), CHGPR(45), CL1P(59, 5), CL1Q(59, 4), CL1UT(59), FRPS(59), IACL1(5
*3, 4), ICR(10), IDICST(5), IMRG(59), MCL1Q(5), MCL2Q(5), MCL3Q(5), MIACL1(
*4), MR(5), MTPSQ(5), NCL2D(59, 4), NFP(45), NYID(5), OMC1P(75), PCL2(59), Q
*CL1P(59, 20), QCL2P(20), QCL3P(20), QR(20), RODC(45), SAD(14, 4), SERCST(5
*), SRMPC(5), TCCHT(59), TMC(53), TPSQ(59, 4), UMCP(53), UTCP(59), SPACE2(
*59), ARMD(45, 45), ARMS(16, 45), AMMFG(61, 28), AFMM(59, 45), ORB(5), ICRB
*(14), ICRS(10)

COMMON /BLK1/ SPMI(61, 45), DMMFG(27, 10), LMMFG(27, 10), MMNC(27, 10), PD
1MI(45, 45), MPDRM(59), MPDFM(59), IPPM(45), MARC(15), SFLB(16, 4), DISTB(7
25), DISTP(59), NOR(75), QNAME(75), QSPR(16), HCHC1(16), MMDM(19), CHCPT(5
3), CHGBT(5), AR, BR, AP, DP1, IBTEF, IPTEF, NSX1, NSPX, A(59, 5), DE(59), CHCNC
42(59), CHCNC1(59), IC1EF, BC2P(5), CL2Q(59, 4), PSE2, ICEEF, BIAC1(59, 4), D
51STD, DCHG1, SSPMI(14, 8, 8)

COMMON /BLK2/ RBA(5), NSP, NSA1, NSMF, NDM, NMMF, NMFCT, BIGI, IMR, IMRN(11
1)

COMMON /BLK5/ CFRCHG(59), BTSPQ(59, 4), BLEND(59, 6), SE(59), GRBIND(4),
1GRBCNV(59), IGDEF, COTSP(59), QSPL1(59)

```
COMMON /BLK7/ NAD, NOD, MXC, NNODES, IPRI(20), IPRO(20), QCNT1, OP4 :
COMMON /BLK8/ NYCNT
COMMON /BLK9/ NW, NSN, NT, NPN, NNN, NMN, NMM, NMD, NDP, NMS, NTN
COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM
2XC, NNOD, NNNE, NNND, ISTPS, NYRBC, NYBASE, QT, QCNT, NQ, NIC, ISUP1, ISUP7, NS
3PM, NSC, NSK, IHCEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL4, ISC1CP, CHGCL5, ACST,
4PMCHG, RMGN, OCST2, C2MGN, RMR, PCR, PMR, MCR, OPCL1, LBCHK, IPNARC, SCL1CP, T
SSMFC, XBP, XBP3, MPDC
COMMON /BLK11/ NCL1P(59), CL1RR(59), IWORK2, START4, OCST, ILI(45), CHGP
1RC(5), TRANSR(59), OACL1T(59), CINSP(59), PMCST(59), PRCST(59), ICST(5, 1
20), HCHG2(16), MXN, IPCEF, CQC2D(59), CQC1D(59), NRPR(59), XCL2P, DTCST, IW
3ORK3, SFPCST, DFPCST, TRMPC, TMCST, RMPC(75), EFRR(59), ADTCST, ASFPCS, ADF
4PCS, ATMCST, ATRMPC
COMMON /BLK12/ ISTATE, MXP, NSA1S, NSMFS, NT1, NT2, NT3, NT4, NPN1, NDP1, NT
1N1, NT5, NT6, NT7, NT8, NT9, NT10, NT11
COMMON /BLK14/ IDXB(9, 12), UNRS(9), B(9, 4), BIMP(4, 2, 4), C(5, 5, 4), RPB(
15, 5), BS(6, 9, 4), BPC(9), BSE(9), BDE(5), BTRANS(11, 5), EBC3P(5), BCHGI(4)
2, PCHGI(4), NRPB(5), CHGMC(5, 5), SIDX(3, 4), RSS(9), CF(5), BMKP(5), BRCST(
35), BNF(2), BTCST(2, 5), ISS, UNUT, UNPC, DGSC(20), IMPHLD, UNS(4), UNGBP(9,
44), UNFPHC, UNPMC, ECHBT(5), BFPHC

INTEGER QR, QT, QCNT, QCNT1, CR, PCR, PMR, RMR, RDA, OPEXF, OPCL1, OP1, OP2, OP
13, OP4, OP5, OP6, OP7, BIGI, OP8, QRB

REAL NFP, IAQL1, MPDRM, MMFC, MPDPM, IPPM, NCL2Q, MCL2Q, MCL3Q, NRPR, NCL1P,
1MFGS, MFGM, IAQL1S, MTPSQ, MCL1Q, ICHG, MOCC, MOPC, MPPC, MRTLP, NCL2P, MBLN
2D, MCL1P, MCL1U, MIAQL1, MMDM, ICP, NCL3, NCL3P, NTPSQ, MMFG

EQUIVALENCE (SPACE1(1), WORK1(1))
EQUIVALENCE (SPMI(1), WORK2(1))
EQUIVALENCE (RDA(1), WORK4(1)), (CPRCHG(1), WORK5(1)), (NAD, WORK6(1)
1), (NYCNT, WORK7), (NW, WORK8(1)), (MMFC(1), WORK9(1)), (NCL1P(1), WOR
2K10(1)), (ISTATE, WORK11(1)), (IDXB(1), WORK12(1)), (ADTCST, WORK13(1
3))
```

DATA MODL/2H A/

TRACESUBSCRIPTS

...SET PARAMETERS FOR WORK ARRAYS:

```
IWORK1=14992
IWORK2=8025
IWORK4=24
IWORK5=890
IWORK6=45
IWORK8=11
IWORK9=407
IWORK0=852
IWORK7=18
IWORKA=767
IWORKB=10
```

```
IWORK1=SIZE OF ARRAY WORK1
IWORK2=SIZE OF ARRAY WORK2
```

...ASSIGN LOGICAL FILE PARAMETERS

IP1=8

IP2=5
IP3=10
IP4=5
OP1=8
OP2=9
OP3=6
OP5=10
OP7=12
OP8=41

IP1=MODEL BASE DATA INPUT - FEDERAL ORDER
IP2=DATA FROM INPUT FORMS - FEDERAL ORDER
IP3=MODEL BASE DATA INPUT - STATE ORDER
IP4=DATA FROM INPUT FORMS - STATE ORDER
OP1=MODEL OUTPUT FOR GRADE A REPORT WRITER - DRW1
OP2=SCRATCH FILE-OUTPUT TO DAMPCLC
OP3=ALL OTHER OUTPUT FROM DAMPSIN
OP5=SCRATCH FILE-OUTPUT TO DAMPSLU
OP7=SCRATCH FILE - OUTPUT TO DAMPCLC
OP8=BASE DATA FOR GRADE B REPORT WRITER - DRW2

REWIND OP2

...SET PARAMETERS FOR GNET

NNODES=381
NAD=4350
NNOD=382
SMXC=1607
BIGI=140737488355328
NNNE=NNODES
NNND=NAD

NNODES=NUMBER OF NODES IN NETWORK
NAD =DIMENSION FOR ARC LENGTH ARRAYS (T,C,CP) IN GNETA
SHOULD BE AT LEAST ONE GREATER THAN TOTAL NUMBER OF
ARCS IN NETWORK EXCLUDING ARCS TO AND FROM SUPER SINK
AND SUPER SOURCE
NNOD =DIMENSION OF NODE LENGTH ARRAYS (U,X,H,P,D,IT,CPX) IN
GNETA SHOULD BE AT LEAST ONE GREATER THAN NNODES
SMXC =DIMENSION OF ARRAYS (ISA,NSA) IN GNETA WHERE
SMXC=(NAD-3*NNOD)/2
BIGI =LARGEST INTEGER -MACHINE SPECIFIC- CURRENT VALUE=2**47

...SET MODEL PARAMETERS

NSA1=45
NSA1S=14
NSX1=NSA1
NSP=16
NSPX=NSP
NSMF=26
NSMFS=14
NDM=19
NMNF=27
NMFCT=28
MXN=9
MXM=10
MXP=5
NW=NSA1-NSP

NSN=2*NSA1+2*NSA1S+NSP
NT=NSA1+NSP
NT1=NT+NSA1S
NT2=NT1+NSA1
NT3=NSA1+NSA1S
NT4=NSN+NSA1
NT5=NSA1+1
NT6=NT+1
NT7=NSMF+1
NT8=NSMF+NSMFS
NT9=NSA1
NT10=NMFGT+NSMFS
NT11=8
NPN=NSN+NT3
NPN1=NPN+NSMF
NNN=NPN1+NSMFS
NMN=NNN+1
NMM=NMM+NSMF
NMD=NMM+1
NDP=NDP+NSA1
NDP1=NDP+NSA1S
NTN=NDP1+NSA1
NTN1=NTN+NSA1S
NMS=NTN1+1
NSC=NTN1+2
NSK=NTN1+3
NSPM=NSMF+NMMF
NSPM3=NSPM+3
MS=9
MIMP=2
MC=5
MSI=MS+MIMP

NSA1 =NUMBER OF FEDERAL ORDER-PROCESSING-CONSUMPTION AREAS
NSA1S =NUMBER OF STATE ORDER-PROCESSING-CONSUMPTION AREAS
NSP =NUMBER OF FEDERAL ORDER AREAS WITH SUPPLY PLANT MILK
NSMF =NUMBER OF FEDERAL SINGLE MANUFACTURING CENTERS
NSMFS =NUMBER OF STATE SINGLE MANUFACTURING CENTERS
NMMF =NUMBER OF MULTIPLE MANUFACTURING CENTERS (F.O. ONLY)
NDM =NUMBER OF FEDERAL ORDER AREAS LINKED TO MULTIPLE MFG. CENTE
NMFGT =MAX(NSMF,NMMF)+1
MXM =COLUMN DIMENSION FOR ARRAYS(LMMFG,DMMFG,MMNC)
MXN= =COLUMN DIMENSION FOR ARRAY(ICST)
MXP =ROW DIMENSION FOR ARRAY(ICST)= MAXIMUM NUMBER OF F.O. AREAS
WITH INSPECTION COSTS
MS =NUMBER OF GRADE B SUPPLY REGIONS
MIMP =NUMBER OF GRADE B PORTS OF TRADE
MC =NUMBER OF GRADE B DEMAND REGIONS

...INITIALIZE VARIABLES AND ARRAYS

NYCNT=1
QCNT=1
QCNT1=QCNT
QT=6
CL2=0.0
CL3=0.0
SCL1CP=0.
ISC1CP=0
ADTCST=0.

```
ASFFCS=0.
ASDFPCS=0.
ATMCST=0.
ATRMPC=0.
DTCST=0.
SFPCST=0.
DFPCST=0.
TRMPC=0.
TMCST=0.
ASIA1(1)=0.
ASFRRC(1)=0.
ASPC1(1)=0.
ASCL2D(1)=0.
ASCL3D(1)=0.
SMFC=0.
SSMFC=0.
AMC1U(1)=0.
AMC1P(1)=0.
AMC2P(1)=0.
AMC3P(1)=0.
AMBLND(1)=0.
AUP(1)=0.
ASNFP(1)=0.
ASRDDC(1)=0.
ASFRPS(1)=0.
ACEXP(1)=0.
AMRTL(1)=0.
AMRPS(1)=0.
AMUCP(1)=0.
ASMFGU(1)=0.
ASTRNS(1)=0.
ASDQC(1)=0.
DO 1 I=1,NMMF
DO 1 J=1,MMX
    LMMFG(I,J)=0
    DMMFG(I,J)=0.
1 CONTINUE
DO 2 I=1,MMX
DO 2 J=1,MMX
    ICST(I,J)=0
2 CONTINUE
DO 3 I=1,NSMF
3 IDMG1(I)=0
DO 4 I=1,NOM
    MMDM(I)=0.0
4 IDMG2(I)=0

...READ DATA FROM PRIMARY INPUT FORM

READ (IP2,44) TITLE,CR,ISTATE,IMFGB,IMR

...READ ORDER NAMES , ID , DISTANCES FOR GENERATING CLASS 1 PRICES
FROM A BASE PRICE USING A LOCATION DIFFERENTIAL , LAST PRICE
AND BASE YEAR FROM SECTIONS 1 AND 2 OF MODEL BASE DATA

READ (IP1,57) (ARMMS(K,25),K=1,15),(ARMMS(K,35),K=1,15)
READ (IP1,50) (ARMMS(K,1),K=1,15),(IR(I),NOR(IR(I)),ONAME(IR(I)),W
1DRK1(I),LDRK1(I+NT),AMMFG(IR(I),1),DMC1P(IR(I)),I=1,NT)
READ (IP1,43) (ARMMS(K,24),K=1,15),NYBASE
IF (ISTATE.LE.0) GO TO 5
```

```
READ (IP3,60) (ARMMD(K,1),K=1,15),(NOR(I),ONAME(I),AMMFG(I-NSP,2),
10MC1P(I),I=NT6,NT1)
5 CONTINUE
IF (IMR.LE.0) GO TO 13
BLWT=0.0
N1=0
N2=0
INNRC=0
READ (IP2,58)
DO 7 I=1,IMR
  N1=N2+1
  READ (IP2,6) INNRC,(IMRG(J),J=N1,N2+INNRC)
6 FORMAT (18I4)
  N2=N2+INNRC
7 IMRN(I)=INNRC
  IMRT=N2
  READ (IP2,11) IMRB,BLWT
  IMRN(11)=IMRB
  IF (IMRB.LE.0) GO TO 13
  IF (IMRB.NE.1) GO TO 9
  DO 8 I=1,NSA1
8 BLDIF(I)=BLWT*AMMFG(I,1)
  GO TO 13
9 READ (IP2,12) (BLDIF(IR(I)),I=1,NSA1)
  DO 10 I=1,NSA1
10 BLDIF(I)=BLDIF(I)*100.0
11 FORMAT (1X/I2,F8.0)
12 FORMAT (1X/(9F8.0))
13 IF (CR.EQ.0) READ (IP2,38) (NCL1P(IR(I)),CHGCL1(IR(I)),I=1,NSA1)
  IF (CR.EQ.0.AND.ISTATE.GT.0) READ (IP2,38) (NCL1P(I),CHGCL1(I),I=N
1TS,NT3)
  NCL2P=0.0
  CHGCL4=0.0
  NCL3P=0.0
  CHGCL5=0.0
  IF (CR.LT.2) READ (IP2,52) NCL2P,CHGCL4,NCL3P,CHGCL5
  READ (IP2,46) MR,GR,NY
  READ (IP2,47) IPRTB,OPEXF

...DETERMINE LENGTH OF RUN IN YEARS AND QUARTERS
  NYRBC=NYBASE+1
  NY=NY-NYBASE
  IF (NY.LT.1) NY=1
  IF (NY.GT.5) NY=5
  NQ=NY*4

...READ DATA FROM PRIMARY INPUT FORM, PART II
  READ (IP2,46) ISUP1,ISUP2,ISUP3,ISUP4,ISUP5,ISUP6,ISUP7,ISUP8,ISUP
19,I12A,I12B,I12C,I12D,I13A,I13B,I13C
  READ (IP2,42) PSE2,PCR,MCR,PMR,RMR,OPCL1
  IF (ISTATE.GT.0) GO TO 14
  READ (IP2,47) ICR
  GO TO 15
14 READ (IP2,47) ICR,ICRS
  READ (IP2,52) DGSC
  READ (IP2,47) ISS
  READ (IP2,47) IMPHLD
  READ (IP2,46) ORB
```



```
READ (IP2,46) ICRB  
15 CONTINUE
```

...READ DATA FROM SUPPLEMENTAL INPUT FORM F-A (OPTIONAL)

```
IKKK=ISUP1+ISUP2+ISUP3+ISUP4+ISUP5+ISUP6  
IF (IKKK.GT.0) READ (IP2,48) (APMM(IR(I),1),APMM(IR(I),2),APMM(IR(I),3),  
APMM(IR(I),4),APMM(IR(I),5),APMM(IR(I),6),I=1,NSA1)  
IF (ISUP1.EQ.1) READ (IP2,52) CL2,CL3
```

...DETERMINE IF OPTIONAL DATA IS TO BE READ IN FROM SUPPLEMENTAL INPUT FORMS F-B, F-C, OR F-D.

```
ISUP7=ISUP7+ISUP8*2+ISUP9*3  
ISUPP7=ISUP7+1  
GO TO (23,16,19,21), ISUPP7
```

...READ DATA FROM SUPPLEMENTAL INPUT FORM F-B (OPTIONAL)

```
16 READ (IP2,58)  
DO 17 J=1,NY  
IJK=(J-1)*4+1  
17 READ (IP2,36) (QCL1P(IR(I),IJK),QCL1P(IR(I),IJK+1),QCL1P(IR(I),IJK  
1+2),QCL1P(IR(I),IJK+3),I=1,NSA1)  
READ (IP2,52) (QCL2P(I),I=1,NQ)  
READ (IP2,52) (QCL3P(I),I=1,NQ)  
DO 18 J=1,NQ  
QCL2P(J)=QCL2P(J)*100.0  
QCL3P(J)=QCL3P(J)*100.0  
DO 18 I=1,NSA1  
18 QCL1P(I,J)=QCL1P(I,J)*100.0  
GO TO 23
```

...READ DATA FROM SUPPLEMENTAL INPUT FORM F-C (OPTIONAL)

```
19 READ (IP2,52) DIFF,CHGBP,BP(1)  
IF (ISUP1.EQ.1) ISUPP7=5  
BP(1)=BP(1)*100.0  
DO 20 J=1,NQ  
XBP=BP(1)+CHGBP*FLOAT(J-1)*100.0  
IF (ISUP1.EQ.1) XBP=BP(1)*(1.0+CHGBP*FLOAT(J-1)*0.01)  
DO 20 I=1,NSA1  
DIST=AMMFG(I,1)
```

...GENERATE CLASS I PRICE SURFACE

```
20 QCL1P(I,J)=QCL1P(XBP,DIFF,DIST)  
GO TO 23
```

...READ DATA FROM SUPPLEMENTAL INPUT FORM F-D (OPTIONAL)

```
21 READ (IP2,41) DIFF  
READ (IP2,52) (BP(J),J=1,NQ)  
READ (IP2,52) (QCL2P(I),I=1,NQ)  
READ (IP2,52) (QCL3P(I),I=1,NQ)  
DO 22 J=1,NQ  
QCL2P(J)=QCL2P(J)*100.0  
QCL3P(J)=QCL3P(J)*100.0  
BP(J)=BP(J)*100.0  
XBP=BP(J)
```

```
DO 22 I=1,NSA1
  DIST=AMMFG(I,1)
...GENERATE CLASS I PRICE SURFACE
  22 QCLIP(I,J)=GCLIP(XBP,DIFF,DIST)
  23 IF (ISTATE.LT.1) GO TO 31
...READ DATA FROM SUPPLEMENTAL INPUT FORM S-A (OPTIONAL)
  JKKK=I12A+I12B+I12C+I12D
  IF (JKKK.GT.0) READ (IP2,64) (SAD(I,1),SAD(I,2),SAD(I,3),SAD(I,4),
  1I=1,NSA1S)
...DETERMINE IF OPTIONAL DATA IS TO BE READ IN FROM SUPPLEMENTAL
  INPUT FORMS S-B, S-C, OR S-D.
  ISTT=1+I13A+2*I13B+3*I13C
  GO TO (31,24,27,29), ISTT
...READ DATA FROM SUPPLEMENTAL INPUT FORM S-B (OPTIONAL)
  24 READ (IP2,58)
  DO 25 J=1,NY
    IJK=(J-1)*4+1
  25 READ (IP2,36) (QCLIP(I,IJK),QCLIP(I,IJK+1),QCLIP(I,IJK+2),QCLIP(I,
  1IJK+3),I=NT5,NT3)
  DO 26 J=1,NQ
    DO 26 I=NT5,NT3
  26 QCLIP(I,J)=QCLIP(I,J)*100.0
  GO TO 31
...READ DATA FROM SUPPLEMENTAL INPUT FORM S-C (OPTIONAL)
  27 READ (IP2,52) DIFFS,CHGBPS,BPS(1)
  BPS(1)=BPS(1)*100.0
  DO 28 J=1,NQ
    XBPS=BPS(1)+CHGBPS*FLOAT(J-1)*100.0
    IF (I12A.EQ.1) XBPS=BPS(1)*(1.0+CHGBPS*FLOAT(J-1)*0.01)
  DO 28 I=NT5,NT3
    DIST=AMMFG(I,2)
...GENERATE CLASS I PRICE SURFACE
  28 QCLIP(I,J)=GCLIP(XBPS,DIFFS,DIST)
  GO TO 31
...READ DATA FROM SUPPLEMENTAL INPUT FORM S-D (OPTIONAL)
  29 READ (IP2,41) DIFFS
  READ (IP2,52) (BPS(J),J=1,NQ)
  DO 30 J=1,NQ
    BPS(J)=BPS(J)*100.0
    XBPS=BPS(J)
  DO 30 I=NT5,NT3
    DIST=AMMFG(I,2)
...GENERATE CLASS I PRICE SURFACE
```

30 GCL1P(I, J)=GCL1P(XBPS, DIFFS, DISTS)

31 DO 32 I=1, NT3
NCL1P(I)=NCL1P(I)*100.0
32 CHGCL1(I)=CHGCL1(I)*100.0
NCL2P=NCL2P*100.0
CHGCL4=CHGCL4*100.0
NCL3P=NCL3P*100.0
CHGCL5=CHGCL5*100.0

...INPUT FORMATS

33 FORMAT (15A5/(10X, 4F11.0))
34 FORMAT (15A5/(20I2))
35 FORMAT (15A5/(10X, 5F8.0))
36 FORMAT (10X, 4F8.0)
37 FORMAT (15A5/(10X, 2F8.0))
38 FORMAT (1X/(10X, 2F8.0))
39 FORMAT (15A5/(10X, 5F8.0))
40 FORMAT (15A5/(10X, F8.0))
41 FORMAT (1X/(10X, F8.0))
42 FORMAT (1X/(F8.0, 5I2))
43 FORMAT (15A5/(10X, I4))
44 FORMAT (1X/(A6, 5I2))
45 FORMAT (15A5/(25I2, /, I6))
46 FORMAT (1X/25I2/I6)
47 FORMAT (1X/(10I2, /, 10I2))
48 FORMAT (1X/(6F8.0))
49 FORMAT (15A5/(10X, 4F8.0))
50 FORMAT (15A5/(I2, I3, A10, 5X, A3, 2X, A10, 10X, F5.0, 5X, F5.0))
51 FORMAT (I3, 7X, I6I3)
52 FORMAT (1X/(10X, 4F8.0))
53 FORMAT (10X, 12F5.0)
54 FORMAT (5X, 15F5.0)
55 FORMAT (10X, 17I4)
56 FORMAT (10X, 10I5)
57 FORMAT (15A5)
58 FORMAT (1X)
59 FORMAT (15A5/(10X, 9F5.0))
60 FORMAT (15A5/(I3, 2X, A10, F5.0, 5X, F5.0))
61 FORMAT (15A5/(10X, 8F3.0))
62 FORMAT (15A5/(10X, 8F5.0))
63 FORMAT (15A5/(2X, 12I4))
64 FORMAT (1X/(4F8.0))

...READ IN BASE DATA
START1=CTIME(T1)

READ (IP1, 33) (ARMMS(K, 2), K=1, 15), ((IACL1(I, J), J=1, 4), I=1, NSA1)
READ (IP1, 35) (ARMMS(K, 3), K=1, 15), ((CL1P(I, J), J=1, 5), I=1, NSA1)
READ (IP1, 35) (ARMMS(K, 4), K=1, 15), ((BLEND(I, J), J=1, 5), I=1, NSA1)
READ (IP1, 33) (ARMMS(K, 5), K=1, 15), ((TFSQ(I, J), J=1, 4), I=1, NSA1)
READ (IP1, 33) (ARMMS(K, 6), K=1, 15), ((CL1Q(I, J), J=1, 4), I=1, NSA1)
READ (IP1, 33) (ARMMS(K, 7), K=1, 15), ((CL2Q(I, J), J=1, 4), I=1, NSA1)
READ (IP1, 57) (ARMMS(K, 8), K=1, 15)
DO 65 I=1, NT
65 READ (IP1, 54) (SPMI(I, J), J=1, NSA1)
READ (IP1, 57) (ARMMS(K, 9), K=1, 15)
DO 66 I=1, NSA1
66 READ (IP1, 54) (PDMI(I, J), J=1, NSA1)

```
READ (IP1,57) (ARMMS(K,10),K=1,15)
DO 67 I=1,NMMF
67 READ (IP1,55) (LMMFG(I,J),J=1,MXM)
READ (IP1,57) (ARMMS(K,11),K=1,15)
DO 68 I=1,NMMF
68 READ (IP1,53) (DMMFG(I,J),J=1,MXM)
READ (IP1,57) (ARMMS(K,23),K=1,15)
DO 69 I=1,NMMF
69 READ (IP1,56) (MMNC(I,J),J=1,MXM)
READ (IP1,49) (ARMMS(K,12),K=1,15),((SPLB(I,J),J=1,4),I=1,NSP)
READ (IP1,39) (ARMMS(K,13),K=1,15),((A(I,J),J=1,5),I=1,NSA1)
READ (IP1,57) (ARMMS(K,14),K=1,15)
DO 70 I=1,NXP
70 READ (IP1,51) IDICST(I),(ICST(I,J),J=1,MXN)
READ (IP1,40) (ARMMS(K,15),K=1,15),(CINSP(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,16),K=1,15),(DISTB(I),I=1,NT)
READ (IP1,40) (ARMMS(K,17),K=1,15),(DISTP(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,18),K=1,15),(ERCST(I),I=1,NT)
READ (IP1,40) (ARMMS(K,19),K=1,15),(PRCST(I),I=1,NSA1)
READ (IP1,37) (ARMMS(K,20),K=1,15),(HCHG1(I),HCHG2(I),I=1,NSP)
READ (IP1,40) (ARMMS(K,21),K=1,15),(SPRC(I),I=1,NSP)
READ (IP1,40) (ARMMS(K,22),K=1,15),(MPDRM(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,26),K=1,15),(MMFC(I),I=1,NMMF)
READ (IP1,40) (ARMMS(K,27),K=1,15),(SMFC(I),I=1,NSMF)
READ (IP1,40) (ARMMS(K,28),K=1,15),(CL1CP(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,29),K=1,15),(MPDPM(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,30),K=1,15),(RMPC(I),I=1,NT)
READ (IP1,40) (ARMMS(K,31),K=1,15),(SE(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,32),K=1,15),(DE(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,33),K=1,15),(IPPM(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,34),K=1,15),(CL1RR(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,36),K=1,15),AR,BR,AP,BP1,(BC2P(I),I=1,5),(B
IC3P(I),I=1,4)
READ (IP1,40) (ARMMS(K,37),K=1,15),RMCN,OCST,C2MGN,OCST2,SE2,ACST
READ (IP1,40) (ARMMS(K,38),K=1,15),DISTD,DCHG1,ECHG,PMCHG
READ (IP1,37) (ARMMS(K,40),K=1,15),(CHGNC1(I),CHGNC2(I),I=1,NSA1)
READ (IP1,40) (ARMMS(K,41),K=1,15),(GRBCNU(I),I=1,NSA1)
READ (IP1,49) (ARMMS(K,42),K=1,15),(GRBIND(I),I=1,4)
READ (IP1,49) (ARMMS(K,43),K=1,15),(CHGBT(I),CHGPT(I),CHGHC(I),CHG
IPRC(I),I=1,5)
READ (IP1,49) (ARMMS(K,44),K=1,15),(CHGPD(I),I=1,20)
READ (IP1,45) (ARMMS(K,39),K=1,15),IPNARC,LECHK,MPDC,IC1EF,IC2EF,I
IPCEF,IGBEF,IBTEF,IPTEF,IHCEF,IPDEF
READ (IP1,34) (ARMMS(K,45),K=1,15),(IPRI(I),I=1,20),(IPRO(I),I=1,2
10)
```

...READ IN BASE DATA FOR STATES

IF (ISTATE.LE.0) GO TO 80

```
NT9=NT3
READ (IP3,35) (ARMMD(K,2),K=1,15),((CL1P(I,J),J=1,5),I=NT5,NT3)
READ (IP3,35) (ARMMD(K,3),K=1,15),((BLEND(I,J),J=1,5),I=NT5,NT3)
READ (IP3,33) (ARMMD(K,4),K=1,15),((TPSQ(I,J),J=1,4),I=NT5,NT3)
READ (IP3,33) (ARMMD(K,5),K=1,15),((IACL1(I,J),J=1,4),I=NT5,NT3)
READ (IP3,33) (ARMMD(K,6),K=1,15),((CL2Q(I,J),J=1,4),I=NT5,NT3)
READ (IP3,39) (ARMMD(K,7),K=1,15),((A(I,J),J=1,5),I=NT5,NT3)
READ (IP3,40) (ARMMD(K,8),K=1,15),(DISTB(I),I=NT6,NT1)
READ (IP3,40) (ARMMD(K,9),K=1,15),(DISTP(I),I=NT5,NT3)
READ (IP3,40) (ARMMD(K,10),K=1,15),(ERCST(I),I=NT6,NT1)
```

```
READ (IP3,40) (ARMMD(K, 11),K=1, 15), (PRCST(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 12),K=1, 15), (MPDRM(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 13),K=1, 15), (SMFC(I), I=NT7, NT8)
READ (IP3,40) (ARMMD(K, 14),K=1, 15), (CL1CP(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 15),K=1, 15), (MPDPH(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 16),K=1, 15), (RMPC(I), I=NT6, NT1)
READ (IP3,40) (ARMMD(K, 17),K=1, 15), (SE(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 18),K=1, 15), (DE(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 19),K=1, 15), (CL1RR(I), I=NT5, NT3)
READ (IP3,37) (ARMMD(K, 20),K=1, 15), (CHGNC1(I), CHGNC2(I), I=NT5, NT3)
READ (IP3,40) (ARMMD(K, 21),K=1, 15), (GRBCNU(I), I=NT5, NT3)
LL=22
DO 71 KK=1,7,2
  READ (IP3,61) (ARMMD(K, LL),K=1, 15), ((SSPMI(I, J, KK), J=1, NT11), I=
1 1, NSA15)
  READ (IP3,62) (ARMMD(K, LL+1),K=1, 15), ((SSPMI(I, J, KK+1), J=1, NT11
1 ), I=1, NSA15)
71 LL=LL+2
  READ (IP3,63) (ARMMD(K, 30),K=1, 15), ((IDX(I, J), J=1, 12), I=1, MS)

...READ IN BASE DATA FOR UNREGULATED

READ (IP3,33) (ARMMD(K, 31),K=1, 15), (UNS(J), J=1, 4)
READ (IP3,59) (ARMMD(K, 32),K=1, 15), (UNRS(I), I=1, MS)
READ (IP3,40) (ARMMD(K, 33),K=1, 15), UNUT
READ (IP3,39) (ARMMD(K, 34),K=1, 15), UNFC, UNFPHC, UNPMC

...READ IN BASE DATA FOR GRADE B

IF (IMFGB.LE.0) GO TO 80

READ (IP3,33) (ARMMD(K, 35),K=1, 15), ((B(I, J), J=1, 4), I=1, MS)
DO 72 L=1,4
72 READ (IP3,33) (ARMMD(K, 35+L),K=1, 15), ((BIMP(L, I, J), J=1, 4), I=1, MIMP
1)
DO 73 KK=1,5
73 READ (IP3,33) (ARMMD(K, 35+KK),K=1, 15), ((C(KK, I, J), J=1, 4), I=1, MC)
  READ (IP3,35) (AMMFG(K, 3),K=1, 15), ((RPB(I, J), J=1, 5), I=1, 5)
DO 74 KK=1,6
74 READ (IP3,33) (AMMFG(K, 3+KK),K=1, 15), ((BS(KK, I, J), J=1, 4), I=1, MS)
  READ (IP3,40) (AMMFG(K, 10),K=1, 15), (BPC(I), I=1, MS)
  READ (IP3,40) (AMMFG(K, 11),K=1, 15), (BSE(I), I=1, MS)
  READ (IP3,40) (AMMFG(K, 12),K=1, 15), (BDE(I), I=1, MC)
  READ (IP3,35) (AMMFG(K, 13),K=1, 15), ((BTRANS(I, J), J=1, 5), I=1, MS1)
  READ (IP3,40) (AMMFG(K, 14),K=1, 15), (BCHGI(I), I=1, 4), (PCHGI(I), I=1,
14)
  READ (IP3,35) (AMMFG(K, 15),K=1, 15), ((CHGMC(I, J), J=1, MC), I=1, 5)
  READ (IP3,33) (AMMFG(K, 16),K=1, 15), ((SIDX(I, J), J=1, 4), I=1, 3)
  READ (IP3,40) (AMMFG(K, 17),K=1, 15), (RSS(J), J=1, MS)
  READ (IP3,39) (AMMFG(K, 18),K=1, 15), (CF(I), I=1, 5)
  READ (IP3,39) (AMMFG(K, 19),K=1, 15), (BMKP(I), I=1, 5), (BRCST(I), I=1, 5
1), (BNF(I), I=1, 2)
  READ (IP3,39) (AMMFG(K, 20),K=1, 15), ((BTCST(I, J), J=1, 5), I=1, 2)
  READ (IP3,39) (AMMFG(K, 21),K=1, 15), BFPHC, BPHC, BOPC

...CONVERT CHEESE AND GIII DATA TO ME (F)

DO 77 J=1,4
DO 75 K=1,MS
  BS(1,K, J)=BS(1,K, J)*100./CF(E)
```

```
BS(4,K,J)=BS(4,K,J)*100./CF(2)
75 CONTINUE
DO 76 K=1,MC
  C(2,K,J)=C(2,K,J)*100./CF(2)
  C(3,K,J)=C(3,K,J)*100./CF(3)
76 CONTINUE
DO 77 K=1,MIMP
  BIMP(2,K,J)=BIMP(2,K,J)*100.0/CF(3)
77 CONTINUE
DO 78 J=1,20
  DGSC(J)=DGSC(J)*100.0/CF(2)
78 CONTINUE
```

...CONVERT BUTTER AND NFDM RETAIL PRICES TO PRODUCT WEIGHT

```
DO 79 J=1,5
DO 79 I=4,5
  RPB(I,J)=RPB(I,J)/CF(I)
79 CONTINUE
80 CONTINUE
```

TIME1=START1-CTIME(T1)

...LIMIT COMPARATIVE REPORTS TO FIVE VARIABLES

```
NIC=0
DO 82 I=1,10
  IF (ICR(I).EQ.0.AND.ICRS(I).EQ.0) GO TO 82
  IF (NIC.GE.5) GO TO 81
  NIC=NIC+1
  GO TO 82
81 ICR(I)=0
  ICRS(I)=0
82 CONTINUE
```

...DETERMINE INTERNAL POSITIONS OF ORDERS WITH INSPECTION COSTS

```
J=1
DO 83 I=1,NSAI
  IF (J.GT.MXP) GO TO 84
  IF (CINSP(I).GT.0.0) ILI(I)=J
83 IF (CINSP(I).GT.0.0) J=J+1
84 CONTINUE
REWIND OP1
```

...DETERMINE INTERNAL INDEX NUMBERS FOR SUPPLY AREAS LINKED TO SINGLE AND MULTIPLE MFG. CENTERS

```
K=1
K4=1
DO 87 I=1,NSMF
DO 85 J=K,NSAI
  IF (WORK1(I+NT).NE.GNAME(J)) GO TO 85
  IDMG1(I)=J
  K2=J
  GO TO 87
85 IDMG2(K4)=J
  K4=K4+1
  IF (K4.GT.NDM+1) WRITE (OP3,88) WORK1(I+NT)
  IF (K4.GT.NDM+1) STOP
```

```
86 CONTINUE
87 K=K2+1
88 FORMAT (1X,5HLABEL,3X,A10,1X,32HFROM COLUMNS 26-35 IN THE FIRST ,2
12HSECTION OF BASE DATA , ,/19X,1SHDOES NOT MATCH ITS ,54HCOUNTERPAR
2T IN COLUMNS 6-15. LABELS MUST BE IDENTICAL.)

...DETERMINE DISTANCES BETWEEN SUPPLY AREAS AND MULTIPLE
MANUFACTURING DUMMY

DO 92 I=1,NMMF
K4=0
DO 91 J=1,MMX
DO 90 K=1,NDM
IF (LMMFG(I,J).NE.IDMG2(K)) GO TO 90
N2=1
IF (LMMFG(I,J).GT.NN) N2=2
DO 89 K1=1,N2
IF (DMMFG(I,J+K4).GT.MMDM(K)) MMDM(K)=DMMFG(I,J+K4)
89 IF (LMMFG(I,J).GT.NN) K4=K4+1
GO TO 91
90 CONTINUE
91 CONTINUE
92 CONTINUE
DO 93 I=1,NDM
93 MMDM(I)=MMDM(I)+100.

...REPLACE BASE DATA WITH SUPPLEMENTAL INPUT

DO 94 I=1,NSA1
IF (ISUP2.EQ.1) DE(I)=APMM(I,2)
IF (ISUP3.EQ.1) SE(I)=APMM(I,3)
IF (ISUP6.EQ.1) CLIRR(I)=APMM(I,6)
IF (I.GT.NSP) GO TO 94
IF (ISUP4.EQ.1) HCHG1(I)=APMM(I+NW,4)*100.0
IF (ISUP5.EQ.1) HCHG2(I)=APMM(I+NW,5)*100.0
APMM(I+NW,4)=HCHG1(I)
APMM(I+NW,5)=HCHG2(I)
94 CONTINUE

...REPLACE STATE BASE DATA WITH SUPPLEMENTAL INPUT

DO 95 I=NT5,NT3
IF (I12A.EQ.1) CHCCL1(I)=SAD(I-NSA1,1)*0.01
IF (I12B.EQ.1) GE(I)=SAD(I-NSA1,2)
IF (I12C.EQ.1) SE(I)=SAD(I-NSA1,3)
IF (I12D.EQ.1) CLIRR(I)=SAD(I-NSA1,4)
95 CONTINUE
IX=NYBASE+1
DO 96 I=1,5
NYID(I)=IX
96 IX=IX+1

...WRITE BASE AND INPUT FORM DATA TO FILE OP1 FOR USE BY REPORT WRITER

WRITE (OP1) TITLE,NY,QR,MR,CR,ISTATE,NSA1,NSA1S,NSP,ICR,ICRS,NIC,I
1SUPP7,NMF,NMMF,NT,NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10,NT11,N
2DM,NMFG,NYBASE,IMR,MODL,IMFCB
IF (IMR.GT.0) WRITE (OP1) (IMRN(I),I=1,IMR),IMRT,(IMRG(I),I=1,IMRT
1),IMRB,ELWT,(BLDIF(IR(I)),I=1,NSA1)
WRITE (OP1) (IR(I),NOR(IR(I)),ONAME(IR(I)),I=1,NT),(IDMG2(I),I=1,N
```

```

10M), (WORK1(I), WORK1(I+NT), I=1, NSPM3)
IF (ISTATE.GT.0) WRITE (OP1) (NOR(I), ONAME(I), I=NT5, NT1)
WRITE (OP1) (OMC1P(IR(I)), NCL1P(IR(I)), CHGCL1(IR(I)), I=1, NSA1)
IF (ISTATE.GT.0) WRITE (OP1) (OMC1P(I+NSP), NCL1P(I), CHGCL1(I), I=NT
15, NT3), I13A, I13B, I13C
WRITE (OP1) BC2P(5), NCL2P, CHGCL4, BC3P(4), NCL3P, CHGCL5, IPRTB, OPEXF
WRITE (OP1) (APMM(IR(I), 1), DE(IR(I)), SE(IR(I)), APMM(IR(I), 4), APMM(
1IR(I), 5), CL1RR(IR(I)), I=1, NSA1)
WRITE (OP1) CL2, CL3, SE2, PSE2, PCR, MCR, PMR, RMR, OPCL1, ISUP1, ISUP2, ISU
1P3, ISUP4, ISUP5, ISUP6, I12A, I12B, I12C, I12D
IF (IMFCB.NE.0) WRITE (OP1) QRE, ICRB, ISS, DGSC, IMPHLD
IF (ISUP7.EQ.1) WRITE (OP1) ((QCL1P(IR(I), J), I=1, NSA1), J=1, NQ), (QC
1L2P(I), I=1, NQ), (QCL3P(I), I=1, NQ)
IF (ISUP7.EQ.1) WRITE (OP1) ((QCL1P(IR(I), J), I=1, NSA1), J=1, NQ)
IF (ISUP7.EQ.2) WRITE (OP1) BP(1), CHGBP, DIFF
IF (ISUP7.EQ.3) WRITE (OP1) (BP(I), I=1, NQ), DIFF, (QCL2P(I), I=1, NQ),
1(QCL3P(I), I=1, NQ)
IF (ISTATE.EQ.0) GO TO S7
WRITE (OP1) (SAD(I-NSA1, 1), DE(I), SE(I), CL1RR(I), I=NT5, NT3)
IF (ISTT.GT.1) WRITE (OP1) ((QCL1P(I, J), I=NT5, NT3), J=1, NQ)
IF (I13B.EQ.1) WRITE (OP1) BPS(1), CHGBPS, DIFFS
IF (I13C.EQ.1) WRITE (OP1) (BPS(I), I=1, NQ), DIFFS
WRITE (OP1) UNUT

```

...PRINT BASE DATA (OPTIONAL)

97 IF (IPRTB.NE.1) GO TO 144

```

N1=NMFGT
LT1=N1+NT
WRITE (OP3, 108)
WRITE (OP3, 109) (ARMMS(K, 25), K=1, 15), (ARMMS(K, 35), K=1, 15)
WRITE (OP3, 121) (ARMMS(K, 1), K=1, 15), (IR(I), NOR(IR(I)), ONAME(IR(I))
1, WORK1(I), WORK1(I+NT), AMMFG(I, 1), OMC1P(I), I=1, NT)
WRITE (OP3, 121) (ARMMS(K, 24), K=1, 15), NYBASE
WRITE (OP3, 130) (ARMMS(K, 2), K=1, 15)
WRITE (OP3, 110) (NOR(I), ONAME(I), (IACL1(I, J), J=1, 4), I=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 3), K=1, 15)
WRITE (OP3, 112) (NOR(I), ONAME(I), (CL1P(I, J), J=1, 5), I=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 4), K=1, 15)
WRITE (OP3, 112) (NOR(I), ONAME(I), (BLEND(I, J), J=1, 5), I=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 5), K=1, 15)
WRITE (OP3, 110) (NOR(I), ONAME(I), (TPSQ(I, J), J=1, 4), I=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 6), K=1, 15)
WRITE (OP3, 110) (NOR(I), ONAME(I), (CL1Q(I, J), J=1, 4), I=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 7), K=1, 15)
WRITE (OP3, 110) (NOR(I), ONAME(I), (CL2Q(I, J), J=1, 4), I=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 8), K=1, 15)
DO 98 I=1, NT
98 WRITE (OP3, 125) NOR(I), ONAME(I), (SPMI(I, J), J=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 9), K=1, 15)
DO 99 I=1, NSA1
99 WRITE (OP3, 125) NOR(I), ONAME(I), (PDMI(I, J), J=1, NSA1)
WRITE (OP3, 130) (ARMMS(K, 10), K=1, 15)
DO 100 I=1, NMMF
100 WRITE (OP3, 118) WORK1(I+N1), WORK1(I+LT1), (LMMFG(I, J), J=1, MXM)
WRITE (OP3, 130) (ARMMS(K, 11), K=1, 15)
DO 101 I=1, NMMF
101 WRITE (OP3, 117) WORK1(I+N1), WORK1(I+LT1), (DMMFG(I, J), J=1, MXM)
WRITE (OP3, 130) (ARMMS(K, 23), K=1, 15)

```



```
DO 102 I=1,NMNF
102 WRITE (OP3,118) WORK1(I+N1),WORK1(I+LT1), (MMNC(I,J),J=1,MXM)
WRITE (OP3,130) (ARMMS(K,12),K=1,15)
WRITE (OP3,116) (NOR(I),ONAME(I), (SPLB(I,J),J=1,4), I=1,NSP)
WRITE (OP3,130) (ARMMS(K,13),K=1,15)
WRITE (OP3,111) (NOR(I),ONAME(I), (A(I,J),J=1,5), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,14),K=1,15)
DO 103 I=1,MXP
103 WRITE (OP3,115) IDICST(I), (ICST(I,J),J=1,MXN)
WRITE (OP3,130) (ARMMS(K,15),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),CINSP(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,16),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),DISTB(I), I=1,NT)
WRITE (OP3,130) (ARMMS(K,17),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),DISTP(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,18),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),ERCST(I), I=1,NT)
WRITE (OP3,130) (ARMMS(K,19),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),PRCST(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,20),K=1,15)
WRITE (OP3,122) (NOR(I),ONAME(I),HCHG1(I),HCHG2(I), I=1,NSP)
WRITE (OP3,130) (ARMMS(K,21),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),SPRC(I), I=1,NSP)
WRITE (OP3,130) (ARMMS(K,22),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),MPDRM(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,26),K=1,15)
WRITE (OP3,119) (WORK1(I+N1),WORK1(I+LT1),MMFC(I), I=1,NMNF)
WRITE (OP3,130) (ARMMS(K,27),K=1,15)
WRITE (OP3,119) (WORK1(I),WORK1(I+NT),SMFC(I), I=1,NSMF)
WRITE (OP3,130) (ARMMS(K,28),K=1,15)
WRITE (OP3,113) (NOR(I),ONAME(I),CLICP(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,29),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),MPDPM(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,30),K=1,15)
WRITE (OP3,113) (NOR(I),ONAME(I),RMPFC(I), I=1,NT)
WRITE (OP3,130) (ARMMS(K,31),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),SE(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,32),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),DE(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,33),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),IFPM(I), I=1,NSA1)
WRITE (OP3,130) (ARMMS(K,34),K=1,15)
WRITE (OP3,114) (NOR(I),ONAME(I),CLIRR(I), I=1,NSA1)
WRITE (OP3,128) (ARMMS(K,36),K=1,15), AR, BR, AP, BP1, (BC2P(I), I=1,5),
1(BC3P(I), I=1,4)
WRITE (OP3,123) (ARMMS(K,37),K=1,15), RMGN, OCST, C2MGN, OCST2, SE2
WRITE (OP3,123) (ARMMS(K,38),K=1,15), DISTD, DCHG1, ECHG, PMCHG
WRITE (OP3,125) (ARMMS(K,40),K=1,15), (NOR(I),ONAME(I),CHGNC1(I), CH
1GNC2(I), I=1,NSA1)
WRITE (OP3,127) (ARMMS(K,41),K=1,15), (NOR(I),ONAME(I),GRBCNU(I), I=
11,NSA1)
WRITE (OP3,133) (ARMMS(K,42),K=1,15), (GRBIND(I), I=1,4)
WRITE (OP3,129) (ARMMS(K,43),K=1,15), (NYID(I),CHGSI(I),CHGPT(I), CH
1GHC(I),CHGPRC(I), I=1,5)
WRITE (OP3,133) (ARMMS(K,44),K=1,15), (CHGPD(I), I=1,20)
WRITE (OP3,124) (ARMMS(K,39),K=1,15), IPNARC, LBCHK, MPDC, IC1EF, IC2EF
1, IPDEF, IGDEF, IUTEF, IFTEF, INDEF, IPDEF
WRITE (OP3,132) (ARMMS(K,45),K=1,15), (IFRI(I), I=1,20), (IPRO(I), I=1
1,20)
WRITE (OP3,134)
```

...PRINT STATE BASE DATA (OPTIONAL)

IF (ISTATE.LE.0) GO TO 144

```
WRITE (OP3,135)
WRITE (OP3,135) (ARMMD(K,1),K=1,15), (NOR(I),ONAME(I),OMC1P(I),I=NT
16,NT1)
WRITE (OP3,130) (ARMMD(K,2),K=1,15)
WRITE (OP3,112) (NOR(I+NSP),ONAME(I+NSP), (CL1P(I,J),J=1,5), I=NT5,N
1T3)
WRITE (OP3,130) (ARMMD(K,3),K=1,15)
WRITE (OP3,112) (NOR(I+NSP),ONAME(I+NSP), (BLEND(I,J),J=1,5), I=NT5,
1NT3)
WRITE (OP3,130) (ARMMD(K,4),K=1,15)
WRITE (OP3,110) (NOR(I+NSP),ONAME(I+NSP), (TPSQ(I,J),J=1,4), I=NT5,N
1T3)
WRITE (OP3,130) (ARMMD(K,5),K=1,15)
WRITE (OP3,110) (NOR(I+NSP),ONAME(I+NSP), (CL1Q(I,J),J=1,4), I=NT5,N
1T3)
WRITE (OP3,130) (ARMMD(K,6),K=1,15)
WRITE (OP3,110) (NOR(I+NSP),ONAME(I+NSP), (CL2Q(I,J),J=1,4), I=NT5,N
1T3)
WRITE (OP3,130) (ARMMD(K,7),K=1,15)
WRITE (OP3,111) (NOR(I+NSP),ONAME(I+NSP), (A(I,J),J=1,5), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,8),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),DISTB(I), I=NT6,NT1)
WRITE (OP3,130) (ARMMD(K,9),K=1,15)
WRITE (OP3,120) (NOR(I+NSP),ONAME(I+NSP),DISTP(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,10),K=1,15)
WRITE (OP3,120) (NOR(I),ONAME(I),ERCST(I), I=NT6,NT1)
WRITE (OP3,130) (ARMMD(K,11),K=1,15)
WRITE (OP3,120) (NOR(I+NSP),ONAME(I+NSP),PRCST(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,12),K=1,15)
WRITE (OP3,114) (NOR(I+NSP),ONAME(I+NSP),MPDRM(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,13),K=1,15)
WRITE (OP3,113) (NOR(I+NSMF),ONAME(I+NSMF),SMFC(I), I=NT7,NT8)
WRITE (OP3,130) (ARMMD(K,14),K=1,15)
WRITE (OP3,113) (NOR(I+NSP),ONAME(I+NSP),CL1CP(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,15),K=1,15)
WRITE (OP3,114) (NOR(I+NSP),ONAME(I+NSP),MPDPM(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,16),K=1,15)
WRITE (OP3,113) (NOR(I),ONAME(I),RMPC(I), I=NT6,NT1)
WRITE (OP3,130) (ARMMD(K,17),K=1,15)
WRITE (OP3,114) (NOR(I+NSP),ONAME(I+NSP),SE(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,18),K=1,15)
WRITE (OP3,114) (NOR(I+NSP),ONAME(I+NSP),DE(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,19),K=1,15)
WRITE (OP3,114) (NOR(I+NSP),ONAME(I+NSP),CL1RR(I), I=NT5,NT3)
WRITE (OP3,126) (ARMMD(K,20),K=1,15), (NOR(I+NSP),ONAME(I+NSP),CHGN
1C1(I),CHGN2(I), I=NT5,NT3)
WRITE (OP3,127) (ARMMD(K,21),K=1,15), (NOR(I+NSP),ONAME(I+NSP),GRBC
1NU(I), I=NT5,NT3)
WRITE (OP3,130) (ARMMD(K,22),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT), (SSPMI(I,J,1),J=1,NT11), I=1
1,NSA15)
WRITE (OP3,130) (ARMMD(K,23),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT), (SSPMI(I,J,2),J=1,NT11), I=1
1,NSA15)
WRITE (OP3,130) (ARMMD(K,24),K=1,15)
```

```
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT),(SSPMI(I,J,3),J=1,NT11),I=1
1,NSA1S)
WRITE (OP3,130) (ARMMD(K,25),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT),(SSPMI(I,J,4),J=1,NT11),I=1
1,NSA1S)
WRITE (OP3,130) (ARMMD(K,26),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT),(SSPMI(I,J,5),J=1,NT11),I=1
1,NSA1S)
WRITE (OP3,130) (ARMMD(K,27),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT),(SSPMI(I,J,6),J=1,NT11),I=1
1,NSA1S)
WRITE (OP3,130) (ARMMD(K,28),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT),(SSPMI(I,J,7),J=1,NT11),I=1
1,NSA1S)
WRITE (OP3,130) (ARMMD(K,29),K=1,15)
WRITE (OP3,137) (NOR(I+NT),ONAME(I+NT),(SSPMI(I,J,8),J=1,NT11),I=1
1,NSA1S)
WRITE (OP3,130) (ARMMD(K,30),K=1,15)
WRITE (OP3,139) ((IDX(I,J),J=1,12),I=1,MS)
WRITE (OP3,130) (ARMMD(K,31),K=1,15)
WRITE (OP3,140) (UNS(J),J=1,4)
WRITE (OP3,130) (ARMMD(K,32),K=1,15)
WRITE (OP3,143) (UNRS(I),I=1,MS)
WRITE (OP3,130) (ARMMD(K,33),K=1,15)
WRITE (OP3,143) UNUT
WRITE (OP3,130) (ARMMD(K,34),K=1,15)
WRITE (OP3,141) UNPC,UNFPAC,UNPMC
```

...PRINT GRADE B DATA

```
IF (IMFGB.LE.0) GO TO 144
WRITE (OP3,120) (ARMMD(K,35),K=1,15)
WRITE (OP3,140) ((BC(I,J),J=1,4),I=1,MS)
DO 104 KK=1,4
WRITE (OP3,130) (ARMMD(K,35+KK),K=1,15)
104 WRITE (OP3,140) ((BIMP(KK,I,J),J=1,4),I=1,MIMP)
DO 105 KK=1,5
WRITE (OP3,130) (ARMMD(K,38+KK),K=1,15)
105 WRITE (OP3,140) ((C(KK,I,J),J=1,4),I=1,MC)
WRITE (OP3,130) (AMMFC(K,3),K=1,15)
DO 106 I=1,5
106 WRITE (OP3,141) (RPE(I,J),J=1,5)
DO 107 KK=1,6
WRITE (OP3,130) (AMMFC(K,3+KK),K=1,15)
107 WRITE (OP3,140) ((CS(KK,I,J),J=1,4),I=1,MS)
WRITE (OP3,130) (AMMFC(K,10),K=1,15)
WRITE (OP3,143) (IPC(I),I=1,MS)
WRITE (OP3,130) (AMMFC(K,11),K=1,15)
WRITE (OP3,143) (BSE(I),I=1,MS)
WRITE (OP3,130) (AMMFC(K,12),K=1,15)
WRITE (OP3,143) (PSE(I),I=1,MC)
WRITE (OP3,130) (AMMFC(K,13),K=1,15)
WRITE (OP3,141) ((BTRANS(I,J),J=1,5),I=1,MSI)
WRITE (OP3,130) (AMMFC(K,14),K=1,15)
WRITE (OP3,143) (ECHGI(I),I=1,4),(TCHGI(I),I=1,4)
WRITE (OP3,130) (AMMFC(K,15),K=1,15)
WRITE (OP3,142) ((CHGMC(I,J),J=1,5),I=1,5)
WRITE (OP3,130) (AMMFC(K,16),K=1,15)
WRITE (OP3,142) ((SIDK(I,J),J=1,4),I=1,3)
```

```
WRITE (OP3,130) (AMMFG(K,17),K=1,15)
WRITE (OP3,142) (RSS(J),J=1,MS)
WRITE (OP3,130) (AMMFG(K,18),K=1,15)
WRITE (OP3,141) (CF(I),I=1,5)
WRITE (OP3,130) (AMMFG(K,19),K=1,15)
WRITE (OP3,141) (BNKP(I),I=1,5),(BRCST(I),I=1,5)
WRITE (OP3,130) (AMMFG(K,20),K=1,15)
WRITE (OP3,141) ((BTCST(I,J),J=1,5),I=1,2)
WRITE (OP3,130) (AMMFG(K,21),K=1,15)
WRITE (OP3,141) BFPHC,BPMC,BOPC
WRITE (OP3,136)
```

...PRINT FORMATS

```
108 FORMAT (1X,31H***** BASE DATA ***** ,////)
109 FORMAT (1X,15A5)
110 FORMAT (1X/(1X,I3,3X,A10,4X,4F15.0))
111 FORMAT (1X/(1X,I3,3X,A10,4X,5F11.1))
112 FORMAT (1X/(1X,I3,3X,A10,4X,5F11.0))
113 FORMAT (1X/(1X,I3,3X,A10,4X,F15.0))
114 FORMAT (1X/(1X,I3,3X,A10,4X,F15.3))
115 FORMAT (1X,I3,20X,9I6)
116 FORMAT (1X/(1X,I3,3X,A10,4X,4F8.2))
117 FORMAT (1X,A3,3X,A10,4X,10F8.0)
118 FORMAT (1X,A3,3X,A10,4X,10I8)
119 FORMAT (1X/(1X,A3,3X,A10,4X,F10.0))
120 FORMAT (1X/(1X,I3,3X,A10,4X,F8.0))
121 FORMAT (1X,15A5/(1X,2I6,3X,A10,10X,A3,3X,A10,15X,F5.0,5X,F5.0))
122 FORMAT (1X/(1X,I3,3X,A10,4X,2F11.0))
123 FORMAT (////1X,15A5/(10X,F8.3))
124 FORMAT (////1X,15A5/(1X,15I2))
125 FORMAT (1X,I3,1X,A10,1X,15F7.0/(16X,15F7.0))
126 FORMAT (////1X,15A5/(1X,I3,3X,A10,4X,2F10.3))
127 FORMAT (////1X,15A5/(1X,I3,3X,A10,4X,F10.0))
128 FORMAT (////1X,15A5/(10X,F8.3))
129 FORMAT (////1X,15A5/(1X,14,6X,4F8.3))
130 FORMAT (////1X,15A5//)
131 FORMAT (////1X,15A5/(10X,5I7))
132 FORMAT (////1X,15A5/(1X,20I2))
133 FORMAT (////1X,15A5/(10X,4F8.3))
134 FORMAT (//1X,40H***** END OF BASE DATA ***** )
135 FORMAT (1X////,31H***** STATE DATA ***** ,////)
136 FORMAT (1X,15A5/(1X,I3,2X,A10,F5.0))
137 FORMAT (1X,(1X,I3,2X,A10,8F5.0))
138 FORMAT (//1X,40H***END OF STATE AND GRADE B BASE DATA***)
139 FORMAT (2X,18I4)
140 FORMAT (1X/(10X,4F11.0))
141 FORMAT (1X/(10X,5F8.0))
142 FORMAT (1X/(10X,5F8.3))
143 FORMAT (1X/(10X,F8.2))
```

144 CONTINUE

...DETERMINE INTERNAL INDEX NUMBERS FOR MERGED ORDERS

IF (IMR.LE.0) GO TO 150

```
N1=0
N2=0
ISMATCH=0
```

```
DO 149 I=1,IMR
  N1=N2+1
  N2=N2+IMRN(I)
DO 149 J=N1,N2
  IF (ISWTC.H.NE.0) GO TO 145
  ISWTC=1
  GO TO 147
145 WRITE (OP3,146) IMRG(J-1),IMR
146 FORMAT (///5X,25H*** THE FOLLOWING NUMBER ,I3,35H IS NOT A VALID O
  RDER NUMBER. FOUND,11H IN MERGER ,I2,19H SPECIFICATION. ***/)
  STOP
147 DO 148 K=1,NSA1
  IF (NOR(K).NE.IMRG(J)) GO TO 148
  ISWTC=0
  IMRG(J)=K
  GO TO 149
148 CONTINUE
149 CONTINUE
150 IF (MPDC.EQ.1.AND.RMR.NE.1) WRITE (OP3,151)
  IF (MPDC.EQ.1.AND.PMR.NE.1) WRITE (OP3,152)
151 FORMAT (//1X,42HMAXIMUM QUARTERLY DECLINE IN SALES TO OWN ,/1X,51H
  1AREA BY PRODUCERS FIXED AT LEVELS SPECIFIED IN BASE,/1X,17H DATA S
  2SECTION 24.)
152 FORMAT (//1X,48HMAXIMUM QUARTERLY DECLINE IN PACKAGED MILK SALES,/
  11X,53H TO OWN AREA BY PROCESSORS FIXED AT LEVELS SPECIFIED ,/1X,25
  2H IN BASE DATA SECTION 28.)
  I=IC1EF+IC2EF+IPCEF+IGBEF+IBTEF+IPTEF+IHCEF+IPDEF
  IF (I.GT.0.AND.OPEXF.NE.1) WRITE (OP3,153)
153 FORMAT (1X,50H***** ONE OR MORE INDIVIDUAL EXOGENOUS *****/
  11X,51H***** FACTORS SELECTED TO VARY FROM BASE DATA **)

...SET EXOGENOUS FACTORS OPTIONS
  IF (OPEXF.NE.1) GO TO 154

  IC1EF=1
  IC2EF=1
  IPCEF=1
  IGBEF=1
  IBTEF=1
  IPTEF=1
  IHCEF=1
  IPDEF=1
154 CONTINUE
  DO 155 I=1,5
  IF (IHCEF.NE.1) CHGHC(I)=1.0
  IF (IBTEF.NE.1) CHGBT(I)=1.0
  IF (IPCEF.NE.1) CHGPC(I)=1.0
  IF (IPTEF.NE.1) CHGPT(I)=1.0
  CHGBT(I)=CHGBT(I)
155 CONTINUE
  IF (ISUP7.EQ.1.OR.ISUP7.EQ.3) NCL2P=OCL2P(OCNT)
  IF (ISUP7.EQ.1.OR.ISUP7.EQ.3) NCL3P=OCL3P(OCNT)
  IF (ISUP7.NE.1.AND.ISUP7.NE.3.AND.NCL2P.EQ.0.0) NCL2P=BC2P(5)
  XCL2P=CEMGN*(OCL2P(5)+OCST2)
  XBP=NCL2P
  IF (ISUP7.NE.1.AND.ISUP7.NE.3.AND.NCL3P.EQ.0.0) NCL3P=BC3P(4)
  XBP3=NCL3P
  IF (PSE2.EQ.0.0) PSE2=SE2
  DO 156 I=1,NTS
```

NRPR(I)=A(I,5)
CPRCHG(I)=0.00

...INCREASE CLASS I CAPACITY BY AMOUNT OF RESERVE REQUIREMENT

SCL1CP=SCL1CP+CL1CP(I)
CL1CP(I)=(1.+CL1RR(I))*CL1CP(I)
ISC1CP=ISC1CP+IFIX(CL1CP(I)+0.5)
IF (I.GT.NSA1) GO TO 156
APRREC(I,1)=0.
APCL1(I,1)=0.
AIACIS(I,1)=0.
ACL37(I,1)=0.
ACL20(I,1)=0.
IF (ISUP1.EQ.1) CHGCL1(I)=APMM(I,1)*0.01
IF (IPDEF.NE.1.AND.I.LT.21) CHGPD(I)=1.0

156 CONTINUE
IF (ISUP1.EQ.1) CHGCL4=CL2*0.01
IF (ISUP1.EQ.1) CHGCL5=CL3*0.01
JK=2*NT1

DO 157 I=1,JK
157 WORK1(I)=0.0
DO 158 I=2,5
BNCL2P(I)=BC2P(I)
BNCL3P(I)=BC3P(I-1)
MIACL1(I-1)=0.
MTPSQ(I)=0.
MCL10(I)=0.
MCL30(I)=0.
MCL20(I)=0.
BMRTLPI(I)=0.
BSTRNS(I)=0.
BSDAC1(I)=0.
SERCST(I)=0.
SRMPC(I)=0.
BMCLIP(I)=0.

158 CONTINUE

...
DO 116 I=1,NT9
DO 114 JK=1,NSA1
114 APMM(I,JK)=0.0
IF (I.LE.NSA1) GO TO 1115
DO 2115 J=1,4
2115 CL10(I,J)=IACL1(I,J)
1115 IF (ISUP7.GE.1) NCL1P(I)=QCL1P(I,QCNT)
IF (ISUP7.LT.1.AND.NCL1P(I).EQ.0.0) NCL1P(I)=QMC1P(I)
CL1P(I,1)=NCL1P(I)

116 CONTINUE

DO 160 I=1,NSA1
DO 159 JK=1,NT9
APMM(JK,I)=0.0
159 CONTINUE
IF (ISUP7.GE.1) NCL1P(I)=QCL1P(I,QCNT)
IF (ISUP7.LT.1.AND.NCL1P(I).EQ.0.0) NCL1P(I)=QMC1P(I)
CL1P(I,1)=NCL1P(I)
160 CONTINUE
IF (ISTATE.LE.0) GO TO 163
DO 162 I=NT5,NT3

```

IF (ISTT.GT.1) NCLIP(I)=QCLIP(I,QCNT)
IF (ISTT.LE.1.AND.NCLIP(I).EQ.0.0) NCLIP(I)=OMC1P(I+NSP)
CLIP(I,1)=NCLIP(I)
DO 161 J=1,4
  CL10(I,J)=IACL1(I,J)
161 CONTINUE
162 CONTINUE
163 CONTINUE

```

...OPTION TO USE IN AREA CLASS 1 SALES AS A DEMAND BASE

```

DO 165 I=1,NSA1
IF (OPCL1.EQ.1) GO TO 165
DO 164 J=1,4
  IACL1(I,J)=CL10(I,J)
164 CONTINUE
165 CONTINUE

```

...CALCULATIONS REQUIRED FOR GRADE A REPORT WRITER FROM BASE DATA

```

DO 166 N=1,10
  IDR(N)=ICR(N)+ICRS(N)
166 CONTINUE
DO 168 I=1,NT9
  DO 167 J=2,5
    APMM(I,J-1)=(CL10(I,J-1)/TPSQ(I,J-1))*100.0
    APMM(I,5)=APMM(I,5)+APMM(I,J-1)/4.0
    XY=CL1CP(I)/(1.0+CL1RR(I))
    APMM(I,J+4)=(CL10(I,J-1)/XY)*100.0
    APMM(I,10)=APMM(I,10)+APMM(I,J+4)/4.0
    APMM(I,J+9)=(A(I,J)*23.2555)/RMGN
    APMM(I,15)=APMM(I,15)+APMM(I,J+9)/4.0
    APMM(I,16)=APMM(I,16)+CL1P(I,J)/4.0
    APMM(I,17)=APMM(I,17)+BLEND(I,J)/4.0
    XCL3T=TPSQ(I,J-1)-(CL10(I,J-1)+CL20(I,J-1))
    APMM(I,J+10)=XCL3T
    ACL20(I,1)=ACL20(I,1)+CL20(I,J-1)
    ACL30(I,1)=ACL30(I,1)+XCL3T
    APRREC(I,1)=APRREC(I,1)+TPSQ(I,J-1)
    APCL1(I,1)=APCL1(I,1)+CL10(I,J-1)
    AIAC15(I,1)=AIAC15(I,1)+IACL1(I,J-1)
    BTPSQ(I,J-1)=TPSQ(I,J-1)
    BIAC1(I,J-1)=IACL1(I,J-1)
    NCL20(I,J-1)=CL20(I,J-1)
    IF (I.GT.NSA1) GO TO 167
    BMRTL(J)=BMRTL(J)+A(I,J)*IACL1(I,J-1)*23.2555
    SERCST(J)=SERCST(J)+TPSQ(I,J-1)*ERCST(I)
    SRMPC(J)=SRMPC(J)+TPSQ(I,J-1)*RMPC(I)
    MCL30(J)=MCL30(J)+XCL3T
    MCL20(J)=MCL20(J)+CL20(I,J-1)
    MIACL1(J-1)=MIACL1(J-1)+IACL1(I,J-1)
    MTPSQ(J)=MTPSQ(J)+TPSQ(I,J-1)
    NCL10(J)=NCL10(J)+CL10(I,J-1)
    BMCLIP(J)=BMCLIP(J)+CLIP(I,J)+CL10(I,J-1)
  167 CONTINUE
  IF (I.GT.NSA1) GO TO 168
  ASCL20(1)=ASCL20(1)+ACL20(I,1)
  ASCL30(1)=ASCL30(1)+ACL30(I,1)
  ASPCL1(1)=ASPCL1(1)+APCL1(I,1)
  ASMRTL(1)=ASMRTL(1)+BMRTL(I,1)

```

```
ERCST(I)=ERCST(I)-ACST
ASPRRC(1)=ASPRRC(1)+APRREC(I,1)
168 CONTINUE
IF (ISTATE.LE.0) GO TO 170
DO 169 I=NT6,NT1
169 ERCST(I)=ERCST(I)-ACST

...SET VARIABLES FROM BASE DATA TO BE USED IN COMPARATIVE
REPORTS IN WORK1 ARRAY

170 IF (NIC.EQ.0) GO TO 172
DO 171 J=1,4
  JX=J+20
  CALL OSET (J+5,0,NIC,ICR,NT9,TPSQ(1,J),CL1Q(1,J),NCL2Q(1,J),APMM(1,
1,JX),APMM(1,J),CL1P(1,J+1),BLEND(1,J+1),APMM(1,J+5),APMM(1,J+10),I
2ACL1(1,J),WORK1,NT9)
171 CONTINUE
  CALL OSET (10,0,NIC,ICR,NT9,APRREC(1,1),APCL1(1,1),ACL2D(1,1),ACL3
1Q(1,1),APMM(1,5),APMM(1,16),APMM(1,17),APMM(1,10),APMM(1,15),AIAC1
2S(1,1),WORK1,NT9)
172 CONTINUE
DO 173 I=1,NMF
173 SMMFC=SMMFC+MMFC(I)
DO 174 I=1,NSMF
174 SSMFC=SSMFC+SMFC(I)
  TSMFC=SSMFC+SMMFC
  DO 175 I=2,5
    NI=(I-2)*150
    BMCL1U(I)=(MCL1Q(I)/MTPSQ(I))*100.0
    BMCL1P(I)=BMCL1P(I)/MCL1Q(I)
    BSUP(I)=MCL1Q(I)*BMCL1P(I)+MCL2Q(I)*BNCL2P(I)+MCL3Q(I)*BNCL3P(I
1 )
    BMBLEN(I)=BSUP(I)/MTPSQ(I)
    BSNFP(I)=BMBLEN(I)-SERCST(I)/MTPSQ(I)-PMCHG
    BSUROD(I)=BSNFP(I)-SRMFC(I)/MTPSQ(I)
    BMRTL(I)=BMRTL(I)/(MIACL1(I-1)*23.2558)
    BWFRPS(I)=BMRTL(I)-BMCL1P(I)/23.2558
    BCEXP(I)=BMRTL(I)*MIACL1(I-1)*23.2558
    BSMUCP(I)=(MCL1Q(I)/SCL1CP)*100.0
    BSUFRP(I)=BWFRPS(I)*MIACL1(I-1)*23.2558
    BSMFCU(I)=(MCL3Q(I)/TSMFC)*100.0
    AMC1U(1)=AMC1U(1)+BMCL1U(I)/4.
    AMC1P(1)=AMC1P(1)+BMCL1P(I)/4.
    AMC2P(1)=AMC2P(1)+BNCL2P(I)/4.
    AMC3P(1)=AMC3P(1)+BNCL3P(I)/4.
    AMBLND(1)=AMBLND(1)+BMBLEN(I)/4.
    AUP(1)=AUP(1)+BSUP(I)
    ASNFP(1)=ASNFP(1)+BSNFP(I)/4.
    ASRODC(1)=ASRODC(1)+BSUROD(I)/4.
    ASFRPS(1)=ASFRPS(1)+BSUFRP(I)
    ACEXP(1)=ACEXP(1)+BCEXP(I)
    AMRTL(1)=AMRTL(1)+BMRTL(I)/4.
    AWFRPS(1)=AWFRPS(1)+BWFRPS(I)/4.
    AMUCP(1)=AMUCP(1)+BSMUCP(I)/4.
    ASMFCU(1)=ASMFCU(1)+BSMFCU(I)/4.

...WRITE BASE DATA TO BE USED BY REPORT WRITER FOR QUARTERLY
COMPARATIVE SUMMARY REPORTS - DAMFCLC

WRITE (OP2) MTPSQ(I),MIACL1(I-1),MCL2Q(I),MCL3Q(I),BMCL1U(I),BM
```



```
1  CL1P(I),BNCL2P(I),BNCL3P(I),EMLEN(I),BSUP(I),BSNFP(I),BSUROD(I
2  ),BSUFRP(I),BCEXP(I),BNRTL(I),BWRFRPS(I),BSMUCP(I),BSMFGU(I),BS
3  TRNS(I),SFPCST,DTCST,UFPCST,BSOAC1(I),TMCST,TRMPC

...WRITE BASE DATA FOR GRADE B REPORT WRITER - DRW2
      IF (IMFGB.NE.0) WRITE (OP8) EMBLEN(I),BNCL3P(I),BSUP(I),BSUROD(
1  I),BCEXP(I)

...STORE DATA IN WORK1 ARRAY
      REWIND OP2
      READ (OP2) (WORK1(N1+IJK),IJK=1,25)
      REWIND OP2
175  CONTINUE

...BASE YEAR DATA (ANNUAL) FOR COMPARATIVE SUMMARY REPORT - DAMPLC
      WRITE (OP2) ASPRRD(1),ASFCL1(1),ASCL2D(1),ASCL3Q(1),AMC1U(1),AMC1P
1(1),AMC2P(1),AMC3P(1),AMBLND(1),AUP(1),ASNFP(1),ASRODC(1),ASFRPS(1
2),ACEXP(1),AMRTL(1),AMFRPS(1),AMUCP(1),ASMFGU(1),ASTRNS(1),ASFPCS,
3ADTCST,ADFFCS,ASOAC1(1),ATMCST,ATRMPC

...CALCULATE UNREG. VARIABLES AND MFG. PRODUCTION
AND WRITE TO REPORT WRITERS - DRW1 AND DRW2
      IF (ISTATE.EQ.0) GO TO 180

      DO 179 K=1,4
      KK=K+1
      UNCEXP=0.0
      DO 178 I=1,MS
      UNC1P=0.0
      UNNRP=0.0
      UNNBP=0.0
      NIXB=0.0
      APMM(I,K)=0.0
      APMM(I+9,K)=0.0
      APMM(I+18,K)=0.0
      DO 176 J=1,12
      IXB=IDX(I,J)
      IF (IXB.EQ.0) GO TO 177
      UNC1P=UNC1P+CL1P(IXB,KK)
      UNNRP=UNNRP+NRPR(IXB)
      NIXB=NIXB+1
      APMM(I,K)=APMM(I,K)+TFSQ(IXB,K)
      APMM(I+9,K)=APMM(I+9,K)+CL1Q(IXB,K)
      IF (IGBEF.EQ.1) APMM(I+18,K)=APMM(I+18,K)-GRBCNU(IXB)*GRB
1  IND(K)
176  CONTINUE
177  UNC1P=UNC1P/FLOAT(NIXB)
      UNNRP=UNNRP/FLOAT(NIXB)
      UNCBP(I,K)=UNC1P*UNUT+EC3P(K)*(1.0-UNUT)
      UNNBP=UNCBP(I,K)-UNFPHC-UNPMC
      UNREG=UNB(K)*UNRS(I)
      UNC1S=UNREG*UNUT
      UNCEXP=UNCEXP+UNNRP*UNC1S
      UNUP=UNCBP(I,K)*UNREG
      UNRODC=UNNBP-UNFC
      APMM(I+18,K)=APMM(I+18,K)+APMM(I,K)-APMM(I+9,K)+B(I,K)+UNREG
```

```
1      -UNC1S
      WRITE (OP1) UNREG,UNC1S,UNGBP(I,K),UNNBP,UNRODC
      WRITE (OP8) UNREG,UNC1S,UNGBP(I,K),UNUP,UNRODC
178    CONTINUE
      WRITE (OP8) UNCEXP
179    CONTINUE
180    WRITE (OP8) TITLE,NYBASE,NY,NQ,QRB,ICRB,((APMM(I,J),I=1,MS),J=1,4)
      1,((APMM(II,JJ),II=10,18),JJ=1,4),B,((APMM(K,L),K=19,27),L=1,4),BIM
      2P,BS,C,((RPB(I,J),I=1,5),J=2,5),BPC,BMKP,CF,BFPHC,BPMC,BOPC,CHGPD,
      ZIPDEF,CHGHC
...STORE ANNUAL DATA IN WORK1 ARRAY
      REWIND OP2
      N1=N1+150
      READ (OP2) (WORK1(N1+IJK),IJK=1,25)
      REWIND OP2
...DETERMINE CLASS1 PRICE AT SUPPLY PLANTS
      DO 181 I=1,NSP
        CLISP(I)=NCLIP(I+NW)
181    CLIP(I,2)=CLISP(I)
...CALCULATE BASE INTERCEPT VALUES FOR DEMAND FUNCTIONS
      DO 182 I=1,NT9
        DO 182 J=1,4
          IF (DE(I).EQ.0.0) A(I,J)=IACL1(I,J)
182    IF (DE(I).NE.0.0) A(I,J)=IACL1(I,J)/(A(I,J)*DE(I))
...CALCULATE BASE CLASS 2 RETAIL PRICES
      DO 183 I=1,5
        BCC3P(I)=BC3P(I)
183    BC2P(I)=C2MGN*(BC2P(I)+OCST2)
...CALCULATE BASE UNIT TRANSPORTATION COSTS
      K=0
      DO 185 I=1,NT
        DO 184 J=1,NSA1
          IF (I.LE.NMMF.AND.J.LE.MXM) DMMFG(I,J)=TRANSM(I,J)
          IF (I.LE.NSA1) PDMI(I,J)=TRANSP(I,J,K)
184    SPMI(I,J)=TRANSP(I,J,K)
185    IF (I.LE.NDM) MMDM(I)=TRANSD(I)
...CALCULATE BASE UNIT TRANSPORTATION COSTS: STATES
      IF (ISTATE.LE.0) GO TO 190
      DO 189 K=1,7,2
        DO 189 I=1,NSA1S
          DO 188 J=1,NT11
            DO 187 K1=1,NSA1
              IF (K.GE.5) GO TO 186
              IF (IFIX(SPMI(I,J,K)).NE.K1) GO TO 187
```

```
SSPMI(I,J,K+1)=TRANSP(I,J,K)
186 DO 188 K=1,NTG
187     IF (SSPMI(I,J,K)).NE.K1) GO TO 187
188     SSPMI(I,J,K+1)=TRANSP(I,J,K)
189 CONTINUE
190 CONTINUE

...WRITE TRANSPORTATION COST MATRIX FOR USE IN SUBROUTINE TCOST

TEMP=-1.0
TEMP1=0.0
K=1
DO 200 J=1,NTG
    IF (ISTATE.EQ.0) GO TO 199
    IF (J.GT.NSA1) GO TO 193

...STATE SUPPLY TO F.O. PROCESSOR

K=1
DO 192 M1=1,NSA1S
    IST=0
    DO 191 M2=1,NT11
        IF (IFIX(SSPMI(M1,M2,K)).NE.J) GO TO 191
        IST=1
        SPACE2(M1)=SSPMI(M1,M2,K+1)
        GO TO 192
191     CONTINUE
        IF (IST.EQ.0) SPACE2(M1)=TEMP
192     CONTINUE
        WRITE (OP7) (SPMI(I,J),I=1,NSA1),(SPACE2(M),M=1,NSA1S)
        GO TO 200

...STATE PROCESSOR TO F.O. SUPPLY

193     K=3
        JJ=J-NSA1
        DO 195 M1=1,NSA1
            IST=0
            DO 194 M2=1,NT11
                IF (IFIX(SSPMI(JJ,M2,K)).NE.M1) GO TO 194
                IST=1
                SPACE2(M1)=SSPMI(JJ,M2,K+1)
                GO TO 195
194         CONTINUE
                IF (IST.EQ.0) SPACE2(M1)=TEMP
195         CONTINUE

...STATE PROCESSOR TO STATE SUPPLY

JM1=JJ+NT
DO 198 M1=NTG,NT1
    IF (M1.EQ.JM1) GO TO 197
    IST=0
    DO 199 M2=1,NT11
        IF (IFIX(SSPMI(JJ,M2,K)).NE.M1) GO TO 196
        IST=1
        SPACE2(M1-NSP)=SSPMI(JJ,M2,K+1)
```

```
          GO TO 198
196      CONTINUE
          IF (IST.EQ.0) SPACE2(M1-NSP)=TEMP
          GO TO 198
197      SPACE2(J)=TEMP1
198      CONTINUE
          WRITE (OP7) (SPACE2(I), I=1,NT9)
          GO TO 200
```

...FEDERAL ORDER ONLY

```
199      WRITE (OP7) (SPMI(I,J), I=1,NSA1)
200      CONTINUE
```

...WRITE OUT BLOCK COMMON FOR SOLVER - DAMPSLU

```
REWIND OP5
WRITE (OP5) (WORK2(I), I=1, IWORK2), (WORK4(J), J=1, IWORK4), (WORK5(K),
1K=1, IWORK5), (WORK6(L), L=1, IWORK6), WORK7, IMFGS, (WORKS(M), M=1, IWORK8
2), (WORK9(N), N=1, IWORK9), (WORK10(II), II=1, IWORK0), (WORK11(JJ), JJ=1,
3IWORK7), IMRG, BLDIF, CHGCL1, QCL1P, QCL2P, QCL3P, CL1P, TITLE, (WORK12(KK)
4, KK=1, IWORKA)
```

...WRITE BLANK COMMON FOR COMP. REPORTS IN REPORT WRITER - DRW1

```
REWIND OP2
WRITE (OP2) WORK1, WORK4, WORK5, WORK6, NYCNT, WORK8, WORK9, WORK10, WORK1
13, WORK11
```

STOP

```
END
SUBROUTINE OSET (QT, NYR, NIC, ICR, NSA1, PPREC, PCL1, PCL2, TCL3, CL1UT, NC
1L1P, BLEND, UTCP, TCCWT, CQC1D, WORK1, NT9)
```

...SUBROUTINE OSET

...COMPARATIVE REPORT DATA PLACED IN WORK1 ARRAY

```
DIMENSION ICR(10), NO(5), PPREC(1), PCL1(1), PCL2(1), TCL3(1), CL1
1UT(1), NCL1P(1), BLEND(1), UTCP(1), TCCWT(1), CQC1D(1), WORK1(1)
```

```
REAL NCL1P
INTEGER QT
```

```
.....N1, NO(1), NO(2), NO(3), NO(4), NO(5)/1, 751, 2101, 3451,
4801, 6151/
```

```
N=750+(NYR*NT9)+((QT-6)*(6*NT9))
```

```
DO 1 I=1, NIC
```

```
1 NO(I)=N*(I-1)*(30*NT9)
```

```
KJ=0
```

```
DO 13 M=1, 10
```

```
IF (ICR(M).EQ.0) GO TO 13
```

```
KJ=KJ+1
```

```
DO 12 I=1, NSA1
```

```
NN=NO(KJ)+I
```

```
GO TO (2, 3, 4, 5, 6, 7, 8, 9, 10, 11), M
```

```
2 WORK1(NN)=PPREC(I)
```

```
3      GO TO 12
      WORK1(NN)=PCL1(I)
      GO TO 12
4      WORK1(NN)=PCL2(I)
      GO TO 12
5      WORK1(NN)=TCL3(I)
      GO TO 12
6      WORK1(NN)=CLIUT(I)
      GO TO 12
7      WORK1(NN)=NCL1P(I)
      GO TO 12
8      WORK1(NN)=BLEND(I)
      GO TO 12
9      WORK1(NN)=UTCP(I)
      GO TO 12
10     WORK1(NN)=TCCWT(I)
      GO TO 12
11     WORK1(NN)=COC1D(I)
12     CONTINUE
13     CONTINUE
      RETURN

      END
      FUNCTION GCL1P(BP,DIFF,DIST)

...GENERATE CLASS I PRICES USING BASE PRICE AND LOCATION DIFFERENTIAL
A LOCATION DIFFERENTIAL

      GCL1P=BP+DIFF*DIST
      RETURN

      END
      FUNCTION TRANSB(I,J,K)

...TRANSPORTATION COST TO PROCESSING CENTERS:

      COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SPLB(16,4),DISTB(7
25),DISTP(59),NOR(75),ONAME(75),QSPR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BP1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
42(59),CHGNC1(59),IC1EF,BC2P(5),CL20(59,4),PSE2,IC2EF,BIAC1(59,4),D
51STD,DCHG1,SSPMI(14,8,8)
      COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIGI,IMR,IMRN(11
1)
      COMMON /BLK8/ NYCNT
      COMMON /BLK9/ NW,NSN,NT,NFN,NNN,NMN,NMM,NMD,NDP,NMS,NTN

      REAL MPDRM,MPDPM,IPPM,MMDM

      TRACESUBSCRIPTS

      TRANSB=-1.0
      IF (K.GT.0) GO TO 2
      IF (SPMI(I,J).GT.DISTB(I).OR.SPMI(I,J).LT.0.0) GO TO 1
      TRANSB=AR+BR*SPMI(I,J)
      IF (SPMI(I,J).GT.DISTD.AND.I.LE.NSX1) TRANSB=TRANSB+DCHG1
1 IF (I.EQ.J) TRANSB=0.
      IF (I.GT.NSX1.AND.I.EQ.(J+NSPX)) TRANSB=0.
      GO TO 3
2 CONTINUE
```

```

IF ((SSPMI(I,J,K+1).GT.DISTB(I+NT)).OR.(SSPMI(I,J,K).LT.0.0)) GO T
10 3
TRANSB=AR+BR*SSPMI(I,J,K+1)
IF (SSPMI(I,J,K+1).GT.DISTD) TRANSB=TRANSB+DCHG1
IF (SSPMI(I,J,K+1).EQ.0.) TRANSB=0.
3 CONTINUE
RETURN

END
FUNCTION TRANSM(I,J)

```

...TRANSPORTATION COST MANUFACTURING CENTERS

```

COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SPLB(16,4),DISTB(7
25),DISTP(59),NDR(75),ONAME(75),QSPR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BP1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
42(59),CHGNC1(59),IC1EF,BC2P(5),CL2Q(59,4),PSE2,IC2EF,BIAC1(59,4),D
SISTD,DCHG1,SSPMI(14,8,8)
COMMON /BLK8/ NYCNT

REAL MPDRM,MPDPM,IPPM,MMDM

TRANSM=AR+BR*DMMFG(I,J)
IF (MMNC(I,J).EQ.0) TRANSM=0.
RETURN

END
FUNCTION TRANSD(I)

```

...TRANSPORTATION COST MULTIPLE MFG. DUMMY

```

COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SPLB(16,4),DISTB(7
25),DISTP(59),NDR(75),ONAME(75),QSPR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BP1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
42(59),CHGNC1(59),IC1EF,BC2P(5),CL2Q(59,4),PSE2,IC2EF,BIAC1(59,4),D
SISTD,DCHG1,SSPMI(14,8,8)
COMMON /BLK8/ NYCNT

REAL MPDRM,MPDPM,IPPM,MMDM

TRANSD=AR+BR*MMDM(I)
RETURN

END
FUNCTION TRANSP(I,J,K)

```

...TRANSPORTATION COST PACKAGED MILK

```

COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SPLB(16,4),DISTB(7
25),DISTP(59),NDR(75),ONAME(75),QSPR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BP1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
42(59),CHGNC1(59),IC1EF,BC2P(5),CL2Q(59,4),PSE2,IC2EF,BIAC1(59,4),D
SISTD,DCHG1,SSPMI(14,8,8)
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIG1,IMR,IMRN(11
1)
COMMON /BLK8/ NYCNT
COMMON /BLK9/ NW,NSN,NT,NPN,NNN,NYN,NMM,NMD,NDR,NMS,NTN

```

```
REAL MPDRM,MPDPM,IPPM,MMDM
TRANSP=-1.0
IF (K.GT.0) GO TO 2
IF (PDMI(I,J).GT.DISTP(I).OR.PDMI(I,J).LT.0.0) GO TO 1
TRANSP=AP+BP1*PDMI(I,J)
1 IF (I.EQ.J) TRANSP=0.
GO TO 3
2 CONTINUE
IF ((SSPMI(I,J,K+1).GT.DISTP(I+NSA1)).OR.(SSPMI(I,J,K).LT.0.0)) GO
1 TO 3
TRANSP=AP+BP1*SSPMI(I,J,K+1)
IF (SSPMI(I,J,K+1).EQ.0.) TRANSP=0.
3 CONTINUE
RETURN

END
FUNCTION CTIME(T1)
...CTIME ROUTINE FOR 6000, 7000 SERIES CDC
CALL SECOND (T1)
CTIME=-T1
RETURN

END
```

DAMPSLV

DAMPSLV, listed on the following pages, includes ten subroutines---NETGEN, PREP, NETGENB, NARCW, FIXUPA, COMP, MRGE, RESET, RESETB, and GNETA; and seven functions---NTPSQ, CL1D, CL2D, GRADEB, RIMPC, CMMF, and UNREG. GNETA is based on GNET, an algorithm designed to solve large scale transshipment networks. GNET is copywrited by Gordon H. Bradley and Gerald G. Brown; for further details, contact Bradley or Brown at the Department of Operations Research, Naval Postgraduate School, Monterey, California, 93940.

Program flow is illustrated in Figure 2. Once all the necessary data is read from DAMPSIN, the main program calls NETGEN. NETGEN sets up the stage one network. GNETA is called by NETGEN to solve the network. GNETA1 and GNETA2 are entry points in GNETA. GNETA1 sets up the costs and restrictions on arcs. GNETA2 solves the network, once it is specified. RESET and RESET1, an entry point in RESET, are called to assemble the solution results.

Program control returns to the main program, which then calls COMP. COMP takes results from NETGEN and RESET and prepares results for DAMPCLC and DRW2 and also reinitializes variables for the next quarterly run. MRGE is called from COMP to calculate blend prices for each regulated market. If the Merge Option is invoked, MRGE also makes the necessary merger calculatons.

If the Grade B option is invoked, PREP is called from COMP. (If PREP is not called, control returns to the main program and the cycle through NETGEN and COMP is repeated until all quarters are run.) PREP controls the Grade B part of the model; model parameters are set and supplies and demands are calculated. PREP calls NETGENB to set up the stage two network. NETGENB calls GNETA to solve the network, and RESETB is called from GNETA to assemble the results. In the first quarter, NETGENB also calls NARCW, which writes network parameters to the output file. Control returns to PREP, which then calls FIXUPA. FIXUPA calculates ending cheese stocks and butter and nonfat dry milk production for stage three. Control returns again to PREP, whereupon NETGENC, an entry point in NETGENB, is called. NETGENC sets up the stage three network and calls GNETA to solve it. GNETA calls RESETB to assemble the results. Control returns to PREP, whereupon FIXUPB is called. FIXUPB, and entry point in FIXUPA, calculates ending stocks for stage three. Control returns to PREP and then to COMP. The procedure is repeated for as many quarters as are requested.

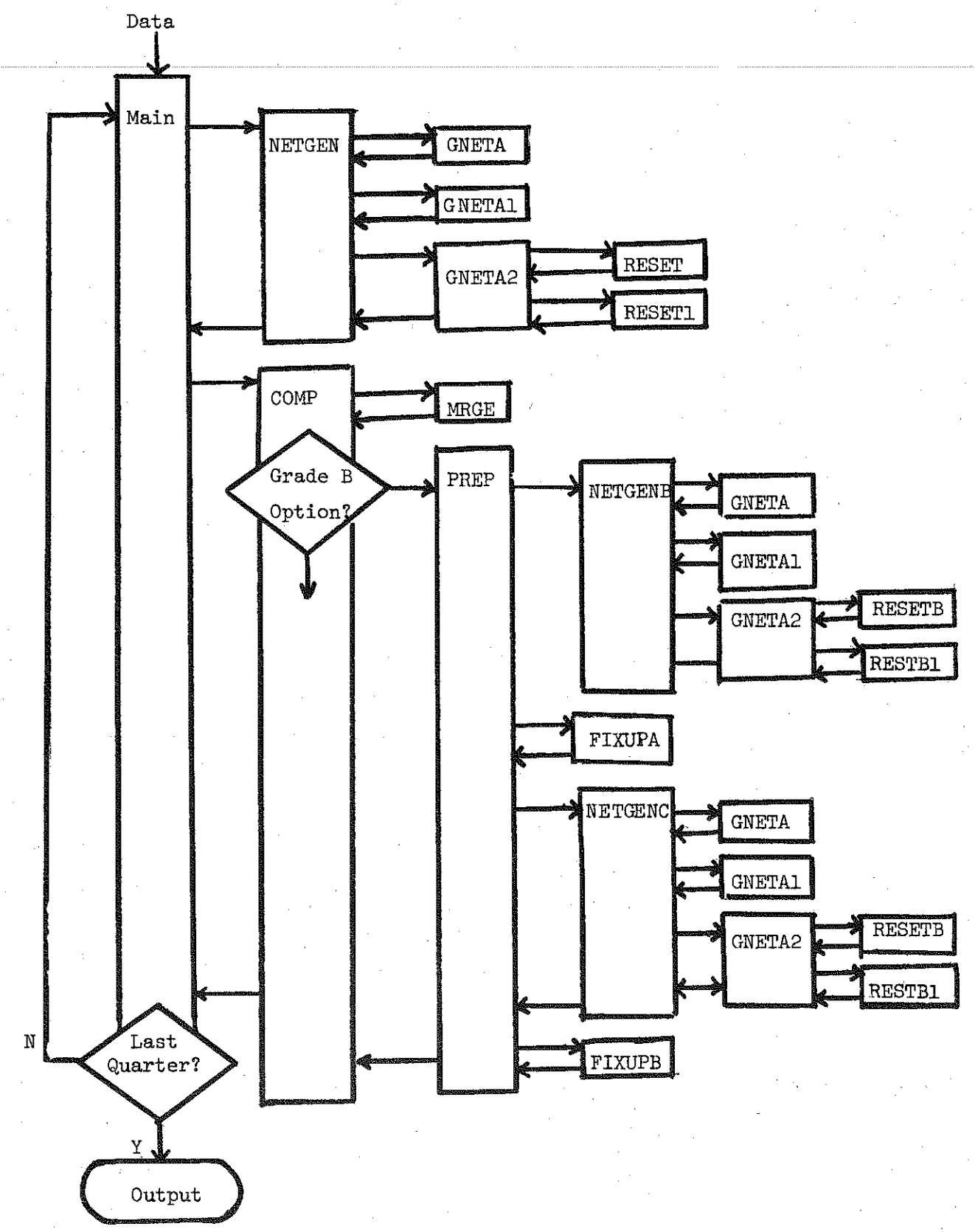


Figure 2. Basic Program Flow in DAMPSLV.

PROGRAM DAMPSLU (INPUT=130,OUTPUT=130,TAPES=OUTPUT,TAPES=INPUT,TAP
1E10=130,TAPE13=130,TAPE42=130,TAPE16=130)

.... DAMPSLU IS THE SECOND PROGRAM OF DAMPS
DAMPSLU SETS UP AND SOLVES THE TRANSHIPMENT PROBLEM
AND PASSES DATA TO DAMPCLC AND THE REPORT WRITERS -
DRW1 AND DRW2. DAMPSLU WAS WRITTEN BY D.R. MARTELLA,
A.M. NOVAKOVIC, AND J.E. PRATT.

DIMENSION WORK2(11619)
DIMENSION WORK4(24), WORK5(890), WORK6(45), WORK8(11)
DIMENSION WORK9(407), WORK10(852), WORK11(18)
DIMENSION WORK12(767)

WORK2 = TOTAL SIZE OF COMMON IN BLK1
WORK4 = TOTAL SIZE OF COMMON IN BLK2
WORK5 = TOTAL SIZE OF COMMON IN BLK5 LESS ICMEF
WORK6 = TOTAL SIZE OF COMMON IN BLK7
WORK8 = TOTAL SIZE OF COMMON IN BLK9
WORK9 = TOTAL SIZE OF COMMON IN BLK10
WORK10 = TOTAL SIZE OF COMMON IN BLK11
WORK11 = TOTAL SIZE OF COMMON IN BLK12
WORK12 = TOTAL SIZE OF COMMON IN BLK14

COMMON /BLK1/ SPMI(61,45), DMMFG(27,10), LMMFG(27,10), MMNC(27,10), PD
IMI(45,45), MPDRM(59), MPDPM(59), IPPM(45), NARC(15), SFLB(16,4), DISTB(7
23), DISTP(59), NOR(75), ONAME(75), QSFR(16), HCHG1(16), MMDM(19), CHGPT(5
3), CHGOT(5), AR, BR, AP, BP1, IBTEF, IPTEF, NSX1, NSPX, A(59,5), DE(59), CHGNC
42(59), CHGNC1(59), IC1EF, DC2P(5), CL20(59,4), PSE2, IC2EF, BIAC1(59,4), D
SISTD, DCHG1, SSPMI(14,8,8), SPACES(3594)
COMMON /BLK2/ REA(5), NSP, NSA1, NSMF, NDM, NMMF, NMFGT, BIGI, IMR, IMRN(11
1)
COMMON /BLK5/ CPRCHG(59), BTPSQ(59,4), BLEND(59,6), SE(59), GRBINE(4),
1GRBCNU(59), IGDEF, COTSP(59), QSPL1(59), ICMEF
COMMON /BLK7/ NAD, NOD, MXC, NNODES, IPRI(20), IPRO(20), QCNT1, OP4, KGNET
COMMON /BLK8/ NYCNT, IMFCB, OP2, OP3, OP5, OPE, OP7
COMMON /BLK9/ NW, NSN, NT, NFN, NNN, NMM, NMM, NMD, NDP, NMS, NTN
COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM
EXC, NNOD, NNNE, NNND, ISTPS, NYRBC, NYBASE, QT, QCNT, NQ, NIC, ISUP1, ISUP7, NS
SPM, NSC, NSK, IHDEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL2, IEC1CP, CHGCL3, ACST,
4PMCHG, RMGN, OCST2, C2MGN, RMR, PCR, PMR, NCR, OPCL1, LBCHK, IPNARC, SCL1CP, T
5SMFC, XBP, XBP3, MPDC, TIME2, TIME3, TIME7
COMMON /BLK11/ NCL1P(59), CL1RR(59), IWORK2, START4, OCST, ILI(45), CHGF
1RC(5), TRANSR(59), OACLIT(59), CENSP(59), PNCST(59), PRECST(59), ICST(5,1
20), HCHG2(16), MXN, IPDEF, CQC2D(59), CQC1D(59), NRFR(59), XCL2P, DTCST, IW
3DRK3, SFPCST, DPFCST, TRMPC, TMCST, RMFC(75), EFRR(59)
COMMON /BLK12/ ISTATE, MXP, NSA19, NSMFS, NT1, NT2, NT3, NT4, NPN1, NDP1, NT
1N1, NT5, NT6, NT7, NT8, NT9, NT10, NT11
COMMON /BLK13/ IMRG(59), BLDIP(59), CHGCL1(59), QCL1P(59,20), QCL2P(20
1), QCL3P(20), CL1P(59,5), TITLE, SPOPS(16), GRC1(59), PCL1(59), TCL3(59),
2PREEC(59), UP(59), UPMRG(20), PRRECM(20), BLNDMR(20), UPAMRG(10), PRECAM
3(10), BLAMR(20)
COMMON /BLK14/ IDXB(9,12), UNRS(9), B(9,4), BIMP(4,2,4), C(5,5,4), RPD(
15,5), BS(6,9,4), BPC(9), BSE(9), DDE(5), BTRANS(11,5), EBC3P(5), BCHGI(4)
2, PCHGI(4), NRPB(5), CHGMC(5,5), SIOX(3,4), RSS(9), CF(5), BNKP(5), BROST(

35),BNF(2),BTCST(2,5),ISS,UNUT,UNPC,DCSC(20),IMPHLD,UNS(4),UNGBP(9,
44),UNFPHC,UNPMC,ECHBT(5),BFPHC

INTEGER QR,QT,QCNT,QCNT1,CR,PCR,PMR,RMR,RDA,OPEXF,OPCL1,OP1,OP2,OP
13,OP4,OP5,OP6,OP7,BIGI

REAL NFP,IACL1,MFORM,MMFC,MPDFM,IPPM,NCL2Q,MCL2Q,MCL3Q,NRPR,NCL1P,
1MFGS,MFGM,IACL1S,MTPSQ,MCL1Q,ICHG,MOCC,MOPC,MPPC,MRTLP,NCL2P,MBLEN
2Q,MCL1P,MCL1U,MIACL1,MMDM,ICP,NCL3,NCL3P,NTPSQ,MMFC

EQUIVALENCE (SPMI(1),WORK2(1))
EQUIVALENCE (RDA(1),WORK4(1)), (CPRCHG(1),WORK5(1)), (NAD,WORK6(1)
1), (NYCNT,WORK7), (NW,WORK8(1)), (MMFC(1),WORK9(1)), (NCL1P(1),WOR
2K10(1)), (ISTATE,WORK11(1)), (IDX8(1),WORK12(1))

DATA MODL/2H /

...SET PARAMETERS FOR WORK ARRAYS

IWORK2=8025
IWORK4=24
IWORK5=890
IWORK6=45
IWORK8=11
IWORK9=407
IWORK1=852
IWORK0=18
IWORK7=767

IWORK2=SIZE OF ARRAY WORK2

IP1=10
IP2=5
OP2=16
OP3=6
OP4=6
OP5=10
OP6=13
OP7=42

IP1=INPUT FROM DAMPSIN
IP2=ALL OTHER INPUT
OP2=OUTPUT TO GRADE A REPORT WRITER - DRW1
OP3=OUTPUT FROM DAMPSLU
OP4=OUTPUT FROM GNET
OP5=SCRATCH FILE FOR BLK1
OP6=SCRATCH FILE - OUTPUT TO DAMPCLC
OP7=OUTPUT TO GRADE B REPORT WRITER - DRW2

REWIND IP1
READ (IP1) (WORK2(I),I=1,IWORK2),(WORK4(J),J=1,IWORK4),(WORK5(K),K
1=1,IWORK5),(WORK6(L),L=1,IWORK6),WORK7,IMFGB,(WORK8(M),M=1,IWORK8)
2,(WORK9(N),N=1,IWORK9),(WORK10(II),II=1,IWORK1),(WORK11(JJ),JJ=1,I
3WORK0),IMRG,BLDIF,CHGCL1,QCL1P,QCL2P,QCL3P,CL1P,TITLE,(WORK12(KK),
4KK=1,IWORK7)

IDMEF=IC2EF

IWORK2=11619
WRITE (OP3,1) TITLE

```
1 FORMAT (1H1,19X,15HDECISION NAME -,1X,A6/20X,22(1H=))
...MAIN LOOP
  DO 3 IDO=1,NO
    START5=CTIME(T1)
    IF (QCNT.EQ.1) GO TO 2
    REWIND OP5
    READ (OP5) (WORK2(I),I=1,IWORK2)
2   CONTINUE
    CALL NETGEN
    TIME4=START4-CTIME(T1)
    START6=CTIME(T1)
    TIME6=START6-CTIME(T1)
    CALL COMP
    WRITE(OP3,6380) TIME1,TIME2,TIME3,TIME4,TIME5,TIME6,TIME7
3   CONTINUE
    STOP
    END
    SUBROUTINE NETGEN
...GENERATE TRANSSHIPMENT PROBLEM AS A CAPACITATED NETWORK
  DIMENSION WORK2(11619)
  COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SFLB(16,4),DISTB(7
25),DISTP(59),NOR(75),ONAME(75),QSPR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BF1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
42(59),CHGNC1(59),IC1EF,BC2P(5),CL2Q(59,4),PSE2,IC2EF,BIAC1(59,4),D
51STD,DCHG1,SSPMI(14,8,8),SPACE3(3594)
  COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFC,BIGI,IMR,IMRN(11
1)
  COMMON /BLK5/ CPRCHG(59),BTSPQ(59,4),BLEND(59,6),SE(59),GRBIND(4),
1GRBCNU(59),IGBEF,CATSP(59),QSPL1(59),ICMEF
  COMMON /BLK7/ NAD,NOD,MXC,NNODES,IPRI(20),IPRO(20),QCNT1,OP4,KNET
  COMMON /BLK8/ NYCNT,IMFCB,OP2,OP3,OP5,OP6,OP7
  COMMON /BLK9/ NW,NSN,NT,NPN,NNN,NMN,NMM,NMD,NOP,NMS,NTN
  COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
2XC,NNOD,NNNE,NNND,ISTPS,NYRDC,NYBASE,QT,QCNT,NG,NIC,ISUP1,ISUP7,NS
3FM,NSC,NSK,IHCEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISCLCP,CHGCL3,ACST,
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LBCHK,IPNARC,SCL1CP,T
5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7
  COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP
1RC(5),TRANSR(59),DACLIT(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1
20),HCHG2(16),MXN,IPCEF,CQC2D(59),CQC1D(59),NR7R(59),XCL2P,DTCST,IW
3ORK3,SFPCST,DFPCST,TRMPC,TMCST,RYMC(75),EFRR(59)
  COMMON /BLK12/ ISTATE,MXP,NSA15,NSMFS,NT1,NT2,NT3,NT4,NPN1,NDP1,NT
1N1,NT5,NT6,NT7,NT8,NT9,NT10,NT11
```

EQUIVALENCE (SPMI(1),WORK2(1))

INTEGER QR,QT,QCNT,QCNT1,CR,PCR,PMR,RMR,RDA,OPEXF,OPCL1,OP1,OP2,OP
13,OP5,OP6,OP7,OP4,BIGI

REAL NFP,IACL1,MPDRM,MMFC,MPDPM,IPPM,NCL2Q,MCL2Q,MCL3Q,NRPR,NCL1P,
1MFGS,MFGM,IACL1S,MTPSQ,MCL1Q,ICHG,MOCC,MDPC,MPPC,MRTL, NCL2P,MBLEN
2D,MCL1P,MCL1U,MIACL1,MMDM,ICP,NCL3,NCL3P,NTPSQ,MMFG

...SET MXC,NOD AS REQUIRED FOR GNET, INITIALIZE VARIABLES, ETC.

```
MXC=SMXC
NOD=NNOD
NAD=NNND
NNODES=NNNE
KNET=0
KKEY=QT-5
WRITE (OP3,63) KKEY,NYRBC
WRITE (OP3,74)
ISTPS=0
ISCL1D=0
NARCS=0
DO 1 I=1,NSP
1 QSPR(I)=1.0
DO 2 I=1,15
  NARC(I)=0
  IF (I.GT.5) GO TO 2
  RDA(I)=0
2 CONTINUE
```

...CALCULATE SUPPLY , CLASS I-II DEMAND

```
DO 3 I=1,NT9
  CQTSP(I)=NTPSQ(I,QT)
  CQC1D(I)=CL1D(I,QT)
3 CQC2D(I)=CL2D(I,QT)
```

...SET ARCS BETWEEN SUPER SOURCE AND SUPPLY NODES AND ARCS
BETWEEN SUPPLY NODES AND DUMMY SUPPLY NODES

START2=CTIME(T1)

CALL GNETA
KKEY=0

```
DO 11 I=1,NSN
  RDA(1)=NSC
  RDA(2)=1
  IF (I.GT.NT2) GO TO 6
  IF (I.GT.NT1) GO TO 7
  IF (I.GT.NT) GO TO 5
  IF (I.GT.NSA1) GO TO 4
  RDA(4)=IFIX(CQTSP(I)+0.5)
  IF (I.GT.NW) RDA(4)=IFIX((1.-SPRC(I-NW))*CQTSP(I)+0.5)
  ISTPS=ISTPS+RDA(4)
  NARC(1)=NARC(1)+1
  GO TO 10
4 RDA(4)=IFIX(SPRC(I-NSA1)*CQTSP(I-NSP)+0.5)
  ISTPS=ISTPS+RDA(4)
  NARC(1)=NARC(1)+1
  GO TO 10
```

5 IF (ISTATE.LE.0) GO TO 11
NARC(1)=NARC(1)+1

...SUPER SOURCE TO STATE SUPPLIES

RDA(4)=IFIX(COTS(I-NSP)+0.5)
ISTPS=ISTPS+RDA(4)
GO TO 10

6 IF (ISTATE.LE.0) GO TO 11

...SUPPLY NODE TO DUMMY SUPPLY - STATE

NARC(2)=NARC(2)+1
RDA(1)=I-NT3
XR=FLOAT(QCNT)*MPDM(I-NT1)
XK=COTS(I-NT1)
RDA(5)=IFIX((1.-XR)*XK+0.5)
GO TO 9

7 CONTINUE
NARC(2)=NARC(2)+1
RDA(1)=I-NT1
IF (RMR.EQ.1) GO TO 10
XS=0.0
XR=FLOAT(QCNT)*MPDM(I-NT1)
IF (MPDC.EQ.1) XR=MPDM(I-NT1)
XK=COTS(I-NT1)
IF ((I-NT1).LE.NW) GO TO 8
NP=I-NT1-NW
XS=SPRC(NP)*XK
XK=(1.-SPRC(NP))*XK
XS=SPLB(NP,QT-5)*XS
8 RDA(5)=IFIX((1.-XR)*XK+0.5)

...OPTIONAL LOWER BOUND CHECK

9 IF (LBCHK.NE.1) GO TO 10
XC=XS+FLOAT(RDA(5))
IF (XC.LE.CLICP(I-NT1).OR.PCR.EQ.1) GO TO 10
XXX=(FLOAT(RDA(5))+XS)-CLICP(I-NT1)
IF (XS.GT.0.0) QSPR(NP)=XS/XC
RDA(5)=IFIX(CLICP(I-NT1))
IF (XS.GT.0.0) RDA(5)=IFIX(CLICP(I-NT1)*(1.0-XS/XC))
JKK=I-NT1
IF (KKEY.EQ.0) WRITE (QP3,64)
KKEY=1
WRITE (QP3,65) NOR(JKK),ONAME(JKK),XXX
10 CONTINUE
CALL SNETA1
11 CONTINUE

...SET ARCS BETWEEN PROCESSING NODES AND (DIRECT SHIP) SUPPLY NODES

KKEY=0
DO 25 I=1,NT3
I1=I+NSN
I3=I+NSP
IF (I.GT.NSA1) GO TO 21
IF (I.GT.NW) IYP=IFIX(CL1SP(I-NW)+HCHG1(I-NW)+0.5)
DO 14 J=1,NSA1
IF (I.EQ.J) GO TO 14

...OMIT ARC IF DISTANCE FROM SUPPLY NODE TO PROCESSING NODE IS
GREATER THAN SPECIFIED MAXIMUM

IF (SPMI(J,I).LT.0.0) GO TO 14

...ADD INSPECTION COSTS WHERE APPLICABLE

XIC=0.0
IF (CINSP(I).LE.0.0) GO TO 13
XIC=CINSP(I)
ICI=ILI(I)
DO 12 KJ=1,MAXN
IF (J.EQ.ICST(ICI,KJ)) XIC=0.0

12
13

CONTINUE
RDA(1)=J
RDA(2)=I1
XYN=CHGHC(NYCNT)*ERCST(J)
XYM=CHGBT(NYCNT)*SPMI(J,I)
RDA(3)=IFIX((NCLIP(I)+BLEND(J,5)-BLEND(I,5)+XYM+XIC-XYN)+0.5
)

...PRICE DIRECT SHIPPED MILK FROM OTHER ORDERS AT LEAST 1 CENT
ABOVE CLASS I PRICE FOR OWN AREA DIRECT SHIPPED MILK

IF (RDA(3).LE.IFIX(NCLIP(I)+1.5)) RDA(3)=IFIX(NCLIP(I)+0.5)+

1

2
CALL GNETA1
NARC(3)=NARC(3)+1

14

CONTINUE

...SET ARCS BETWEEN PROCESSING NODES AND SUPPLY PLANT NODES

DO 19 J=1,NSP
K2=J+NSA1
IF (SPMI(K2,I).LT.0.0) GO TO 19
RDA(1)=K2
RDA(2)=I1
IF (K2.NE.I3) GO TO 15
RDA(3)=IYP
RDA(5)=SPLB(J,OT-5)*SPRC(J)*CQTSP(J+NW)*QSPR(J)
IF (FLOAT(RDA(5)).LE.CLICP(I).OR.PCR.EQ.1) GO TO 18
XXX=FLOAT(RDA(5))-CLICP(I)
RDA(5)=IFIX(CLICP(I))
IF (KKEY.EQ.0) WRITE (OP3,66)
KKEY=1
WRITE (OP3,65) NOR(I3),ONAME(I3),XXX
GO TO 18

...ADD INSPECTION COSTS WHERE APPLICABLE

15

XIC=0.0
IF (CINSP(I).LE.0.0) GO TO 17
XIC=CINSP(I)
ICI=ILI(I)
DO 16 KJ=1,MAXN
IF (K2.EQ.ICST(ICI,KJ)+NSP) XIC=0.0

16
17

CONTINUE
XYM=CHGBT(NYCNT)*SPMI(K2,I)
RDA(3)=IFIX(CLISP(J)*XYM+XIC+HCHG2(J)+0.5)

...PRICE SUPPLY PLANT MILK FROM OTHER ORDERS AT LEAST 1 CENT ABOVE
PRICE OF SUPPLY PLANT MILK IN OWN ORDER

```
18 IF (RDA(3).EQ.IYP) WRITE (6,67)
   IF (RDA(3).EQ.IYP) RDA(3)=RDA(3)+1
   CONTINUE
   CALL GNETA1
   NARC(4)=NARC(4)+1
19 CONTINUE
```

...STATE SUPPLY TO F.O. PROCESSING

```
IF (ISTATE.NE.1) GO TO 24
RDA(2)=I1
DO 20 J=1,NSA1S
DO 20 K=1,NT11
   IJK=IFIX(SSPMI(J,K,1))
   IF (IJK.NE.1) GO TO 20
   RDA(1)=J+NT
   XYN=CHGHC(NYCNT)*ERCST(J+NT)
   XYM=CHGBT(NYCNT)*SSPMI(J,K,2)
   RDA(3)=IFIX((NCLIP(I)+BLEND(J+NT,5)-BLEND(I,5)+XYM-XYN)+0.5)
   IF (RDA(3).LE.IFIX(NCLIP(I)+1.5)) RDA(3)=IFIX(NCLIP(I)+0.5)+
1   2
   CALL GNETA1
   NARC(3)=NARC(3)+1
20 CONTINUE
   GO TO 24
21 IF (ISTATE.LT.0) GO TO 25
```

...STATE ^ F.O. SUPPLY TO STATE PROCESSING

```
RDA(2)=I1
J=I-NSA1
DO 23 K=1,NT11
   IJK=IFIX(SSPMI(J,K,3))
   IF (IJK.LE.0) GO TO 23
   IF (ISTATE.NE.1.AND.IJK.LE.NT) GO TO 23
   JKL=IJK-NSP
   IF (JKL.EQ.I) GO TO 23
   IF (IJK.GT.NSA1.AND.IJK.LE.NT) GO TO 22
   RDA(1)=IJK
   XYN=CHGHC(NYCNT)*ERCST(IJK)
   XYM=CHGBT(NYCNT)*SSPMI(J,K,4)
   IF (JKL.LE.NSA1) JKL=IJK
   RDA(3)=IFIX((NCLIP(I)+BLEND(JKL,5)-BLEND(I,5)+XYM-XYN)+0.5)
   IF (RDA(3).LE.IFIX(NCLIP(I)+1.5)) RDA(3)=IFIX(NCLIP(I)+0.5)+
1   2
   CALL GNETA1
   NARC(3)=NARC(3)+1
   GO TO 23
```

...F.O. SUPPLY PLANT TO STATE PROCESSING

```
22 RDA(1)=IJK
   XYN=CHGBT(NYCNT)*SSPMI(J,K,4)
   RDA(3)=IFIX(CLISP(IJK-NSA1)+CHG1(IJK-NSA1)*XYM+0.5)
   IF (XYM.EQ.0.0) RDA(3)=RDA(3)+1
   CALL GNETA1
```



```
      NARC(4)=NARC(4)+1
23  CONTINUE
      RDA(1)=I+NT1
      RDA(2)=I1
      RDA(3)=IFIX(NCL1P(I)+0.5)
      CALL GNETA1
      NARC(5)=NARC(5)+1
      GO TO 25
```

...SET ARCS BETWEEN PROCESSING NODES AND DUMMY SUPPLY NODES

```
24  RDA(1)=NT1+I
      RDA(2)=I1
      RDA(3)=IFIX(NCL1P(I)+0.5)
      CALL GNETA1
      NARC(5)=NARC(5)+1
25  CONTINUE
```

...SET ARCS BETWEEN (SINGLE) MANUFACTURING NODES AND SUPPLY NODES

```
DO 26 I=1,NSMF
      RDA(2)=I+NPN
      N2=2
      K1=IDMG1(I)
      IF (K1.GT.NW) N2=3
DO 26 J=1,N2
      IF (J.EQ.1) K2=K1
      IF (J.EQ.2.AND.N2.EQ.3) K2=K1+NSP
      IF (J.EQ.2.AND.N2.EQ.2.OR.J.EQ.3.AND.N2.EQ.3) K2=K1+NT1
      RDA(1)=K2
      RDA(3)=IFIX(NCL3P+0.5)
      CALL GNETA1
      NARC(6)=NARC(6)+1
```

```
26  CONTINUE
      IF (ISTATE.LE.0) GO TO 28
DO 27 I=1,NSMFS
      RDA(2)=I+NPN1
DO 27 J=1,2
      RDA(1)=I+NT
      IF (J.EQ.2) RDA(1)=I+NT2
      RDA(3)=IFIX(NCL3P+0.5)
      CALL GNETA1
      NARC(6)=NARC(6)+1
27  CONTINUE
```

...SET ARCS BETWEEN (SINGLE) MANUFACTURING NODE DUMMY AND SUPPLY NODES

```
28  RDA(2)=NMN
DO 29 I=1,NSMF
      RDA(1)=IDMG1(I)
      RDA(3)=IFIX((NCL3P+ECHG)+0.5)
      CALL GNETA1
      NARC(7)=NARC(7)+1
```

```
29  CONTINUE
      IF (ISTATE.LE.0) GO TO 31
DO 30 I=1,NSMFS
      RDA(1)=I+NT
      RDA(3)=IFIX(NCL3P+ECHG+0.5)
      CALL GNETA1
```

```
      NARC(7)=NARC(7)+1
30 CONTINUE
31 DO 32 I=1,NSMF
    IF (IDMG1(I).LE.NW) GO TO 32
    RDA(1)=IDMG1(I)+NSP
    RDA(3)=IFIX((NCL3P+ECHG)+0.5)
    CALL GNETA1
    NARC(7)=NARC(7)+1
32 CONTINUE
    DO 33 I=1,NSMF
    RDA(1)=IDMG1(I)+NT1
    RDA(3)=IFIX((NCL3P+ECHG)+0.5)
    CALL GNETA1
    NARC(7)=NARC(7)+1
33 CONTINUE
    IF (ISTATE.LE.0) GO TO 35
    DO 34 I=1,NSNFS
    RDA(1)=I+NT2
    RDA(3)=IFIX((NCL3P+ECHG)+0.5)
    CALL GNETA1
    NARC(7)=NARC(7)+1
34 CONTINUE
```

...SET ARCS BETWEEN (MULTIPLE) MANUFACTURING NODES AND SUPPLY NODES

```
35 DO 37 I=1,NMNF
    RDA(2)=I+NMN
    K4=1
    DO 36 J=1,MXN
    K1=LMMFG(I,J)
    IF (K1.LE.0) GO TO 37
    N2=2
    IF (K1.GT.NW) N2=3
    DO 35 K=1,N2
    IF (K.EQ.1) K2=K1
    IF (K.EQ.2.AND.N2.EQ.3) K2=K1+NSP
    IF (K.EQ.2.AND.N2.EQ.2.OR.K.EQ.3.AND.N2.EQ.3) K2=K1+NT1
    RDA(1)=K2
    XYM=DMMFG(I,K4)+CHGBT(NYCNT)
    RDA(3)=IFIX((NCL3P+XYM)+0.5)
    CALL GNETA1
    NARC(8)=NARC(8)+1
    IF (K.EQ.2.OR.K.EQ.3) K4=K4+1
36 CONTINUE
37 CONTINUE
```

...SET ARCS BETWEEN (MULTIPLE) MANUFACTURING NODE DUMMY AND SUPPLY NODES

```
DO 38 I=1,NMD
    RDA(2)=NMD
    K1=IDMG2(I)
    N2=2
    IF (K1.GT.NW) N2=3
    DO 38 J=1,N2
    IF (J.EQ.1) K2=K1
    IF (J.EQ.2.AND.N2.EQ.3) K2=K1+NSP
    IF (J.EQ.2.AND.N2.EQ.2.OR.J.EQ.3.AND.N2.EQ.3) K2=K1+NT1
    RDA(1)=K2
    XYM=CHGBT(NYCNT)+MMDM(I)
```

```
        RDA(3)=IFIX((NCL3P+XYM)+0.5)
        CALL GNETA1
        NARC(9)=NARC(9)+1
38 CONTINUE
```

...SET ARCS BETWEEN DUMMY PROCESSING NODES AND PROCESSING NODES

```
DO 39 I=1,NSA1
  RDA(1)=NSM+I
  RDA(2)=NMD+I
  IF (PCR.NE.1) RDA(4)=IFIX(CLICP(I)+0.5)
  CALL GNETA1
  NARC(10)=NARC(10)+1
39 CONTINUE
IF (ISTATE.LE.0) GO TO 41
DO 40 I=1,NSA1S
  RDA(1)=I+ND4
  RDA(2)=I+NDP
  RDA(4)=IFIX(CLICP(I+NSA1)+0.5)
  CALL GNETA1
  NARC(10)=NARC(10)+1
40 CONTINUE
```

...SET ARCS BETWEEN FINAL DEMAND NODES AND DUMMY PROCESSING NODES

```
41 KKEY=0
DO 47 J=1,NSA1
  RDA(2)=J+NDP1
DO 45 I=1,NSA1
  IF (PDMI(I,J).LT.0.0) GO TO 45
```

...ADD INSPECTION COSTS WHERE APPLICABLE

```
        XIC=0.0
        IF (I.EQ.J) GO TO 43
        IF (CINSP(J).LE.0.0) GO TO 43
        XIC=CINSP(J)
        ICI=ILI(J)
        DO 42 KJ=1,MAXN
          IF (I.EQ.ICST(ICI,KJ)) XIC=0.0
42 CONTINUE
43 RDA(1)=I+NMD
   XRS=CHOPRC(NYCNT)*PRCST(I)
   XYM=CHOPT(NYCNT)*PDMI(I,J)
   RDA(3)=IFIX(XRS+XYM+XIC+0.5)
   XP=(1.-FLOAT(GCNT))*MPDPM(I)
   IF (OPCL1.EQ.1) XP=IPPM(I)-FLOAT(GCNT)*MPDPM(I)
   IF (OPCL1.EQ.1.AND.MPIC.EQ.1) XP=IPPM(I)-MPDPM(I)
   IF (OPCL1.NE.1.AND.MPIC.EQ.1) XP=1.0-MPDPM(I)
   XK=CAC1B(J)*(1.+CLIRR(J))
   XK=IFIX(XK+0.5)
   IF (I.EQ.J.AND.PMR.NE.1) RDA(5)=IFIX(XP*XK)
   IF (PCR.EQ.1) GO TO 44
   IF (RDA(5).LE.IFIX(CLICP(I))) GO TO 44
   XXX=FLOAT(RDA(5))-CLICP(I)
   RDA(5)=IFIX(CLICP(I))
   IF (KKEY.EQ.0) WRITE (QP3,68)
   KKEY=1
   WRITE (QP3,65) NOR(I),ONAME(I),XXX
44 CONTINUE
```

```
CALL GNETA1  
NARC(11)=NARC(11)+1  
45 CONTINUE
```

...STATE PROCESSING TO F.O. DEMAND

```
IF (ISTATE.NE.1) GO TO 47  
DO 46 K=1,NSA1S  
DO 46 L=1,NT11  
IJK=IFIX(SSPMI(K,L,5))  
IF (IJK.NE.J) GO TO 46  
RDA(1)=K+NDP  
XRS=CHGPRC(NYCNT)*PRCST(K+NSA1)  
XYM=CHOPT(NYCNT)*SSPMI(K,L,6)  
RDA(3)=IFIX(XRS+XYM*0.5)  
CALL GNETA1  
NARC(11)=NARC(11)+1  
46 CONTINUE  
47 CONTINUE
```

...F.O. ^ STATE PROCESSING TO STATE DEMAND

```
IF (ISTATE.LE.0) GO TO 51  
DO 50 J=1,NSA1S  
KEY=0  
RDA(2)=J+NTN  
DO 50 K=1,NT11  
IJK=IFIX(SSPMI(J,K,7))  
IF (IJK.LT.0.AND.KEY.EQ.1) GO TO 50  
IF (KEY.EQ.1) GO TO 49  
IF (IJK.GT.0.AND.IJK.LT.J+NT) GO TO 49  
RDA(1)=J+NDP  
KEY=1  
XRS=CHGPRC(NYCNT)*PRCST(J+NSA1)  
RDA(3)=IFIX(XRS*0.5)  
XP=(1.-FLOAT(QCNT)*MPDPM(J+NSA1))  
XK=(COC1D(J+NSA1)*(1.+CLIRR(J+NSA1)))  
XK=IFIX(XK*0.5)  
IF (PMR.NE.1) RDA(5)=IFIX(XP*XK)  
IF (PCR.EQ.1) GO TO 48  
IF (RDA(5).LE.IFIX(CLICP(J+NSA1))) GO TO 48  
RDA(5)=IFIX(CLICP(J+NSA1))  
48 CALL GNETA1  
NARC(11)=NARC(11)+1  
IF (IJK.LE.0) GO TO 50  
49 CONTINUE  
IF (ISTATE.GT.1.AND.IJK.LE.NSA1) GO TO 50  
IF (IJK.GT.NSA1) IJK=IJK-NSP  
JKL=IJK+NTD  
RDA(1)=JKL  
XRS=CHGPRC(NYCNT)*PRCST(IJK)  
XYM=CHOPT(NYCNT)*SSPMI(J,K,8)  
RDA(3)=IFIX(XRS+XYM*0.5)  
CALL GNETA1  
NARC(11)=NARC(11)+1  
50 CONTINUE
```

...SET ARCS BETWEEN MANUFACTURING SINK NODE AND ALL MANUFACTURING
NODES

```
51 NP1=NSMF+1
DO 52 I=1, NP1
  RDA(1)=NPN+I
  IF (I.GT.NSMF) RDA(1)=NMN
  RDA(2)=NMS
  IF (I.LE.NSMF.AND.MCR.NE.1) RDA(4)=SMFC(I)
  CALL GNETA1
  NARC(12)=NARC(12)+1
52 CONTINUE
IF (ISTATE.LE.0) GO TO 54
DO 53 I=1, NSMF8
  RDA(1)=I+NPN1
  RDA(2)=NMS
  RDA(4)=IFIX(SMFC(I+NSMF))
  CALL GNETA1
  NARC(12)=NARC(12)+1
53 CONTINUE
54 NP2=NMMF+1
DO 55 I=1, NP2
  RDA(1)=NMM+I
  RDA(2)=NMS
  IF (I.LE.NMMF.AND.MCR.NE.1) RDA(4)=MMFC(I)
  CALL GNETA1
  NARC(13)=NARC(13)+1
55 CONTINUE

...SET ARCS BETWEEN SUPER SINK AND FINAL DEMAND NODES

DO 56 I=1, NSAI
  XK=CCID(I)*(1.+CLIRR(I))
  RDA(1)=NDP1+I
  RDA(2)=NSK
  RDA(4)=IFIX(XK+0.5)
  ISCL1D=ISCL1D+RDA(4)
  CALL GNETA1
  NARC(14)=NARC(14)+1
56 CONTINUE
IF (ISTATE.LE.0) GO TO 58
DO 57 I=1, NSAIS
  RDA(1)=I+NTN
  RDA(2)=NSK
  XK=CCID(I+NSAI)*(1.+CLIRR(I+NSAI))
  RDA(4)=IFIX(XK+0.5)
  ISCL1D=ISCL1D+RDA(4)
  CALL GNETA1
  NARC(14)=NARC(14)+1
57 CONTINUE

...STOP IF TOTAL DEMAND IS GREATER THAN TOTAL SUPPLY OR TOTAL
PROCESSING CAPACITY

58 WRITE (OP3,69) ISCL1D,ISTPS
IF (QCNT.EQ.1) WRITE (OP3,70) SCL1CP,ISC1CP,SSMFC,SMMFC
SMD1=FLOAT(ISCL1D-ISTPS)
IF (ISCL1D.GT.ISC1CP.AND.PCR.NE.1) GO TO 59
IF (SMD1.LE.0.0) GO TO 60
WRITE (OP3,71) SMD1
STOP 11
59 XK=ISCL1D-ISC1CP
WRITE (OP3,72) XK
```

STOP 12
60 CONTINUE

...SET ARC BETWEEN SUPER SINK AND MANUFACTURING SINK

```
RDA(1)=NMS
RDA(2)=NSK
NARC(15)=NARC(15)+1
RIA(4)=1STPS-ISCL1D
DO 61 I=1,15
61 NARCS=NARCS+NARC(I)
IF (IPNARC.NE.1.OR.OCNT.NE.1) GO TO 62
WRITE (OP3,73) (NARC(I),I=1,15),NARCS
62 CONTINUE
CALL GNETA1

REWIND OP5
WRITE (OP5) (WORK2(I),I=1,IWORK2)

TIME2=START2-CTIME(T1)

CALL GNETA2
RETURN
```

...FORMATS

```
63 FORMAT (//21X,7HQQUARTER,I2,1X,4HYEAR,I5/21X,19(1H=))
64 FORMAT (1H0,40HTHE SUM OF (OWN AREA) DIRECT SHIPPED AND,/1H0,43HSU
1PLY PLANT SHIPPING REQUIREMENTS EXCEEDS ,/1H0,31H(OWN AREA) PROCE
2SSING CAPACITY.,/1H0,37HTOTAL SHIPPING REQUIRMENTS HAVE BEEN ,/1H0
3,36HSET EQUAL TO PROCESSING CAPACITY BY ,/1H0,36HPROPORTIONAL REDU
4CTION OF EACH TYPE ,/1H0,29HOF SHIPPING REQUIRMENT.,2X,4HAREA,8X,2
57HEXCESS AMOUNT (THOUS. LBS.)/8X,4(1H-),8X,15(1H-))
65 FORMAT (2X,I3,2X,A10,F14.0)
66 FORMAT (1H0,41HSUPPLY PLANT SHIPPING REQUIREMENT EXCEEDS,36H PROCE
1SSING CAPACITY IN ITS OWN AREA/1X,58H**** SHIPPING REQUIREMENT SET
2 EQUAL TO PROCESSING CAPACITY,5H ****/8X,4HAREA,8X,28HEXCESS AMOUN
3T (THOUS. LBS.)/8X,4(1H-),8X,15(1H-))
67 FORMAT (1X,32HSUPPLY PLANT TIE BREAKER INVOKED)
68 FORMAT (1H0,42HOWN AREA PACKAGED MILK SALES REQUIREMENT EXCEEDS,20
1H PROCESSING CAPACITY/1X,54H**** REQUIREMENT SET EQUAL TO PROCESSI
2NG CAPACITY ****/8X,4HAREA,8X,27HEXCESS AMOUNT (THOUS. LBS.)/8X,4(
31H-),8X,15(1H-))
69 FORMAT (//1X,42HTOTAL CLASS I DEMAND + REQUIRED RESERVE = ,I15,4X,
111HTHOUS. LBS./1X,42HTOTAL SUPPLY = ,I
215,4X,111HTHOUS. LBS.)
70 FORMAT (////1X,40HTOTAL PROCESSING CAPACITY = ,F15.0,4
1X,111HTHOUS. LBS./1X,40HTOTAL PROCESSING CAPACITY + RESERVE = ,I
214,5X,111HTHOUS. LBS./1X,28HTOTAL MANUFACTURING CAPACITY,/1X,16
3MSINGLE = ,F15.0,4X,111HTHOUS. LBS./1X,16HMULTIPLE = ,
4F15.0,4X,111HTHOUS. LBS.)
71 FORMAT (1X,36HTOTAL DEMAND EXCEEDS TOTAL SUPPLY BY,I15,3X,111HTHOU
1. LBS.)
72 FORMAT (1X,28HTOTAL CLASS I DEMAND EXCEEDS/1X,36HTOTAL CLASS I PRO
1CESSING CAPACITY BY,F15.0,3X,111HTHOUS. LBS.)
73 FORMAT (////1H0,20HNUMBER OF ARCS IN EACH SECTOR//1X,12HARCS BETWEEN
1H//13X,29HSUPER SOURCE AND SUPPLY NODES,20X,I10/13X,45HDIRECT SHI
2PED AND DIRECT SHIPPED DUMMY NODES,4X,I10/13X,35HDIRECT SHIPPED A
3ND PROCESSING NODES,14X,I10/13X,36HSUPPLY PLANT AND PROCESSING NOO
4ES,16X,I10/13X,41HDIRECT SHIPPED DUMMY AND PROCESSING NODES,8X,I10
```

5/13X,37HSUPPLY AND SINGLE MANUFACTURING NODES,12X,I10/13X,40HSUPPL
 6Y AND SINGLE MANUFACTURING NODE DU,3HMMY,6X,I10/13X,37HSUPPLY NOD
 7ES AND MULTIPLE MANUFACTURI,8HNG NODES,4X,I10/13X,32HSUPPLY NODES
 8AND MULTIPLE MANUFA,18HCTURING NODE DUMMY,I9/13X,25HPROCESSING NOD
 9ES AND DUMM,18HY PROCESSING NODES,6X,I10/13X,21HDUMMY PROCESSING N
 *ODE,18HS AND DEMAND NODES,10X,I10/13X,20HSINGLE MANUFACTURING,2SH
 *NODES AND MANUFACTURING SINK,I10/13X,14HMULTIPLE MANUF,37HACTURING
 * NODES AND MANUFACTURING SINK,I8/13X,27HDEMAND NODES AND SUPER SIN
 *K,22X,I10/13X,33HMANUFACTURING SINK AND SUPER SINK,16X,I10//3X,17H
 *TOTAL NO. OF ARCS,42X,I10//)

74 FORMAT (/1X,14HGRADE A MARKET,/1X,14(1H-))

END
 SUBROUTINE PREP

...PREP CONTROLS EXECUTION OF MANUFACTURING MILK SECTION OF DAMPS

COMMON /BLK1/ COMFG(9),COIMP(4,2),COBS(2,3,9),COC(5,5),COBNS(2,9),
 1NARC(15),BMMD1(31,15),BPOP(31),BPRR(15),BMMD2(58,10),POPB(58),PRRB
 2(10),RESID(18,2),RESIDC(18,2),RESIDG(18,2),CHEESE(9),WK2(9),WK1(9)
 3,CQB(9),CQUREG(9),CQUC1S(9),CQUGBP(9),CQUC1P(9),CQUNBP(9),CQURD(9
 4),CQUUP(9),CQUNRP(9),CGES(6,9),TRCS1,TRCS2,CINU(9,3),BNINU(2,9),SP
 5ACE(4224),SPACE1(5376)

COMMON /BLK5/ CPRCHG(59),BTPSQ(59,4),BLEND(59,6),SE(59),GRBIND(4),
 1GRBCNU(59),IGBEF,COTSP(59),QSPL1(59),ICMEF

COMMON /BLK7/ NAD,NOD,MXC,NNODES,IPRI(20),IPRO(20),QCNT1,OP4,KGNET
 COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7

COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
 120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
 2XC,NNOD,NNNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NQ,NIC,ISUP1,ISUP7,NS
 3PM,NSC,NSK,IHCEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISC1CP,CHGCL3,ACST,
 4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LBCHK,IPNARC,SCL1CP,T
 5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7

COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP
 1RC(5),TRANSR(59),OACL1T(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1
 20),HCHG2(16),MXN,IPCEF,CQC2D(59),CQC1D(59),NRPR(59),XCL2P,DTCST,IN
 3ORK3,SFPCST,DFPCST,TRMPC,TMCST,RMPC(75),EFR(59)

COMMON /BLK14/ IDXB(9,12),UNRS(9),B(9,4),BIMP(4,2,4),C(5,5,4),RPB(
 15,5),BS(6,9,4),BPC(9),BSE(9),BDE(5),BTRANS(11,5),BBC3P(5),BCHGI(4)
 2,PCHGI(4),NRPB(5),CHGMC(5,5),SIDX(3,4),RSS(9),CF(5),BMKP(5),ERCST(
 35),BNF(2),BTCST(2,5),ISS,UNUT,UNPC,DGSC(20),IMPHLD,UNS(4),UNGBP(9,
 44),UNFPHC,UNPMC,ECHBT(5),BFPHC

COMMON /BLK15/ MS,MIMP,MC,M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M
 113,M14,M15,M16,M17,M18,M19,M20,M21,M22,M23,M24,M25,M26,M27,M28,M29
 2,M30,M31,M32,M33,M34,M35,M36,M37,M38

INTEGER QR,QT,QCNT,QCNT1,CR,PCR,PMR,RMR,RDA,OPEXF,OPCL1,OP1,OP2,OP
 13,OP5,OP6,OP7,OP4,BIGI

REAL NFP,IACL1,MPDRM,MMFC,MPDPM,IPPM,NCL2Q,MCL2Q,MCL3Q,NRPR,NCL1P,
 1MFGS,MFGM,IACL1S,MTPSQ,MCL1Q,ICHG,MOCC,MOPC,MFPC,MRTLP,NCL2P,MLEN
 2D,MCL1P,MCL1U,MACL1,MMGM,ICP,NCL3,NCL3P,NTPSQ,MMFG,NRPB

DATA RESID/36*0.0/,RESIDC/36*0.0/,RESIDG/36*0.0/

...SET GRADE B MODEL PARAMETERS

MS=9
 MIMP=2
 MC=5

...STAGE 1

```
M1=MS+1
M2=MS+MIMP
M3=M2+1
M4=M2+MIMP
M5=M4+1
M6=M4+MS
M7=MS+1
M8=MS+MS
M9=MS+1
M10=MS+MC
M11=M10+1
M12=M10+MC
M13=M12+1
M14=M12+MC
M15=M14+1
M16=M15+1
M17=M16+1
```

...STAGE 2

```
M18=MS+1
M19=MS+MIMP
M20=M19+1
M21=M19+MS
M22=M21+1
M23=M21+MS
M24=M23+1
M25=M23+MS
M26=M25+1
M27=M25+MIMP
M28=M27+1
M29=M27+MS
M30=M29+1
M31=M29+MS
M32=M31+1
M33=M31+MC
M34=M33+1
M35=M33+MC
M36=M35+1
M37=M36+1
M38=M37+1
```

...COMPUTE MFG. PRODUCTION BY REGIONS

```
COUCEX=0.0
DO 3 I=1,MS
  COMFB(I)=0.0
  WK1(I)=0.0
  WK2(I)=0.0
  COUCIP(I)=0.0
  CGUNRP(I)=0.0
  NIXB=0
  DO 1 J=1,12
    IXB=IDXB(I,J)
    IF (IXB.EQ.0) GO TO 2
    COUCIP(I)=COUCIP(I)+NCLIP(IXB)
    CGUNRP(I)=CGUNRP(I)+NRPR(IXB)
```



```

NIXB=NIXB+1
WK1(I)=WK1(I)+CQTSP(IXB)
WK2(I)=WK2(I)+CQC1D(IXB)
IF (IGBEF.EQ.1) COMFG(I)=COMFG(I)-FLOAT(NYCNT)*GRBCNU(IXB)*G
1 RBIND(QT-5)
1 CONTINUE
2 CQUC1P(I)=CQUC1P(I)/FLOAT(NIXB)
  CQUNRP(I)=CQUNRP(I)/FLOAT(NIXB)
  CQUGBP(I)=CQUC1P(I)*UNUT+NCL3P*(1.0-UNUT)
  CQUNBP(I)=CQUGBP(I)-UNFPHC*CHGHC(NYCNT)-UNPMC
  CQUREG(I)=UNREG(I)
  CQUC1S(I)=CQUREG(I)*UNUT
  CQUC1EX=CQUC1EX+CQUNRP(I)*CQUC1S(I)
  CQUUP(I)=CQUGBP(I)*CQUREG(I)
  U1=1.0
  IF (IPDEF.EQ.1) U1=CHGPD(QCNT)-1.0
  CQUROD(I)=CQUNBP(I)-UNPC*U1
  CQB(I)=GRADEB(I)+COMFG(I)
  COMFG(I)=CQB(I)+WK1(I)-WK2(I)+CQUREG(I)*(1.0-UNUT)
3 CONTINUE

...WRITE UNREGULATED VARIABLES TO GRADE A REPORT WRITER - DRW1
WRITE (QP2) CQUREG,CQUC1S,CQUGBP,CQUNBP,CQUROD

...COMPUTE NEW RETAIL PRICES FOR MANUFACTURED PRODUCTS
NRPB(1)=(NCL2P+BRCST(1))*BMKP(1)
NRPB(2)=(NCL3P+BRCST(2))*BMKP(2)
NRPB(3)=(NCL3P+BRCST(3))*BMKP(3)
NRPB(4)=((NCL3P*BNF(1))+BRCST(4))*BMKP(4)/CF(4)
NRPB(5)=((NCL3P*BNF(2))+BRCST(5))*BMKP(5)/CF(5)

...CALCULATE IMPORTS, BEGINNING STOCKS, AND CONSUMPTION, BY REGION
...IMPORTS
DO 4 I=1,4
DO 4 J=1,MIMP
  CQIMP(I,J)=RIMFC(I,J)
4 CONTINUE

...BEGINNING STOCKS (IF ISS=0 GOVT STOCKS ARE NOT RECYCLED)
DO 5 K=1,2
DO 5 I=1,3
DO 5 J=1,MS
  II=I
  IF (K.EQ.2) II=I+3
  CQBS(K,I,J)=BS(II,J,4)
  IF (ISS.EQ.0.AND.K.EQ.2) CQBS(K,I,J)=0.5*BS(II,J,4)
5 CONTINUE

...CONSUMPTION (PLUS MFD. STOCKS REQUIREMENT)
DO 6 I=1,5
DO 6 J=1,MC
  CQC(I,J)=CMMP(I,J)
  IF (I.EQ.2) CQC(I,J)=CQC(I,J)*(1.0+SIDX(1,QT-5))
  IF (I.GE.4) CQC(I,J)=CQC(I,J)*(1.0+SIDX(I-2,QT-5))
```

```

6 CONTINUE
  CALL NETGNB
  CALL FIXUPA
  CALL NETGNC
  CALL FIXUPB

```

...OUTPUT DATA FOR REPORT WRITER - DRW2

```

DO 7 I=1,6
DO 7 K=1,MS
  COES(I,K)=BS(I,K,4)
7 CONTINUE

DO 8 I=2,5
DO 8 J=1,MC
  CQC(I,J)=CMMP(I,J)
8 CONTINUE

WRITE (DP7) WK1,COB,WK2,COMFG,CQIMP,COES,CQC,NRPB,CQUREG,CQUC1S,CQ
1UGBP,CQUUP,CQUROD,CQUCEX,BMMD1,BMMD2,TRCS1,TRCS2

RETURN

END
SUBROUTINE NETGNB

```

...NETGENB GENERATES GRADE B TRANSHIPMENT PROBLEM FOR CNET
 THE PROBLEM IS SOLVED IN TWO STAGES. IN STAGE ONE, A NETWORK
 IS GENERATED FOR CLASS II, CHEESE, AND MISCELLANEOUS CLASS III
 PRODUCTS. IN STAGE TWO (NETGNC), A NETWORK IS GENERATED FOR
 BUTTER AND NONFAT DRY MILK.

```

COMMON /BLK1/ COMFG(9),CQIMP(4,2),COBS(2,3,9),CQC(5,5),COBNS(2,9),
1NARC(15),BMMD1(31,15),BPOP(31),BPRR(15),BMMD2(58,10),POPB(58),PRRB
2(10),RESID(18,2),RESIDC(18,2),RESIDG(18,2),CHEESE(9),WK2(9),WK1(9)
3,COB(9),CQUREG(9),CQUC1S(9),CQUGBP(9),CQUC1P(9),CQUNBP(9),CQUROD(9
4),CQUUP(9),CQUNRP(9),COES(6,9),TRCS1,TRCS2,CINU(9,3),BNINU(2,9),SP
5ACE(4224),SPACE1(5376)
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIGI,IMR,IMRN(11
1)
COMMON /BLK5/ CFRCHG(59),BTSPG(59,4),BLEND(59,6),SE(59),GRBIND(4),
1GRBCNV(59),ICREF,COTSP(59),QSPL1(59),ICMEF
COMMON /BLK7/ NAD,NDD,MXC,NKCES,IFRI(20),IPRO(20),QCNT1,OP4,KCNET
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7
COMMON /BLK10/ MMFC(27),CL1CP(39),CL1SP(16),IR(61),CHGHC(5),CHCPD(
120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMFC,SM
2XC,NNOD,MMNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NG,NIC,ISUP1,ISUP7,NS
3PM,NSC,NSK,INDEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISC1CP,CHGCL3,ACST,
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,FMR,MCR,OPCL1,LECHK,IPNAPC,SCL1CP,T
5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7
COMMON /BLK14/ IDXB(9,12),UNRS(9),B(9,4),BIMP(4,2,4),C(5,5,4),RPB(
15,5),BS(6,9,4),BPC(9),BSE(9),BDE(5),BTRANS(11,5),BBC3P(5),ECHGI(4)
2,FUNGI(4),NRPB(5),CHGHC(5,5),SIDX(3,4),RSS(9),CP(5),BMKP(5),ERCST(
3),BMP(5),BTCST(2,5),ISS,UNUT,UNFC,DGSC(20),IMFHLD,UNS(4),UNGBP(9,
4),UNTHD,UNPDC,ECHST(5),BFPNC
COMMON /BLK17/ MS,M1P,MC,M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M

```

113, M14, M15, M16, M17, M18, M19, M20, M21, M22, M23, M24, M25, M26, M27, M28, M29
2, M30, M31, M32, M33, M34, M35, M36, M37, M38

INTEGER QR, QT, QCNT, QCNT1, CR, PCR, PMR, RMR, RDA, OPEXF, OPCL1, OP1, OP2, OP
13, OP5, OP6, OP7, OP4, BIGI

REAL NFP, IACL1, MPDRM, MMFC, MPDPM, IPPM, NCL2Q, MCL2Q, MCL3Q, NRPR, NCL1P,
1MFGS, MFGM, IACL1S, MTPSQ, MCL1Q, ICHG, MOCC, MOPC, MPPC, MRTLP, NCL2P, MBLN
2D, MCL1P, MCL1U, MIACL1, MMDM, ICP, NCL3, NCL3P, NTPSQ, MMFG

...SET PARAMETERS REQUIRED FOR GNET AND RESETB

MXC=425
NOD=51
NAD=1000
NNODES=47
KGNET=1

DO 1 I=1,15
NARC(I)=0
IF (I.GT.5) GO TO 1
RDA(I)=0
1 CONTINUE

***** STAGE 1 *****

...SET ARCS FROM SUPER SOURCE TO SUPPLY, IMPORTS, AND STOCKS

ISPS=0
IDMS=0
CALL GNETA
DO 7 J=1, M3
RDA(1)=M16
RDA(2)=J
IF (J.GT.M3) GO TO 2
RDA(4)=IFIX(COMFG(J)+0.5)
ISPS=ISPS+RDA(4)
GO TO 6
2 IF (J.GT.M2) GO TO 3
RDA(4)=IFIX(COIMP(1, J-M3)+0.5)
ISPS=ISPS+RDA(4)
GO TO 6
3 IF (J.GT.M4) GO TO 4
RDA(4)=IFIX(COIMP(2, J-M2)+0.5)
ISPS=ISPS+RDA(4)
GO TO 6
4 IF (J.GT.M6) GO TO 5
RDA(4)=IFIX(COBS(1,1, J-M4)+0.5)
ISPS=ISPS+RDA(4)
GO TO 6
5 RDA(4)=IFIX(COBS(2,1, J-M6)+0.5)
ISPS=ISPS+RDA(4)
6 NARC(1)=NARC(1)+1
CALL GNETA1
7 CONTINUE

...SET ARCS TO CLASS II CONSUMPTION FROM SUPPLY

DO 8 J=M9, M10
DO 8 I=1, M5

```
      RDA(1)=I
      RDA(2)=J
      RDA(3)=IFIX((((BTCST(1,1)+BTCST(2,1)*BTRANS(I,J-M8))/CF(1))*ECH
1      BT(NYCNT))+0.5)
      NARC(2)=NARC(2)+1
      CALL GNETA1
8 CONTINUE
```

...SET ARCS TO CHEESE CONSUMPTION FROM SUPPLY AND CHEESE IMPORTS AND COMMERCIAL AND GOVERNMENT STOCKS

```
DO 10 J=M11,M12
  RDA(2)=J
  DO 9 I=1,M2
    RDA(1)=I
    RDA(3)=IFIX((((BTCST(1,2)+BTCST(2,2)*BTRANS(I,J-M10))/CF(2))
1    *ECHBT(NYCNT))+0.5)
    IF (I.LT.M1) RDA(3)=RDA(3)+1
    NARC(3)=NARC(3)+1
    CALL GNETA1
  9 CONTINUE
DO 10 I=M5,M8
  RDA(1)=I
  II=I-M4
  IF (I.GE.M7) II=I-M6
  RDA(3)=IFIX((((BTCST(1,2)+BTCST(2,2)*BTRANS(II,J-M10))/CF(2))*E
1  CHBT(NYCNT))+0.5)
  NARC(3)=NARC(3)+1
  CALL GNETA1
10 CONTINUE
```

...SET ARCS TO MISC. III CONSUMPTION FROM SUPPLY AND III IMPORTS

```
DO 12 J=M13,M14
  RDA(2)=J
  DO 11 I=1,M5
    RDA(1)=I
    RDA(3)=IFIX((((BTCST(1,3)+BTCST(2,3)*BTRANS(I,J-M12))/CF(3))
1    *ECHBT(NYCNT))+0.5)
    NARC(4)=NARC(4)+1
    CALL GNETA1
  11 CONTINUE
DO 12 I=M3,M4
  RDA(1)=I
  RDA(3)=IFIX((((BTCST(1,3)+BTCST(2,3)*BTRANS(I-M2+M5,J-M12))/CF(
1  3))*ECHBT(NYCNT))+0.5)
  NARC(4)=NARC(4)+1
  CALL GNETA1
12 CONTINUE
```

...SET ARCS TO RESIDUAL SINK FROM SUPPLY AND COMM. AND GOVT. STOCKS

```
      RDA(2)=M15
DO 13 I=1,M5
  RDA(1)=I
  NARC(5)=NARC(5)+1
  CALL GNETA1
13 CONTINUE
DO 14 I=M5,M8
  RDA(1)=I
```

```
        NARC(6)=NARC(6)+1
        CALL GNETA1
14 CONTINUE

...SET ARCS TO SUPER SINK FROM CLASS II, CHEESE, AND MISC. CLASS III
DEMANDS AND RESIDUAL SINK

        DO 19 I=M9,M15
            RDA(1)=I
            RDA(2)=M17
            IF (I.GT.M10) GO TO 15
            RDA(4)=IFIX(CQC(1,I-M8)+0.5)
            IDMS=IDMS+RDA(4)
            GO TO 18
15         IF (I.GT.M12) GO TO 16
            RDA(4)=IFIX(CQC(2,I-M10)+0.5)
            IDMS=IDMS+RDA(4)
            GO TO 18
16         IF (I.GT.M14) GO TO 17
            RDA(4)=IFIX(CQC(3,I-M12)+0.5)
            IDMS=IDMS+RDA(4)
            GO TO 18
17         RDA(4)=ISPS-IDMS
18         NARC(7)=NARC(7)+1
            CALL GNETA1
19 CONTINUE
        IF (OCNT.EQ.1) CALL NARCW (NARC,DP3)
        CALL GNETA2

        RETURN

        ENTRY NETGNC
```

***** STAGE 2 *****

```
...SET PARAMETERS AND PERFORM COMPUTATIONS FOR STAGE TWO

        MXC=1700
        NOD=73
        NAD=2000
        NNODES=69
        KNET=2

        DO 20 L=1,5
            RDA(L)=0
20 CONTINUE

...SET ARCS FROM SUPER SOURCE TO BUTTER AND NFDM SUPPLIES, IMPORTS,
AND COMMERCIAL AND GOVERNMENT STOCKS.

        ISPS=0
        IDMS=0
        CALL GNETA
        DO 29 J=1,M31
            RDA(1)=M37
            RDA(2)=J
            IF (J.GT.M5) GO TO 21
            RDA(4)=IFIX(CQBNS(1,J)+0.5)
            ISPS=ISPS+RDA(4)
            GO TO 23
```

```
21 IF (J.GT.M19) GO TO 22
   RDA(4)=IFIX(COIMP(3,J-M9)+0.5)
   ISPS=ISPS+RDA(4)
   GO TO 28
22 IF (J.GT.M21) GO TO 23
   RDA(4)=IFIX(COBS(1,2,J-M19)+0.5)
   ISPS=ISPS+RDA(4)
   GO TO 28
23 IF (J.GT.M23) GO TO 24
   RDA(4)=IFIX(COBS(2,2,J-M21)+0.5)
   ISPS=ISPS+RDA(4)
   GO TO 28
24 IF (J.GT.M25) GO TO 25
   RDA(4)=IFIX(COENS(2,J-M23)+0.5)
   ISPS=ISPS+RDA(4)
   GO TO 28
25 IF (J.GT.M27) GO TO 26
   RDA(4)=IFIX(COIMP(4,J-M25)+0.5)
   ISPS=ISPS+RDA(4)
   GO TO 28
26 IF (J.GT.M29) GO TO 27
   RDA(4)=IFIX(COBS(1,3,J-M27)+0.5)
   ISPS=ISPS+RDA(4)
   GO TO 28
27 RDA(4)=IFIX(COBS(2,3,J-M29)+0.5)
   ISPS=ISPS+RDA(4)
28 NARC(8)=NARC(8)+1
   CALL GNETA1
29 CONTINUE
```

...SET ARCS TO BUTTER CONSUMPTION FROM BUTTER SUPPLY, IMPORTS,
AND COMMERCIAL AND GOVERNMENT STOCKS

```
DO 33 J=M32,M33
   RDA(2)=J
DO 33 I=1,M23
   RDA(1)=I
   IF (I.GE.M20) GO TO 30
   RDA(3)=IFIX(((BTCST(1,4)+BTCST(2,4)*BTRANS(I,J-M31))*ECHBT(NYCN
1 T))+0.5)
   GO TO 32
20 IF (I.GE.M22) GO TO 31
   RDA(3)=IFIX(((BTCST(1,4)+BTCST(2,4)*BTRANS(I-M19,J-M31))*ECHBT(
1 NYCNT))+0.5)
   GO TO 32
31 RDA(3)=IFIX(((BTCST(1,4)+BTCST(2,4)*BTRANS(I-M21,J-M31))*ECHBT(
1 NYCNT))+0.5)
22 NARC(9)=NARC(9)+1
   CALL GNETA1
33 CONTINUE
```

...SET ARCS TO NDM CONSUMPTION FROM NDM SUPPLY, IMPORTS, AND
COMMERCIAL AND GOVERNMENT STOCKS

```
DO 37 J=M34,M35
   RDA(2)=J
DO 37 I=M24,M31
   RDA(1)=I
   IF (I.GE.M28) GO TO 34
   RDA(3)=IFIX(((BTCST(1,5)+BTCST(2,5)*BTRANS(I-M23,J-M33))*ECHBT(
```

```
1 NYCNT))+0.5)
GO TO 36
34 IF (I.GE.M30) GO TO 35
RDA(3)=IFIX(((BTCST(1,5)+BTCST(2,5)*BTRANS(I-M27,J-M33))*ECHBT(
1 NYCNT))+0.5)
GO TO 36
35 RDA(3)=IFIX(((BTCST(1,5)+BTCST(2,5)*BTRANS(I-M29,J-M33))*ECHBT(
1 NYCNT))+0.5)
36 NARC(10)=NARC(10)+1
CALL GNETA1
37 CONTINUE
```

...SET ARCS TO RESIDUAL SINK FROM SUPPLIES AND STOCKS

```
RDA(2)=M36
DO 38 I=1,M5
RDA(1)=I
NARC(11)=NARC(11)+1
CALL GNETA1
38 CONTINUE
DO 39 I=M20,M23
RDA(1)=I
NARC(12)=NARC(12)+1
CALL GNETA1
39 CONTINUE
DO 40 I=M24,M25
RDA(1)=I
NARC(13)=NARC(13)+1
CALL GNETA1
40 CONTINUE
DO 41 I=M28,M31
RDA(1)=I
NARC(14)=NARC(14)+1
CALL GNETA1
41 CONTINUE
```

...SET ARCS TO SUPER SINK FROM BUTTER AND NFDM DEMAND AND RESIDUAL SINK

```
DO 45 I=M32,M36
RDA(1)=I
RDA(2)=M38
IF (I.GT.M33) GO TO 42
RDA(4)=IFIX(COC(4,I-M31)+0.5)
IDMS=IDMS+RDA(4)
GO TO 44
42 IF (I.GT.M35) GO TO 43
RDA(4)=IFIX(COC(5,I-M33)+0.5)
IDMS=IDMS+RDA(4)
GO TO 44
43 RDA(4)=ISPS-IDMS
44 NARC(15)=NARC(15)+1
CALL GNETA1
45 CONTINUE
IF (OCNT.EQ.1) CALL NARCW (NARC,OP3)
CALL GNETA2
RETURN

END
SUBROUTINE NARCW (NARC,OP3)
```

...WRITE OUT NUMBER OF ARCS BY SECTOR AND TOTAL NUMBER

```

COMMON /BLK7/ NAD,NOD,MXC,NNODES,IPRI(20),IPRO(20),GCNT1,OP4,KGNET
DIMENSION NARC(15)
INTEGER OPS
NRC=0
IF (KGNET.EQ.2) GO TO 4
DO 1 I=1,7
  NRC=NRC+NARC(I)
1 CONTINUE
WRITE (OP3,2)
2 FORMAT (///1X,29HNUMBER OF ARCS IN EACH SECTOR//)
WRITE (OP3,3) (NARC(I),I=1,7),NRC
3 FORMAT (5X,15H--- STAGE 1 ---//1X,12HARCS BETWEEN//10X,29HSUPER SO
1URCE AND SUPPLY NODES,21X,110/10X,51HCLASS II CONSUMPTION REGIONS
2AND SUPPLIES ,19/10X,39HCHEESE CONSUMPTION TO SUPPLY AND
3STOCKS,11X,110/10X,40HCIII CONSUMPTION TO SUPPLIES AND IMPORTS,10X
4,110/10X,42HRESIDUAL SINK FROM SUPPLY REGIONS ,8X,110/10X,
54HRESIDUAL SINK FROM COMM. AND GOVT. STOCKS ,8X,110/10X,50HCII,CH
6ESEE,CIII DEMAND AND RESIDUAL TO SUPER SINK ,110//14X,20HTOTAL NUM
7BER OF ARCS,26X,110//)
RETURN
4 DO 5 J=8,15
  NRC=NRC+NARC(J)
5 CONTINUE
WRITE (OP3,2)
WRITE (OP3,6) (NARC(I),I=8,15),NRC
6 FORMAT (5X,15H--- STAGE 2 ---//1X,12HARCS BETWEEN//10X,46HSUPER SO
1URCE TO BUTTER,NFDM,IMPORTS,AND STOCKS,4X,110/10X,48HBUTTER DEMAND
2 FROM SUPPLIES, IMPORTS, AND STOCKS,2X,110/10X,47HNFDM DEMANDS FRO
3M SUPPLIES, IMPORTS, AND STOCKS,3X,110/10X,43HRESIDUAL SINK FROM B
4UTTER SUPPLY REGIONS ,7X,110/10X,39HRESIDUAL SINK FROM BUTTER ST
5OCKS ,11X,110/10X,38HRESIDUAL SINK FROM NFDM SUPPLY REGIONS,
612X,110/10X,38HRESIDUAL SINK FROM NFDM STOCKS ,11X,110/10X
7,47HBUTTER, NFDM DEMANDS AND RESIDUAL TO SUPER SINK,3X,110//14X,20
8HTOTAL NUMBER OF ARCS,26X,110//)
RETURN

END
SUBROUTINE FIXUPA

```

...UPDATES AND COMPUTES VALUES FROM STAGE 1 OF NETGENB

```

COMMON /BLK1/ COMFG(9),COINP(4,2),COBS(2,3,9),COG(5,5),COBNS(2,9),
1NARC(15),BMMD1(31,15),BPOP(31),BPRR(15),BMMD2(58,10),POPB(58),PRRB
2(10),RESID(18,2),RESIDC(18,2),RESIDG(18,2),CHEESE(9),WK2(9),WK1(9)
3,COB(9),COUREG(9),COUC1S(9),COUC2P(9),COUC1P(9),COUNBP(9),COUROD(9
4),COUUP(9),COUNRP(9),CGES(8,9),TRCS1,TRCS2,CINU(9,3),BINU(2,9),DS
5(4,9),SPACE(4189),SPACE1(5376)
COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
120),ERCST(75),SPRC(16),IDMG1(25),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
2XC,NNDD,NNNE,NNND,ISTPS,NYRBC,NYBASE,OT,GCNT,NO,NIC,ISUP1,ISUP7,NS
3PM,NSC,NSK,IMDEF,IPDEF,EDGE,NCL2P,NCL3P,CHGCL2,ISCL1P,CHGCL3,ACST,
4PMCHG,RMGN,DCST2,CEMGN,RMR,PCR,PYR,MCR,OPCL1,LECHK,IPNARC,SCL1CP,T
5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7
COMMON /BLK14/ IDXB(9,12),UNRS(9),B(9,4),BIMP(4,2,4),C(5,5,4),RPB(
15,5),BS(6,9,4),DPC(9),BSE(9),BDE(9),BTRANG(11,5),BECSP(5),BCHGI(4)
2,PCHGI(4),NRPB(5),CHGMC(5,5),SIDX(3,4),RSS(9),CF(5),BMKP(5),BRCST(
35),BMF(2),BTCST(2,5),ISC,UNUT,UNPC,DGSC(20),IMPHLB,UNS(4),UNGBP(9,
44),UNFPHC,UNPRC,ECHBT(5),BFRAC

```



```
COMMON /BLK15/ MS, MIMP, MC, M1, M2, M3, M4, M5, M6, M7, M8, M9, M10, M11, M12, M13, M14, M15, M16, M17, M18, M19, M20, M21, M22, M23, M24, M25, M26, M27, M28, M29, M30, M31, M32, M33, M34, M35, M36, M37, M38
```

```
INTEGER QT, QCNT
```

```
...COMPUTE DESIRED STOCK LEVELS
```

```
DO 1 J=1, MS  
  DS(4, J)=RSS(J)*DGSC(QCNT)  
1 CONTINUE
```

```
...COMPARE OLD CHEESE STOCK CARRYOVER WITH DESIRED STOCKS
```

```
DO 3 J=1, MS  
  PRINT 2, (CINU(J, L), L=1, 3), RESIDC(J, 1), RESIDG(J, 1), RESID(J, 1)  
2 FORMAT (/5X, 7HREMAINS, 6F12.0)  
3 CONTINUE  
  PRINT 19
```

```
DO 7 J=1, MS  
  DS(1, J)=0.0  
  DO 4 L=1, 3  
    DS(1, J)=DS(1, J)+CINU(J, L)
```

```
4 CONTINUE  
  D1=0.0  
  D2=0.0  
  BS(1, J, 4)=CINU(J, 2)+RESIDC(J, 1)  
  BS(4, J, 4)=CINU(J, 3)+RESIDG(J, 1)  
  D1=DS(1, J)-BS(1, J, 4)  
  IF (D1.GT.0.0) GO TO 5  
  RESID(J, 1)=RESID(J, 1)+CINU(J, 1)  
  BS(1, J, 4)=DS(1, J)  
  BS(4, J, 4)=BS(4, J, 4)-D1  
  GO TO 7
```

```
5 D2=D1-BS(4, J, 4)  
  IF (D2.GT.0.0) GO TO 6  
  BS(1, J, 4)=BS(1, J, 4)+D1  
  BS(4, J, 4)=BS(4, J, 4)-D1  
  RESID(J, 1)=RESID(J, 1)+CINU(J, 1)  
  GO TO 7  
6 BS(1, J, 4)=BS(1, J, 4)+BS(4, J, 4)+D2  
  BS(4, J, 4)=0.0  
  RESID(J, 1)=RESID(J, 1)+CINU(J, 1)-D2
```

```
7 CONTINUE  
DO 9 J=1, MS  
  D4=0.0  
  D4=DS(4, J)-BS(4, J, 4)  
  IF (D4.LT.0.0) GO TO 9  
  RESID(J, 1)=RESID(J, 1)-D4  
  IF (RESID(J, 1).LT.0.0) GO TO 8  
  BS(4, J, 4)=BS(4, J, 4)+D4  
  GO TO 9
```

```
8 D4=D4+RESID(J, 1)  
  BS(4, J, 4)=BS(4, J, 4)+D4  
  RESID(J, 1)=0.0  
  PRINT 15, J
```

```
9 CONTINUE  
  PRINT 19  
  PRINT 20, (BS(1, J, 4), J=1, MS)  
  PRINT 19
```

```
PRINT 16, (DS(4,J),J=1,MS)
PRINT 20, (BS(4,J,4),J=1,MS)
PRINT 19
PRINT 18, (RESID(I,1),I=1,S)
```

...COMPUTE BUTTER AND NFDM PRODUCTION FOR STAGE TWO

```
CFC=CF(2)/100.0
CH1=0.08165
FL1=0.0029
DO 10 J=1,MS
  COBNS(1,J)=RESID(J,1)*CF(4)/127.0+CH1*CHEESE(J)*CFC+FL1*WK2(J)
  COBNS(2,J)=RESID(J,1)*CF(5)/170.0
```

10 CONTINUE

```
DO 11 I=1,18
DO 11 J=1,2
  RESID(I,J)=0.0
  RESIDC(I,J)=0.0
  RESIDG(I,J)=0.0
```

11 CONTINUE

RETURN

ENTRY FIXUPB

...UPDATES AND COMPUTES VALUES FROM STAGE 2 OF NETGENB

...COMPARE OLD BUTTER AND NFDM CARRYOVER WITH DESIRED COMM. STOCKS

```
DO 12 I=2,3
DO 12 J=1,MS
  BS(I,J,4)=BNINU(I-1,J)
  JJ=J
  IF (I.EQ.3) JJ=J+MS
  II=I+3
  ES(II,J,4)=RESIDG(JJ,2)+RESIDC(JJ,2)+RESID(JJ,2)
```

12 CONTINUE

```
DO 13 I=1,18
DO 13 J=1,2
  RESID(I,J)=0.0
  RESIDC(I,J)=0.0
  RESIDG(I,J)=0.0
```

13 CONTINUE

```
DO 14 I=2,3
  PRINT 17, I, (BS(I,J,4),J=1,MS)
  II=I+3
  PRINT 17, II, (ES(II,J,4),J=1,MS)
  PRINT 19
```

14 CONTINUE

15 FORMAT (5X,54HDESIRED GOVT. STOCK EXCEEDS AVAILABLE SUPPLY IN REGI
10N,13)

16 FORMAT (5X,14HDESIRED STOCKS,9F10.0)

17 FORMAT (5X,12,2X,14HENDING STOCKS,9F10.0)

18 FORMAT (5X,14HSURPLUS MILK =,9F12.0)

19 FORMAT (1H0)

20 FORMAT (5X,14HENDING STOCKS,9F10.0)

RETURN

END
SUBROUTINE COMP

...CALCULATE OUTPUT FOR REPORT WRITER

```
COMMON /BLK1/ RMD(59,59),SPOD(59),PRRD(59),TRNSRD(59),TRMC(59),  
1RMS(16,59),PRRD(59),CL3SP(16),CL3DS(59),PRRS(16),CL1SL(59),IACL1  
2S(59),PMM(59,59),OACL1S(59),MMFG(75,42)  
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIGI,IMR,IMRN(11  
1)  
COMMON /BLK5/ CPRCHG(59),BTPSQ(59,4),BLEND(59,6),SE(59),GRBIND(4),  
1GRBCNU(59),IGBEF,CQTSP(59),GSPL1(59),ICMEF  
COMMON /BLK7/ NAD,NOD,MXC,NNODES,IPRI(20),IPRO(20),QCNT1,OP4,KGNET  
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7  
COMMON /BLK9/ NW,NSN,NT,NFN,NNN,NMN,NMM,NMD,NDP,NMS,NTN  
COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(1  
20),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM  
2XC,NNOD,NNNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NQ,NIC,ISUP1,ISUP7,NS  
3PM,NSC,NSK,THCF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISCLCP,CHGCL3,ACST,  
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LBCHK,IPNARC,SCL1CP,T  
5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7  
COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP  
1RC(5),TRANSR(59),OACL1T(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1  
20),HCHG2(16),MXN,IPCEF,CQC2D(59),CQC1D(59),NRPR(59),XCL2P,DTCST,IW  
3ORK3,SFPST,BFPCST,TRMPC,TMCST,RMPC(75),EFRR(59)  
COMMON /BLK12/ ISTATE,MXP,NSA1S,NSMFS,NT1,NT2,NT3,NT4,NPN1,NDP1,NT  
1N1,NT5,NT6,NT7,NT8,NT9,NT10,NT11  
COMMON /BLK13/ IMRG(59),BLDIF(59),CHGCL1(59),OCL1P(59,20),OCL2P(20  
1),OCL3P(20),CL1P(59,5),TITLE,SPOPS(16),GRC1(59),PCL1(59),TCL3(59),  
2PRECC(59),UP(59),UPMRG(20),PRRECH(20),BLNDRM(20),UPAMRG(10),PRECAM  
3(10),BLAMR(20)  
COMMON /BLK14/ IDXB(9,12),UNRS(9),B(9,4),BIMP(4,2,4),C(5,5,4),RPB(1  
5,5),BS(6,3,4),BPC(9),BSE(9),BDE(5),BTRANS(11,5),BBC3P(5),BCHGI(4)  
2,PCHGI(4),NRPB(5),CHGMC(5,5),SIDX(3,4),RSS(9),CF(5),BMKP(5),BRCST(3  
5),BNF(2),BTCST(2,5),ISS,UNUT,UNPC,DGSC(20),IMPHLD,UNS(4),UNGBP(9,  
44),UNFPHC,UNPMC,ECHBT(5),BFPHC
```

DIMENSION W1(11619), W2(890), W3(852), W4(2104)

INTEGER QR,QT,QCNT,QCNT1,CR,PCR,PMR,RMR,RDA,OPEXF,OPCL1,OP1,OP2,OP
13,OP5,OP6,OP7,OP4,BIGI

REAL NFP,IACL1,MPDM,MMFC,MPDM,IPPM,NCL2Q,MCL2Q,MCL3Q,NRPR,NCL1P,
1MFGS,MFGM,IACL1S,MTPSQ,MCL1Q,ICHG,MOCC,MOPC,MPPC,MRTLP,NCL2P,MLEN
2D,MCL1P,MCL1U,MIACL1,MMDM,ICP,NCL3,NCL3P,NTPSQ,MMFG

EQUIVALENCE (RMD(1),W1(1)), (CPRCHG(1),W2(1)), (NCL1P(1),W3(1)),
1(IMRG(1),W4(1))

...(RE)INITIALIZE QUARTERLY SUMMATION VARIABLES
START3=CTIME(T1)

```
DO 1 I=1,NT9  
1 NRPR(I)=0.0  
TMC2=0.0  
SMC2D=0.0  
SUCL1=0.0
```

...(RE)INITIALIZE ANNUAL SUMMATION VARIABLES

```
IF (QT.NE.6) GO TO 3
DO 2 I=1,10
  UPAMR(I)=0.0
  PRECAM(I)=0.0
  BLAMR(I)=0.0
2 BLAMR(I+10)=0.0
3 CONTINUE
DO 4 I=1,NSP
4 SPOPS(I)=0.0
```

...CALCULATE EFFECTIVE RESERVE FOR PROCESSING CENTERS
ADJUST COSTS AND MOVEMENTS ACCORDINGLY

```
DO 5 I=1,NT9
  EFRR(I)=0.0
  IF (CL1SLS(I).NE.0.0) EFRR(I)=(IACL1S(I)+OACL1S(I))/CL1SLS(I)
  TRANSR(I)=TRANSR(I)*(EFRR(I))
  TRNSRD(I)=TRNSRD(I)*(EFRR(I))
  TRMC(I)=TRMC(I)*(EFRR(I))
DO 5 J=1,NSP
  IF ((I-NW).EQ.J) GO TO 5
  TMOP=TMOP+RMMS(J,I)
  XYZ=RMMS(J,I)*(EFRR(I))
  SPOPS(J)=SPOPS(J)+XYZ
  CL3SP(J)=CL3SP(J)+RMMS(J,I)-XYZ
5 CONTINUE
```

...CALCULATE MARKET POOL AND BLEND PRICE

```
DO 9 I=1,NT9
  GRC1(I)=IACL1S(I)+OACL1S(I)
  EFRR1=1.0
  IF (CL1SLS(I).NE.0.0) EFRR1=1.-GRC1(I)/CL1SLS(I)
  IF (I.GT.NW.AND.I.LE.NSA1) GO TO 8
  PCL1(I)=(PRROD(I)+PRRD(I))*(1.0-EFRR1)
  RRC3=(PRROD(I)+PRRD(I))-PCL1(I)
  UCL1=PCL1(I)*NCL1P(I)
  GRC3=CL3DS(I)
  UTP1=0.
  GO TO 7
6 XYZ=(PRROD(I)+PRRD(I)+PRRS(I-NW))*(1.0-EFRR1)
  PCL1(I)=SPOPS(I-NW)+XYZ
  RRC3=(PRROD(I)+PRRD(I)+PRRS(I-NW))-XYZ
  GRC3=CL3SP(I-NW)+CL3DS(I)
  WXYZ=PRRS(I-NW)*(1.0-EFRR1)
  UCL1=(PCL1(I)-(WXYZ+SPOPS(I-NW)))*NCL1P(I)
  UTP1=(CL1SP(I-NW))*(WXYZ+SPOPS(I-NW))
7 XY=GRC3+RRC3
  TCL3(I)=XY
  QCL2=QCC2D(I)
  IF (TCL3(I).GT.QCL2) GO TO 8
  SMC2D=SMC2D+QCL2-TCL3(I)
  QCL2=TCL3(I)
8 TCL3(I)=TCL3(I)-QCL2
  PRREC(I)=XY+PCL1(I)
  UCL1=UTP1+UCL1
  IF (I.LE.NSA1) SUCL1=SUCL1+UCL1
  UP(I)=UCL1+QCL2+NCL3P+TCL3(I)*NCL3P
```

```
9 CONTINUE
CALL MRGE (BLEND,UP,PRREC,IMRG,UPMRG,PRRECM,BLNDMR,UPAMRG,PRECAM,B
1LAMR,NT3,PCL1,BLDIF,NT9)
DO 10 I=1,NT9
  IF (GRC1(I).EQ.0.0) TRMC(I)=0.0
  IF (GRC1(I).NE.0.0) TRMC(I)=(TRMC(I)+TRNSRD(I))/GRC1(I)
10 CONTINUE
DO 12 I=1,NT9
  DO 11 J=1,NT9
11  NRPR(I)=NRPR(I)+TRMC(J)*PMM(J,I)
    NRPR(I)=NRPR(I)+PMCST(I)

    CL1D1=IFIX(QC1D(I)+0.5)
    IF (CL1D1.NE.0.0) GO TO 12
    CL1D1=1.0
12  NRPR(I)=(NRPR(I)*RMGN)/(CL1D1*23.2558)

...WRITE TO DAMPLC
WRITE (OP6) W1,W2,W3,W4,SMC20,SUCL1
IF (IMFGB.NE.0) CALL PREP

...NEW DATA FOR NEXT SOLUTION
TIME7=START7-CTIME(T1)

IF (QCNT.EQ.NQ) GO TO 14
DO 13 I=1,NT9
  IF (ISUP1.NE.1.AND.ISUP7.LT.1) NCL1P(I)=CL1P(I,1)+CHGCL1(I)*FLO
1  AT(QCNT)
  IF (ISUP7.GE.1) NCL1P(I)=QCL1P(I,QCNT+1)
  IF (ISUP1.EQ.1.AND.ISUP7.LT.1) NCL1P(I)=(1.+FLOAT(QCNT)*CHGCL1(
1  I))*CL1P(I,1)
  IF (NCL1P(I).LT.0.0) NCL1P(I)=0.0
  IF (I.LE.NW.OR.I.GT.NSA1) GO TO 13
  IF (ISUP1.NE.1.AND.ISUP7.LT.1) CL1SP(I-NW)=CL1P(I-NW,2)+CHGCL1(
1  I)*FLOAT(QCNT)
  IF (ISUP7.GE.1) CL1SP(I-NW)=QCL1P(I,QCNT+1)
  IF (ISUP1.EQ.1.AND.ISUP7.LT.1) CL1SP(I-NW)=(1.+FLOAT(QCNT)*CHGC
1  L1(I))*CL1P(I-NW,2)
  IF (CL1SP(I-NW).LT.0.0) CL1SP(I-NW)=0.0
13 CONTINUE

...CALCULATE NEW CLASS 2 , CLASS 3 PRICES
XYN=QCST2
IF (IPCEF.EQ.1) XYN=CHGPRC(NYCNT)*QCST2
XCL2P=C2MGN*(NCL2P+XYN)
IF (ISUP1.EQ.1) NCL2P=(1.+FLOAT(QCNT)*CHGCL2)*XBP
IF (ISUP1.EQ.1) NCL3P=(1.+FLOAT(QCNT)*CHGCL3)*XBP3
IF (ISUP1.NE.1.AND.ISUP7.NE.1.AND.ISUP7.NE.3) NCL2P=XBP+CHGCL2*FLO
1AT(QCNT)
IF (ISUP1.NE.1.AND.ISUP7.NE.1.AND.ISUP7.NE.3) NCL3P=XBP3+CHGCL3*FL
1OAT(QCNT)
IF (ISUP7.EQ.1.OR.ISUP7.EQ.3) NCL2P=QCL2P(QCNT+1)
IF (ISUP7.EQ.1.OR.ISUP7.EQ.3) NCL3P=QCL3P(QCNT+1)
IF (NCL2P.LT.0.0) NCL2P=0.0
IF (NCL3P.LT.0.0) NCL3P=0.0
14 CONTINUE

...REINITIALIZE DATA BASE WITH CURRENT QUARTER DATA
SUMXX=0.0
SUMCP=0.0
```

```
DO 17 I=1,NT9
  QSPL1(I)=CQTSP(I)
  IABCD=IFIX(CQTSP(I)+0.5)
  XX=PRREC(I)-FLOAT(IABCD)
  IF (I.LE.NW.OR.I.GT.NSA1) GO TO 15
  X1=IFIX((1.0-SPRC(I-NW))*CQTSP(I)+0.5)
  X2=IFIX(SPRC(I-NW)*CQTSP(I)+0.5)
  SPRC(I-NW)=X2/PRREC(I)
  XX=PRREC(I)-(X1+X2)
15 CONTINUE

  IF (BTPSQ(I,QT-5).GT.0.0) CPRCHG(I)=CPRCHG(I)+XX/BTPSQ(I,QT-5)
  IF (BTPSQ(I,QT-5).EQ.0.0.AND.PRREC(I).GT.0.0) BTPSQ(I,QT-4)=PRR
1 EC(I)

  BLEND(I,5)=0.0
  XPR1=0.0
  XPR=RMPC(I)
  IF (I.GT.NSA1) XPR=RMPC(I+NSP)
  IF (I.LE.NW.OR.I.GT.NSA1) GO TO 16
  XPR=SPRC(I-NW)*RMPC(I+NSP)+(1.-SPRC(I-NW))*RMPC(I)
16 IF (IPDEF.EQ.1) XPR1=CHGPD(QCNT)*XPR-XPR
  IF (BLEND(I,6).GT.XPR1) BLEND(I,5)=BLEND(I,6)-XPR1
  IF (PRREC(I).EQ.0.0.OR.BLEND(I,5).EQ.0.0) CPRCHG(I)=0.0
17 CONTINUE
```

...INCREMENT QUARTER COUNTERS

```
QCNT=QCNT+1
QCNT1=QCNT
QT=QT+1
IF (QT.NE.10) GO TO 18
NYRBC=NYRBC+1
NYCNT=NYCNT+1
QT=6
18 CONTINUE
RETURN
```

```
END
SUBROUTINE MRGE (BLEND,UP,PRREC,IMRG,UPMRG,PRRECM,BLNDMR,UPAMRG,PRECAM,
BLAMR,MSA1,PCL1,BLDIF,NT9)
```

...CALCULATE BLEND PRICE FOR ORDERS

```
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,MMMF,NMFGT,BIGI,IMR,IMRN(11
1)
DIMENSION BLEND(MSA1,6),UP(MSA1),PRREC(MSA1),IMRG(MSA1),UPMRG(
120),PRRECM(20),BLNDMR(20),UPAMRG(10),PRECAM(10),BLAMR(20),PC
2L1(MSA1),BLDIF(MSA1)
```

```
DO 1 I=1,NT9
  BLEND(I,6)=0.0
  IF (PRREC(I).NE.0.0) BLEND(I,6)=UP(I)/PRREC(I)
1 CONTINUE
IF (IMR.LE.0) RETURN
N1=0
N2=0
DO 4 I=1,IMR
  N1=N2+1
  N2=N2+IMRN(I)
  N4=I+10
  UPMRG(I)=0.0
  UPMRG(N4)=0.0
  PRRECM(I)=0.0
  PRRECM(N4)=0.0
```

```
DO 2 J=N1,N2
  N3=IMRG(J)
  WT=0.0
  IF (IMRN(11).GT.0) WT=BLDIF(N3)
  UPMRG(N4)=UPMRG(N4)+WT*PRREC(N3)
  PRREC(N4)=PRREC(N4)+PCL1(N3)
  UPMRG(I)=UPMRG(I)+UP(N3)
2  PRREC(I)=PRREC(I)+PRREC(N3)
  BLNDR(I)=0.0
  BLNDR(N4)=0.0
  IF (PRREC(I).LE.0.0) GO TO 3
  BLNDR(I)=UPMRG(I)/PRREC(I)
  BLNDR(N4)=(PRREC(N4)/PRREC(I))*100.0
3  UPAMRG(I)=UPAMRG(I)+PRREC(N4)
  PRECAM(I)=PRECAM(I)+PRREC(I)
  BLAMR(I)=BLAMR(I)+(BLNDR(I)/4.0)
  BLAMR(N4)=BLAMR(N4)+(BLNDR(N4)/4.0)
DO 4 K=N1,N2
  N3=IMRG(K)
  WT=0.0
  IF (IMRN(11).GT.0) WT=BLDIF(N3)
  BLEND(N3,6)=0.0
  IF (PRREC(I).NE.0.0) BLEND(N3,6)=((UPMRG(I)-UPMRG(N4))/PRREC(
1  I))+WT
4  CONTINUE
  RETURN
```

END
REAL FUNCTION NTPSQ(I,QT)

...SUPPLY FUNCTION(S)

```
COMMON /BLK5/ CPRCHG(59),BTPSQ(59,4),BLEND(59,6),SE(59),GRBIND(4),
1GRBCNU(59),IGBEF,CQTSP(59),QSPL1(59),ICMEF
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7
```

INTEGER QT

```
NTPSQ=0.0
IF (BLEND(I,5).EQ.0.0) RETURN
X1=0.0
IF (IGBEF.EQ.1) X1=FLOAT(NYCNT)*(GRBCNU(I)*GRBIND(QT-5))
B1=BTPSQ(I,QT-5)+CPRCHG(I)*BTPSQ(I,QT-5)
IF (SE(I).GT.0.0.AND.BLEND(I,QT-5).GT.0.0) GO TO 1
NTPSQ=B1+X1
GO TO 2
1 A1=B1/BLEND(I,QT-5)**SE(I)
  NTPSQ=A1*(BLEND(I,5)**SE(I))+X1
2 IF (NTPSQ.LT.0.0) NTPSQ=0.0
  RETURN
```

END
FUNCTION CLID(I,QT)

...CLASS I DEMAND FUNCTION(S)

```
COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SPLB(16,4),DISTB(7
25),DISTP(59),NOR(75),ONAME(75),QSFR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BP1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
```

```
42(59),CHGNC1(59),IC1EF,BC2P(5),CL2Q(59,4),PSE2,IC2EF,BIAC1(59,4),D
SISTD,DCHG1,SSPMI(14,8,8),SPACE3(3594)
COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP
1RC(5),TRANSR(59),OACLIT(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1
20),HCHG2(16),MXN,IPCEF,CQC2D(59),CQC1D(59),NRPR(59),XCL2P,DTCST,IW
3ORK3,SFPCST,DFPCST,TRMPC,TMCST,RMPC(75),EFRR(59)
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7
```

```
REAL MPDRM,MPDPM,IPPM,MMDM,NCL1P,NRPR
INTEGER QT,QCNT
```

```
X1=BIAC1(I,QT-5)*((CHGNC1(I))*FLOAT(NYCNT))
IF (IC1EF.NE.1) X1=0.0
CL1D=A(I,QT-5)+X1
IF (DE(I).NE.0.0) CL1D=(A(I,QT-5)*NRPR(I)**DE(I))+X1
IF (CL1D.LT.0.0) CL1D=0.0
RETURN
```

```
END
FUNCTION CL2D(I,QT)
```

...CLASS II DEMAND FUNCTIONS

```
COMMON /BLK1/ SPMI(61,45),DMMFG(27,10),LMMFG(27,10),MMNC(27,10),PD
1MI(45,45),MPDRM(59),MPDPM(59),IPPM(45),NARC(15),SPLB(16,4),DISTB(7
25),DISTP(59),NOR(75),QNAME(75),QSPR(16),HCHG1(16),MMDM(19),CHGPT(5
3),CHGBT(5),AR,BR,AP,BP1,IBTEF,IPTEF,NSX1,NSPX,A(59,5),DE(59),CHGNC
42(59),CHGNC1(59),IC1EF,BC2P(5),CL2Q(59,4),PSE2,IC2EF,BIAC1(59,4),D
SISTD,DCHG1,SSPMI(14,8,8),SPACE3(3594)
COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP
1RC(5),TRANSR(59),OACLIT(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1
20),HCHG2(16),MXN,IPCEF,CQC2D(59),CQC1D(59),NRPR(59),XCL2P,DTCST,IW
3ORK3,SFPCST,DFPCST,TRMPC,TMCST,RMPC(75),EFRR(59)
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7
```

```
REAL MPDRM,MPDPM,IPPM,MMDM,NCL1P,NRPR
INTEGER QCNT,QT
```

```
X1=CL2Q(I,QT-5)*(CHGNC2(I))*FLOAT(NYCNT)
IF (IC2EF.NE.1) X1=0.0
IF (PSE2.NE.0.0) GO TO 1
CL2D=CL2Q(I,QT-5)+X1
GO TO 2
1 BC2=CL2Q(I,QT-5)/(BC2P(QT-5)**PSE2)
CL2D=BC2*XCL2P**PSE2+X1
2 IF (CL2Q(I,QT-5).EQ.0.0) CL2D=0.0
IF (CL2D.LT.0.0) CL2D=0.0
RETURN
```

```
END
FUNCTION GRADEB(I)
```

...GRADE B MILK SUPPLY

```
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7
COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
2XC,NNOD,NNNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NQ,NIC,ISUP1,ISUP7,NS
3PM,NSC,NSK,IPDEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISC1CP,CHGCL3,ACST,
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LBCHK,IPNARC,SCL1CP,T
```



```
SSMFC, XBP, XBP3, MPDC, TIME2, TIME3, TIME7  
COMMON /BLK14/ IDXB(9, 12), UNRS(9), B(9, 4), BIMP(4, 2, 4), C(5, 5, 4), RPB(  
15, 5), BS(6, 9, 4), BPC(9), BSE(9), BDE(5), BTRANS(11, 5), BBC3P(5), BCHGI(4)  
2, PCHGI(4), NRPB(5), CHGMC(5, 5), SIDX(3, 4), RSS(9), CF(5), BMKP(5), BRCST(  
35), BNF(2), BTCST(2, 5), ISS, UNUT, UNPC, DGSC(20), IMPHLD, UNS(4), UNGBP(9,  
44), UNFPHC, UNPMC, ECHBT(5), BFPHC
```

```
INTEGER QT, QCNT  
REAL NCL3P
```

```
GRADEB=0.0  
IF (NCL3P.EQ.0.0) RETURN  
GBP=BBC3P(QT-5)  
GBP=NCL3P  
IF (IPDEF.EQ.1) GBP=GBP-BPC(I)*(CHGPD(QCNT)-1.0)  
IF (BSE(I).GT.0.0.AND.GBP.GT.0.0) GO TO 1  
GRADEB=B(I, QT-5)  
GO TO 2  
1 GRADEB=B(I, QT-5)/GBP**BSE(I)*GBP**BSE(I)  
2 IF (GRADEB.LT.0.0) GRADEB=0.0  
RETURN
```

```
END  
FUNCTION RIMPC(I, J)
```

...IMPORTS OF MFG MILK PRODUCTS

```
COMMON /BLK8/ NYCNT, IMFGB, OP2, OP3, OP5, OP6, OP7  
COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(  
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM  
2XC, NNOD, NNNE, NNND, ISTPS, NYRBC, NYBASE, QT, QCNT, NG, NIC, ISUP1, ISUP7, NS  
3PM, NSC, NSK, IHCEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL2, ISC1CP, CHGCL3, ACST,  
4PMCHG, RMGN, QCST2, C2MGN, RMR, PCR, PMR, MCR, OPCL1, LBCHK, IPNARC, SCL1CP, T  
SSMFC, XBP, XBP3, MPDC, TIME2, TIME3, TIME7  
COMMON /BLK14/ IDXB(9, 12), UNRS(9), B(9, 4), BIMP(4, 2, 4), C(5, 5, 4), RPB(  
15, 5), BS(6, 9, 4), BPC(9), BSE(9), BDE(5), BTRANS(11, 5), BBC3P(5), BCHGI(4)  
2, PCHGI(4), NRPB(5), CHGMC(5, 5), SIDX(3, 4), RSS(9), CF(5), BMKP(5), BRCST(  
35), BNF(2), BTCST(2, 5), ISS, UNUT, UNPC, DGSC(20), IMPHLD, UNS(4), UNGBP(9,  
44), UNFPHC, UNPMC, ECHBT(5), BFPHC
```

```
INTEGER QT
```

```
IF (IMPHLD.NE.0) GO TO 1  
RIMPC=(BIMP(I, J, QT-5)+BCHGI(I))*(1.0+PCHGI(I))  
GO TO 2  
1 BIMP(I, J, QT-5)=(BIMP(I, J, QT-5)+BCHGI(I))*(1.0+PCHGI(I))  
RIMPC=BIMP(I, J, QT-5)  
2 IF (RIMPC.LT.0.0) RIMPC=0.0  
RETURN
```

```
END  
FUNCTION CMMP(I, J)
```

...CONSUMPTION OF MFG MILK PRODUCTS

```
COMMON /BLK5/ CPRCHG(59), BTPSQ(59, 4), BLEND(59, 6), SE(59), GRBIND(4),  
1GRBCNU(59), IGBEF, CQTSP(59), QSPL1(59), ICMEF  
COMMON /BLK8/ NYCNT, IMFGB, OP2, OP3, OP5, OP6, OP7  
COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(  
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM
```

```
2XC, NNOD, NNNE, NNND, ISTPS, NYRBC, NYBASE, QT, QCNT, NQ, NIC, ISUP1, ISUP7, NS
3PM, NSC, NSK, IHCEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL2, ISC1CP, CHGCL3, ACST,
4PMCHG, RMGN, OCST2, C2MGN, RMR, PCR, PMR, MCR, OPCL1, LBCHK, IPNARC, SCL1CP, T
5SMFC, XBP, XBP3, MPDC, TIME2, TIME3, TIME7
COMMON /BLK14/ IDXB(9,12), UNRS(9), B(9,4), BIMP(4,2,4), C(5,5,4), RPB(
15,5), BS(6,9,4), BPC(9), BSE(9), BDE(5), BTRANS(11,5), BBC3P(5), BCHGI(4)
2, PCHGI(4), NRPB(5), CHGMC(5,5), SIDX(3,4), RSS(9), CF(5), BMKP(5), BRCST(
35), BNF(2), BTCST(2,5), ISS, UNUT, UNPC, DGSC(20), IMPHLD, UNS(4), UNGBP(9,
44), UNFPHC, UNPMC, ECHBT(5), BFPHC
```

```
REAL NRPB
INTEGER QT
```

```
X1=C(I,J,QT-5)*CHGMC(I,J)*FLOAT(NYCNT)
IF (ICMEF.NE.1) X1=0.0
IF (BDE(I).NE.0.0) GO TO 1
CMMP=C(I,J,QT-5)+X1
GO TO 2
1 CMMP=C(I,J,QT-5)/RPB(I,QT-5)**BDE(I)*NRPB(I)**BDE(I)+X1
2 IF (CMMP.LT.0.0) CMMP=0.0
RETURN
```

```
END
FUNCTION UNREG(I)
```

...UNREGULATED MILK SUPPLY

```
COMMON /BLK1/ COMFG(9), COIMP(4,2), COBS(2,3,9), COC(5,5), COBNS(2,9),
1NARC(15), BMMD1(31,15), BPOP(31), BPRR(15), BMMD2(58,10), POPB(58), PRRB
2(10), RESID(18,2), RESIDC(18,2), RESIDG(18,2), CHEESE(9), WK2(9), WK1(9)
3, COB(9), COUREG(9), COUC1S(9), COUGBP(9), COUC1P(9), COUNBP(9), COUROD(9
4), COUVP(9), COUNRP(9), COES(6,9), TRCS1, TRCS2, SPACE(4269), SPACE1(5376
5)
COMMON /BLK8/ NYCNT, IMFGB, OP2, OP3, OP5, OP6, OP7
COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM
2XC, NNOD, NNNE, NNND, ISTPS, NYRBC, NYBASE, QT, QCNT, NQ, NIC, ISUP1, ISUP7, NS
3PM, NSC, NSK, IHCEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL2, ISC1CP, CHGCL3, ACST,
4PMCHG, RMGN, OCST2, C2MGN, RMR, PCR, PMR, MCR, OPCL1, LBCHK, IPNARC, SCL1CP, T
5SMFC, XBP, XBP3, MPDC, TIME2, TIME3, TIME7
COMMON /BLK14/ IDXB(9,12), UNRS(9), B(9,4), BIMP(4,2,4), C(5,5,4), RPB(
15,5), BS(6,9,4), BPC(9), BSE(9), BDE(5), BTRANS(11,5), BBC3P(5), BCHGI(4)
2, PCHGI(4), NRPB(5), CHGMC(5,5), SIDX(3,4), RSS(9), CF(5), BMKP(5), BRCST(
35), BNF(2), BTCST(2,5), ISS, UNUT, UNPC, DGSC(20), IMPHLD, UNS(4), UNGBP(9,
44), UNFPHC, UNPMC, ECHBT(5), BFPHC
```

```
INTEGER QT, QCNT
```

```
IF (COUGBP(I).EQ.0.0) RETURN
UP=COUGBP(I)
IF (IPDEF.EQ.1) UP=UP-UNPC*(CHGPD(QCNT)-1.0)
IF (BSE(I).GT.0.0.AND.UNGBP(I,QT-5).GT.0.0) GO TO 1
UNREG=UNRS(I)*UNS(QT-5)
GO TO 2
1 UNREG=(UNRS(I)*UNS(QT-5))/UNGBP(I,QT-5)**BSE(I)*UP**BSE(I)
2 IF (UNREG.LT.0.0) UNREG=0.0
```

```
RETURN
```

```
END
```

SUBROUTINE RESET

...STORE OPTIMAL NETWORK FLOWS FOR USE BY COMP

```
COMMON /BLK1/ RMMD(59,59),SPOPD(59),PRROD(59),TRNSRD(59),TRMC(59),
1RMM5(16,59),PRRD(59),CL3SP(16),CL3DS(59),PRRS(16),CL1SLS(59),IACL1
2S(59),PMM(59,59),OACL1S(59),MMFG(75,42)
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIGI,IMR,IMRN(11
1)
COMMON /BLK9/ NW,NSN,NT,NPN,NNN,NMN,NMM,NMD,NDP,NMS,NTN
COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
2XC,NNOD,NNNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NQ,NIC,ISUP1,ISUP7,NS
3PM,NSC,NSK,IHCEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISC1CP,CHGCL3,ACST,
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LBCHK,IPNARC,SCL1CP,T
5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7
COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP
1RC(5),TRANSR(59),OACL1T(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1
20),HCHG2(16),MXN,IPCEF,CQC2D(59),CQC1D(59),NRPR(59),XCL2P,DTCST,IW
3ORK3,SFPCST,DFPCST,TRMPC,TMCST,RMPC(75),EFRR(59)
COMMON /BLK8/ NYCNT,IMFGB,OP2,OP3,OP5,OP6,OP7
COMMON /BLK12/ ISTATE,MXP,NSA1S,NSMFS,NT1,NT2,NT3,NT4,NPN1,NDP1,NT
1N1,NT5,NT6,NT7,NT8,NT9,NT10,NT11
```

```
REAL MFCM,MFGS,IACL1S,NCL1P,MMFG,NRPR,NCL3P
INTEGER RDA,BIGI,OP1,OP2,OP3,OP5,OP6,OP7,QT,QCNT,RMR,PCR,PMR,OPCL1
```

```
START4=CTIME(T1)
```

```
DO 3 I=1,NT1
  IF (I.GT.NT3) GO TO 2
  DO 1 J=1,NT3
    RMMD(I,J)=0.
    PMM(I,J)=0.
    IF (I.GT.NSP) GO TO 1
    RMMS(I,J)=0.
```

```
1 CONTINUE
  SPOPD(I)=0.
  PRROD(I)=0.
  TRNSRD(I)=0.
  TRANSR(I)=0.0
  OACL1T(I)=0.0
  PMCST(I)=0.0
  TRMC(I)=0.
  PRRD(I)=0.
  CL3DS(I)=0.
  OACL1S(I)=0.
  IACL1S(I)=0.
  CL1SLS(I)=0.
  IF (I.GT.NSP) GO TO 2
  PRRS(I)=0.
  CL3SP(I)=0.0
```

```
2 CONTINUE
  DO 3 J=1,NT10
3 MMFG(I,J)=0.0
  TMCST=0.0
  RETURN
```

```
ENTRY RESET1
IF (RDA(1).LE.NSA1.AND.RDA(2).LE.NT2) RETURN
```

```
IF (RDA(1).GT.NT.AND.RDA(1).LE.NT1.AND.RDA(2).LE.NSN) RETURN  
IF (RDA(2).GT.NPN) GO TO 13  
IF (RDA(1).LE.NSA1) GO TO 5  
IF (RDA(1).GT.NT.AND.RDA(1).LT.NT1) GO TO 4  
IF (RDA(1).GT.NSA1.AND.RDA(1).LE.NT) GO TO 7  
IF (RDA(1).GT.NT1.AND.RDA(1).LE.NSN) GO TO 12
```

...FLOWS FROM DIRECT SHIPPED NODES TO PROCESSING NODES

```
4 K1=RDA(1)-NSP  
GO TO 6  
5 K1=RDA(1)  
6 K2=RDA(2)-NSN  
RMD(K1,K2)=FLOAT(RDA(4))  
SOPD(K1)=SOPD(K1)+FLOAT(RDA(4))  
PRRD(K2)=PRRD(K2)+FLOAT(RDA(4))  
TRMC(K2)=TRMC(K2)+FLOAT(RDA(4))*FLOAT(RDA(3))  
RETURN
```

...FLOWS FROM SUPPLY PLANT NODES TO PROCESSING NODES

```
7 K1=RDA(1)  
K2=RDA(2)-NSN  
K3=K1-NSA1  
K4=RDA(1)-NSP  
IF (K2.NE.K4) GO TO 8  
TRMC(K2)=TRMC(K2)+FLOAT(RDA(3))*FLOAT(RDA(4))  
PRRS(K3)=PRRS(K3)+FLOAT(RDA(4))  
GO TO 11  
8 WXYZ=FLOAT(RDA(3))  
XIC=0.0  
IF (CINSP(K2).EQ.0.0) GO TO 10  
XIC=CINSP(K2)  
IC1=ILI(K2)  
DO 9 I=1, MXN  
9 IF (ICST(IC1, I).EQ.K4) XIC=0.0  
10 TRCST=WXYZ-NCLIP(K4)-HCHG2(K3)-XIC  
TRNSR(K2)=TRNSR(K2)+FLOAT(RDA(4))*TRCST  
TRNSRD(K2)=TRNSRD(K2)+WXYZ*FLOAT(RDA(4))  
11 RMMS(K3,K2)=FLOAT(RDA(4))  
RETURN
```

...FLOWS FROM DUMMY SUPPLY NODES TO PROCESSING NODES

```
12 K1=RDA(1)-NT1  
K2=RDA(2)-NSN  
RMD(K1,K2)=RMD(K1,K2)+FLOAT(RDA(4))  
PRRD(K2)=PRRD(K2)+FLOAT(RDA(4))  
TRMC(K2)=TRMC(K2)+FLOAT(RDA(4))*FLOAT(RDA(3))  
RETURN  
13 IF (RDA(2).EQ.NMS) RETURN  
IF (RDA(2).GT.NMD) GO TO 19  
K2=RDA(2)-NPN  
IF (RDA(2).GT.NPN1.AND.RDA(2).LT.NMN) K2=K2+2  
IH1=RDA(1)  
IF (RDA(2).EQ.NMN) K2=NMD-NMN  
IF (RDA(2).GT.NMN) K2=RDA(2)-NMN  
IF (RDA(1).LE.NSA1) GO TO 15  
IF (RDA(1).GT.NT.AND.RDA(1).LE.NT1) GO TO 14  
IF (RDA(1).GT.NSA1.AND.RDA(1).LE.NT) GO TO 17
```

```
IF (RDA(1).GT.NT1.AND.RDA(1).LE.NSN) GO TO 18
...FLOWS FROM DIRECT SHIPPED NODES TO MANUFACTURING NODES
14 K1=RDA(1)-NSP
   GO TO 16
15 K1=RDA(1)
16 CL3DS(K1)=CL3DS(K1)+FLOAT(RDA(4))
   MMFG(IH1,K2)=MMFG(IH1,K2)+FLOAT(RDA(4))
   TMCST=TMCST+(FLOAT(RDA(3))-IFIX(NCL3P+0.5))*FLOAT(RDA(4))
   RETURN
...FLOWS FROM SUPPLY PLANT NODES TO MANUFACTURING NODES
17 K1=RDA(1)-NSP
   CL3SP(K1-NW)=CL3SP(K1-NW)+FLOAT(RDA(4))
   MMFG(IH1,K2)=MMFG(IH1,K2)+FLOAT(RDA(4))
   TMCST=TMCST+(FLOAT(RDA(3))-IFIX(NCL3P+0.5))*FLOAT(RDA(4))
   RETURN
...FLOWS FROM DUMMY SUPPLY NODES TO MANUFACTURING NODES
18 K1=RDA(1)-NT1
   K11=K1
   IF (RDA(1).GT.NT2) K11=RDA(1)-NT3
   CL3DS(K1)=CL3DS(K1)+FLOAT(RDA(4))
   MMFG(K11,K2)=MMFG(K11,K2)+FLOAT(RDA(4))
   TMCST=TMCST+(FLOAT(RDA(3))-IFIX(NCL3P+0.5))*FLOAT(RDA(4))
   RETURN
19 IF (RDA(2).GT.NDP1) GO TO 20
...FLOWS FROM PROCESSING NODES TO DUMMY PROCESSING NODES
   K1=RDA(1)-NSN
   CL1SLS(K1)=FLOAT(RDA(4))
   RETURN
...FLOWS FROM DUMMY PROCESSING NODES TO FINAL DEMAND NODES
20 K1=RDA(1)-NMD
   K2=RDA(2)-NDP1
   IF (K1.NE.K2) GO TO 21
   AK=FLOAT(RDA(4))/(1.0+CL1RR(K2))
   IACL1S(K1)=AK
   PMM(K1,K2)=AK
   XRT=CHGPRC(NYCNT)*OCST
   PMCST(K2)=PMCST(K2)+(FLOAT(RDA(3))+XRT)*AK
   RETURN
21 AK=FLOAT(RDA(4))/(1.0+CL1RR(K2))
   PMM(K1,K2)=AK
   OACL1S(K1)=OACL1S(K1)+AK
   XIC=0.0
   IF (K2.GT.NSA1) GO TO 23
   IF (CINSP(K2).EQ.0.0) GO TO 23
   XIC=CINSP(K2)
   IC2=ILI(K2)
   DO 22 I=1, MXN
22 IF (ICST(IC2,I).EQ.K1) XIC=0.0
23 XRT=CHGPRC(NYCNT)*OCST
   XRS=CHGPRC(NYCNT)*PRCST(K1)
```

```
TRCST=FLOAT(RDA(3))-XRS-XIC
OACL1T(K1)=OACL1T(K1)+TRCST*AK
PMCST(K2)=PMCST(K2)+(FLOAT(RDA(3))+XRT)*AK
RETURN
```

```
END
SUBROUTINE RESETB
```

...STORES NETWORK FLOWS

```
COMMON /BLK1/ CQMG(9),CQIMP(4,2),CQBS(2,3,9),CQC(5,5),CQBNS(2,9),
1NARC(15),BMMD1(31,15),BPOP(31),BPRR(15),BMMD2(58,10),POPB(58),PRRB
2(10),RESID(18,2),RESIDC(18,2),RESIDG(18,2),CHEESE(9),WK2(9),WK1(9)
3,CQB(9),CQUREG(9),CQUC1S(9),CQUGBP(9),CQUC1P(9),CQUNBP(9),CQURD(9)
4,CQUUP(9),CQUNRP(9),CQES(6,9),TRCS1,TRCS2,CINU(9,3),BINUV(2,9),SP
5ACE(4224),SPACE1(5376)
```

```
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIGI,IMR,IMRN(11
1)
```

```
COMMON /BLK7/ NAD,NOD,MXC,NNODES,IPRI(20),IPRO(20),QCNT1,OP4,KGNET
COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
2XC,NNOD,NNNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NQ,NIC,ISUP1,ISUP7,NS
3PM,NSC,NSK,IHCEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL2,ISC1CP,CHGCL3,ACST,
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LECHK,IPNARC,SCL1CP,T
5SMFC,XBP,XBP3,MPDC,TIME2,TIME3,TIME7
```

```
COMMON /BLK14/ IDXB(9,12),UNRS(9),B(9,4),BIMP(4,2,4),C(5,5,4),RPB(
15,5),BS(6,9,4),BPC(9),BSE(9),BDE(5),BTRANS(11,5),BBC3P(5),BCHGI(4)
2,PCHGI(4),NRPB(5),CHGMC(5,5),SIDX(3,4),RSS(9),CF(5),BMKP(5),BRCST(
35),BNF(2),BTCST(2,5),ISS,UNUT,UNPC,DGSC(20),IMPHLD,UNS(4),UNGBP(9,
44),UNFPHC,UNPMC,ECHBT(5),BFPHC
```

```
COMMON /BLK15/ MS,MIMP,MC,M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M
113,M14,M15,M16,M17,M18,M19,M20,M21,M22,M23,M24,M25,M26,M27,M28,M29
2,M30,M31,M32,M33,M34,M35,M36,M37,M38
```

```
INTEGER QR,QT,QCNT,QCNT1,CR,PCR,PMR,RMR,RDA,OPEXF,OPCL1,OP1,OP2,OP
13,OP5,OP6,OP7,OP4,BIGI
```

```
REAL NFP,IACL1,MPDRM,MMFC,MPDPM,IPPM,NCL2Q,MCL2Q,MCL3Q,NRPR,NCL1P,
1MFGS,MFGM,IACL1S,MTPSQ,MCL1G,ICHG,MOCC,MOFC,MPPC,MRTLP,NCL2P,MBLEN
2D,MCL1P,MCL1U,MIACL1,MMDM,ICP,NCL3,NCL3P,NTPSQ,MMFG
```

```
IF (KGNET.NE.1) GO TO 4
```

```
DO 1 J=1,15
  BPRR(J)=0.0
DO 1 I=1,31
  BPOP(I)=0.0
1 BMMD1(I,J)=0.0
DO 2 I=1,2
DO 2 J=1,18
  RESID(J,I)=0.0
  RESIDC(J,I)=0.0
2 RESIDG(J,I)=0.0
  TRCS1=0.0
DO 3 L=1,3
DO 3 J=1,9
  CINU(J,L)=0.0
3 CONTINUE
```

```
RETURN
```

```
4 DO 5 I=1,10
  PRRB(I)=0.0
  DO 5 J=1,58
    POPB(J)=0.0
5 BMD2(J,I)=0.0
  TRCS2=0.0
  DO 6 J=1,9
  DO 6 I=1,2
    BNINU(I,J)=0.0
6 CONTINUE
```

RETURN

***** STAGE 1 *****

```
ENTRY RESTB1
IF (RDA(2).GE.M17) RETURN
IF (RDA(1).LE.M5.AND.RDA(2).GT.M8) GO TO 7
IF (RDA(1).GT.M5.AND.RDA(1).LE.M4) GO TO 11
IF (RDA(1).GT.M4.AND.RDA(1).LE.M6) GO TO 12
IF (RDA(1).GT.M6.AND.RDA(1).LE.M8) GO TO 15
```

...FLOWS FROM MILK SUPPLY TO DEMANDS

```
7 K1=RDA(1)
  K2=RDA(2)-M8
  IF (RDA(2).EQ.M15) GO TO 10
  BMM1=FLOAT(RDA(4))
  IF (RDA(2).LT.M11.OR.RDA(2).GT.M12) GO TO 8
  BMM1=BMM1/(1.0+SIDX(1,QT-5))
  CINU(K1,1)=CINU(K1,1)+FLOAT(RDA(4))-BMM1
8 BMD1(K1,K2)=BMM1
  BPOP(K1)=BPOP(K1)+FLOAT(RDA(4))
  BPRR(K2)=BPRR(K2)+FLOAT(RDA(4))
  TRCS1=TRCS1+FLOAT(RDA(3))*BMM1
  IF (RDA(2).GE.M11.AND.RDA(2).LE.M12) GO TO 9
```

RETURN

```
9 CHEESE(K1)=CHEESE(K1)+FLOAT(RDA(4))
```

RETURN

```
10 RESID(K1,KGNET)=RESID(K1,KGNET)+FLOAT(RDA(4))
```

RETURN

...IMPORTS TO DEMANDS

```
11 K1=RDA(1)
  K2=RDA(2)-M8
  BMD1(K1,K2)=FLOAT(RDA(4))
  BPOP(K1)=BPOP(K1)+FLOAT(RDA(4))
  BPRR(K2)=BPRR(K2)+FLOAT(RDA(4))
  TRCS1=TRCS1+FLOAT(RDA(3))*RDA(4)
```

RETURN

...COMM. STOCKS TO DEMANDS

```
12 K1=RDA(1)
   K2=RDA(2)-M8
   IF (RDA(2).EQ.M15) GO TO 14
   BMM1=FLOAT(RDA(4))
   IF (RDA(2).LT.M11.OR.RDA(2).GT.M12) GO TO 13
   BMM1=BMM1/(1.0+SIDX(1,QT-5))
   CINU(K1-M4,2)=CINU(K1-M4,2)+FLOAT(RDA(4))-BMM1
13 BMD1(K1,K2)=BMM1
   BPOP(K1)=BPOP(K1)+FLOAT(RDA(4))
   BPRR(K2)=BPRR(K2)+FLOAT(RDA(4))
   TRCS1=TRCS1+FLOAT(RDA(3))*BMM1
```

RETURN

```
14 K1=K1-M4
   RESIDC(K1,KGNET)=RESIDC(K1,KGNET)+FLOAT(RDA(4))
```

RETURN

...GOUT. STOCKS TO DEMANDS

```
15 K1=RDA(1)
   K2=RDA(2)-M8
   IF (RDA(2).EQ.M15) GO TO 17
   BMM1=FLOAT(RDA(4))
   IF (RDA(2).LT.M11.OR.RDA(2).GT.M12) GO TO 16
   BMM1=BMM1/(1.0+SIDX(1,QT-5))
   CINU(K1-M6,3)=CINU(K1-M6,3)+FLOAT(RDA(4))-BMM1
16 BMD1(K1,K2)=BMM1
   BPOP(K1)=BPOP(K1)+FLOAT(RDA(4))
   BPRR(K2)=BPRR(K2)+FLOAT(RDA(4))
   TRCS1=TRCS1+FLOAT(RDA(3))*BMM1
```

RETURN

```
17 K1=K1-M6
   RESIDG(K1,KGNET)=RESIDG(K1,KGNET)+FLOAT(RDA(4))
```

RETURN

***** STAGE 2 *****

```
ENTRY RESTB2
IF (RDA(2).GE.M38) RETURN
IF (RDA(1).LE.M23) GO TO 18
IF (RDA(1).GT.M23.AND.RDA(1).LT.M36) GO TO 21
```

...BUTTER SUPPLIES TO BUTTER DEMANDS

```
18 K1=RDA(1)
   K2=RDA(2)-M31
   IF (RDA(2).EQ.M36) GO TO 20
   BMM2=FLOAT(RDA(4))
   IF (RDA(1).GE.M18.AND.RDA(1).LE.M19) GO TO 19
   BMM2=BMM2/(1.0+SIDX(2,QT-5))
   KK=K1
   IF (K1.GE.M20) KK=K1-M19
   IF (K1.GE.M22) KK=K1-M21
   BNINU(1,KK)=BNINU(1,KK)+FLOAT(RDA(4))-BMM2
19 BMD2(K1,K2)=BMM2
```



```
POPB(K1)=POPB(K1)+FLOAT(RDA(4))
PRRB(K2)=PRRB(K2)+FLOAT(RDA(4))
TRCS2=TRCS2+FLOAT(RDA(3))*BMM2
```

```
RETURN
```

```
20 IF (RDA(1).LE.M5) RESID(K1,KGNET)=RESID(K1,KGNET)+FLOAT(RDA(4))
   IF (RDA(1).GT.M19.AND.RDA(1).LE.M21) RESIDC(K1-M19,KGNET)=RESIDC(K
11-M19,KGNET)+FLOAT(RDA(4))
   IF (RDA(1).GT.M21) RESIDG(K1-M21,KGNET)=RESIDG(K1-M21,KGNET)+FLOA
1(RDA(4))
```

```
RETURN
```

```
...NFDN SUPPLIES TO DEMANDS
```

```
21 K1=RDA(1)
   K2=RDA(2)-M31
   IF (RDA(2).EQ.M36) GO TO 23
   BMM2=FLOAT(RDA(4))
   IF (RDA(1).GE.M26.AND.RDA(1).LE.M27) GO TO 22
   BMM2=BMM2/(1.0+SIDX(3,GT-5))
   KK=K1-M23
   IF (K1.GE.M28) KK=K1-M27
   IF (K1.GE.M30) KK=K1-M29
   BNINU(2, KK)=BNINU(2, KK)+FLOAT(RDA(4))-BMM2
```

```
22 BMM2(K1, K2)=BMM2
   POPB(K1)=POPB(K1)+FLOAT(RDA(4))
   PRRB(K2)=PRRB(K2)+FLOAT(RDA(4))
   TRCS2=TRCS2+FLOAT(RDA(3))*BMM2
```

```
RETURN
```

```
23 IF (RDA(1).LE.M25) RESID(K1-M21,KGNET)=RESID(K1-M21,KGNET)+FLOAT(R
1DA(4))
   IF (RDA(1).GT.M27.AND.RDA(1).LE.M29) RESIDC(K1-M27+MS,KGNET)=RESID
1C(K1-M27+MS,KGNET)+FLOAT(RDA(4))
   IF (RDA(1).GT.M29) RESIDG(K1-M27,KGNET)=RESIDG(K1-M27,KGNET)+FLOA
1(RDA(4))
```

```
RETURN
```

```
END
FUNCTION CTIME(T1)
```

```
...CTIME ROUTINE FOR 6000, 7000 SERIES CDC
```

```
CALL SECOND (T1)
CTIME=-T1
RETURN
```

```
END
SUBROUTINE GNETA
```

```
...SOLVE CAPACITATED TRANSSHIPMENT PROBLEM
```

```
*****
BASED ON GNET, COPYRIGHT 1975 BY
GORDON H. BRADLEY, NAVAL POSTGRADUATE SCHOOL, MONTEREY, CA 93940
GERALD G. BROWN, NAVAL POSTGRADUATE SCHOOL, MONTEREY, CA 93940
```

GLENN W. GRAVES, UNIVERSITY OF CALIFORNIA, LOS ANGELES, CA 90024
FOR INFORMATION CONTACT WESTERN MANAGEMENT SCIENCE INSTITUTE,
GRADUATE SCHOOL OF MANAGEMENT, UCLA
LOS ANGELES, CA, 90024

DAMPCLC

DAMPCLC, listed on the following pages, does calculations used primarily by DRW1. Annual sums and various percentages and absolute changes are computed. The subroutine COMP does most calculations. The subroutine TCOST computes aggregate transportation and production costs. The subroutine OSET prepares a table for DRW1 and is identical to OSET in DAMPSIN. Though they have the same name, COMP in DAMPCLC is different from COMP in DAMPSLV, both are descendants of a subroutine called COMP used in FMMOPS.

PROGRAM DAMPCLC (INPUT, OUTPUT, TAPE9, TAPE12, TAPE13, TAPE17, TAPE18, TAPE19, TAPE43)

.... DAMPCLC IS THE THIRD PROGRAM OF DAMPS
DAMPCLC READS DATA FROM DAMPSIN AND RESULTS FROM DAMPSLN
AND MAKES CALCULATIONS FOR THE REPORT WRITERS DRW1 AND
DRW2. DAMPCLC WAS WRITTEN BY D.R. MARTELLA, J.E. PRATT,
AND A.M. NOVAKOVIC.

COMMON /SPACE1(9500), ADLEND(59), ACEXP(2), ACHGC1(45), ACHGC2(45), ACHG
103(45), ACHGPR(45), ACHIR(45), ACL1P(59), ACL1UT(59), ACL2D(59, 2), ACL3
20(59, 2), AFRPS(59), AGRC1(45), AIAC1S(59, 2), AMBLND(2), AMC1P(2), AMC1U(5
2), AMC2P(2), AMC3P(2), AMRTL(2), AMUCP(2), ANFP(45), ADAC1T(45), APCL1(5
49, 2), APRREC(59, 2), ARODC(45), ARPR(59), ASCL2D(2), ASCL3Q(2), ASFRPS(2)
5, ASIAC1(2), ASMFGU(2), ASNFP(2), ASOAC1(2), ASPCL1(2), ASPRRC(2), ASRODC
6(2), ASTRNS(2), ATCCWT(59), ATMMC(53), ATRANS(45), ATRMC(45), AUMCP(53),
7AUTCP(59), AUP(2), AWRPS(2), BCEXP(5), BC3P(5), BLDIF(59), BMBLEN(5), BM
8CL1P(5), BMCL1U(5), BMRTLP(5), BNCL2P(5), BNCL3P(5), BP(20), BPS(20), BSM
9FCU(5), BSMCP(5), BSNFP(5), BSOAC1(5), BSTRNS(5), BSUFRP(5), BSUP(5), BS
*UROD(5), BWRPS(5), CHGCL1(59), CHGCL0(45), CHGCL2(45), CHGCL3(45), CHGI
*AC(45), CHGPR(45), CL1P(59, 5), CL1Q(59, 4), CL1UT(59), FRPS(59), IACL1(5
*9, 4), ICR(10), IDICST(5), IMRG(59), MCL1Q(5), MCL2Q(5), MCL3Q(5), MIACL1(
*4), MR(5), MTPSQ(5), NCL2Q(59, 4), NFP(45), NYID(5), OMC1P(75), PCL2(59), Q
*CL1P(59, 20), QCL2P(20), QCL3P(20), QR(20), RODC(45), SAD(14, 4), SERCST(5
*), SRMPC(5), TCCWT(59), TMMC(53), TPSQ(59, 4), UMCP(53), UTCP(59), IRR(59)
*, BLAMR(20), BLNDMR(20), UP(59), UPAMRG(10), UPMRG(20), PRECAM(10), PRREC
*M(20), PRREC(59), SPOPS(15), TCL3(59), PCL1(59), GRC1(59), ARMMD(59, 59),
*ARMM5(15, 59), AMMFG(75, 42), APMM(59, 59)

COMMON /BLK1/ WORK2(11619)

COMMON /BLK2/ WORK3(24)

COMMON /BLK5/ WORK4(890)

COMMON /BLK7/ WORK5(45)

COMMON /BLK8/ NYCNT

COMMON /BLK9/ WORK6(11)

COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM
2XC, NNDD, NNNE, NNND, ISTPS, NYRBC, NYBASE, QT, QCNT, NQ, NIC, ISUP1, ISUP7, NS
3PM, NSC, NSK, IMDEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL4, ISC1P, CHGCL5, ACST,
4PMCHG, RMGN, OCST2, C2MGN, RMR, PCR, PMR, MCR, OPCL1, LBCHK, IPNARC, SCL1CP, T
5SMFC, XBP, XBP3, MPDC, OP1, OP2, OP6, OP7, OP8

COMMON /BLK11/ WORK8(852), WORK10(10), SMC2Q, SUCL1

COMMON /BLK12/ ISTATE, MXP, NSAL9, NSMFS, NT1, NT2, NT3, NT4, NPN1, NDP1, NT
1N1, NT5, NT6, NT7, NT8, NT9, NT10, NT11

DIMENSION WORK1(14992), WORK7(407), WORK9(18)

INTEGER QR, QT, QCNT, QCNT1, CR, PCR, PMR, RMR, RDA, OPEXF, OPCL1, OP1, OP2, OP
13, OPS, OP6, OP7, OP8, OP4, DIGI

REAL NFP, IACL1, MPDRM, MMFC, MPBPM, IPPM, NCL2Q, MCL2Q, MCL3Q, NRPR, NCL1P,
1MFGS, MFGM, IACL1S, MTPSQ, MCL1Q, ICHG, MOCC, MOPC, MPPC, MRTLP, NCL2P, MLEN
2D, MCL1P, MCL1U, MIACL1, MMDM, ICP, NCL3, NCL3P, NTPSQ, MMFG

EQUIVALENCE (SPACE1(1), WORK1(1)), (MMFC(1), WORK7(1)), (ISTATE, WORK
19(1))

...SET LOGICAL FILE PARAMETERS

```
IP1=9
IP2=13
OP1=17
OP2=18
OP6=19
OP7=12
OP8=43

... IP1=INPUT FROM DAMPSIN
... IP2=QUARTERLY INPUT FROM DAMPSLU
... OP1=OUTPUT TO GRADE A REPORT WRITER - DRW1
... OP2=SCRATCH FILE FOR TCOST COMMON
... OP6=SCRATCH FILE FOR TCOST
... OP7=TCOST MILEAGE MATRIX FROM GAPSIN
... OP8=OUTPUT TO GRADE B REPORT WRITER - DRW2

READ (IP1) WORK1,WORK3,WORK4,WORK5, NYCNT, WORK6,WORK7,WORK8,WORK10,
1WORK9

....MAIN LOOP
DO 1 IDID=1,NO
READ (IP2) WORK2,WORK4,WORK8, IMRG, BLDIF, CHGCL1, QCL1P, QCL2P, QCL3
1 P, CL1P, TITLE, SPOPS, GRC1, PCL1, TCL3, PRREC, UP, UPMRG, PRREC1, BLNDR,
2 UPAMRG, PRECAM, BLAMR, SMC20, SVCL1

CALL COMP

1 CONTINUE

....WRITE DATA FOR COMPARATIVE REPORTS

J2=0
NIC2=NIC+1
DO 2 I=1, NIC2
J1=J2
IF (I.EQ.1) J2=J1+750
IF (I.GT.1) J2=J1+(30*NT9)
J1=J1+1
IF (I.GT.1.AND.NIC.EQ.0) STOP 2155
2 WRITE (OP1) (WORK1(IJK), IJK=J1, J2)

STOP

END
SUBROUTINE COMP

...CALCULATE OUTPUT FOR REPORT WRITER

DIMENSION WORK3(6662)

WORK3=SIZE OF COMMON IN TCOST

COMMON SPACE1(9600), ABLEND(59), ACEXP(2), ACHGC1(45), ACHGC2(45), ACHG
1C3(45), ACHGPR(45), ACHIAC(45), ACL1P(59), ACL1UT(59), ACL2D(59, 2), ACL3
2Q(59, 2), AFRPS(59), AGRC1(45), AIAC1S(59, 2), AMELND(2), AMC1P(2), AMC1U(
32), AMC2P(2), AMC3P(2), ANRTL(2), AMUCP(2), ANFP(45), AOAC1T(45), APCL1(5
49, 2), APRREC(59, 2), ARODC(45), ARPR(59), ASCL2D(2), ASCL3Q(2), ASFRPS(2)
5, ASIAC1(2), ASMFGU(2), ASNFP(2), ASDAC1(2), ASPCL1(2), ASPRRC(2), ASRODC
```

```
6(2), ASTRNS(2), ATCCWT(59), ATMMC(53), ATRANS(45), ATRMC(45), AUMCF(53),
7AUTCP(59), AUP(2), AWFRPS(2), BCEXP(5), BC3P(5), BLDIF(59), BMLEN(5), BM
8CL1P(5), BMCL1U(5), BMRTLP(5), BNCL2P(5), BNCL3P(5), BP(20), BPS(20), BSM
9FGU(5), BSMUCP(5), BSMFP(5), BSOACL(5), BSTRNS(5), BSUFRP(5), BSUP(5), BS
*URON(5), BWFRPS(5), CHGCL1(59), CHGCL0(45), CHGCL2(45), CHGCL3(45), CHGI
*AC(45), CHGPRR(45), CL1P(59,5), CL1Q(59,4), CL1UT(59), FRPS(59), IACL1(5
*9,4), ICR(10), IDICST(5), IMRC(59), MCL1Q(5), MCL2Q(5), MCL3Q(5), MIACL1(
*4), MR(5), MTPSQ(5), NCL2Q(59,4), NFP(45), NYID(5), OMC1P(75), PCL2(59), Q
*CL1P(59,20), QCL2P(20), QCL3P(20), QR(20), RODC(45), SAD(14,4), SERCST(5
*), SRMPC(5), TCCWT(59), TMMC(53), TPSQ(59,4), UMCP(53), UTCP(59), IRR(59)
*, ELAMR(20), BLNDMR(20), UP(59), UPAMRG(10), UPMRG(20), PRECAM(10), PPREC
*M(20), PPREC(59), SPOPS(18), TCL3(59), PCL1(59), GRC1(59), ARMMD(59,59),
*ARMMS(16,59), AMMFG(75,42), APMM(59,59)
COMMON /BLK1/ RMMD(59,59), SPOPB(59), PRROD(59), TRANSR(59), TRMC(59),
1RMTS(16,59), PRRD(59), CL3SP(16), CL3DS(59), PRRS(16), CL1SLS(59), IACL1
25(59), PMM(59,59), OACL1S(59), MMFG(75,42)
COMMON /BLK2/ RDA(5), NSP, NSA1, NSMF, NDM, NMMF, NMFGT, BIGI, IMR, IMRN(11
1)
COMMON /BLK5/ CPRCHG(59), BTSPQ(59,4), BLEND(59,6), SE(59), GRBIND(4),
1GRBCNU(59), IGDEF, COTSP(59), QSPL1(59)
COMMON /BLK7/ NAD, NOD, MXC, NNODES, IPRI(20), IPRO(20), GCNT1
COMMON /BLK8/ NYCNT
COMMON /BLK9/ NW, NSN, NT, NPN, NNN, NMN, NMM, NMD, NDP, NMS, NTN
COMMON /BLK10/ MMFC(27), CL1CP(59), CL1SP(16), IR(61), CHGHC(5), CHGPD(
120), ERCST(75), SPRC(16), IDMG1(26), IDMG2(19), SMFC(40), SSMFC, SMMFC, SM
2XC, NNOD, NNNE, NNND, ISTPS, NYREC, NYBASE, QT, QCNT, NQ, NIC, ISUF1, ISUP7, NS
3PM, NSC, NSK, IHDEF, IPDEF, ECHG, NCL2P, NCL3P, CHGCL4, ISC1CP, CHGCL5, ACST,
4PMCHG, RMGN, OCST2, C2MGN, RMR, PCR, PMR, MCR, OPCL1, LBCHK, IPNARC, SCL1CP, T
5SMFC, XBP, XBP3, MPDC, OP1, OP2, OP6, OP7, OP8
COMMON /BLK11/ NCL1P(59), CL1RR(59), IWORK2, START4, OCST, ILI(45), CHGP
1RC(5), TRANS(59), OACL1T(59), CINS(59), PMCST(59), PRST(59), ICST(5,1
20), HCHG2(16), MXN, IPDEF, OOC2D(59), OOC1D(59), NRPR(59), XCL2P, DTCST, IW
3ORK3, SFPCST, DFPCST, TRMPC, TMCST, RMPC(75), EFRR(59), ADTCST, ASFPCS, ADF
4PCS, ATMCST, ATRMPC, ECHBT(5), SMC2D, SUCL1
COMMON /BLK12/ ISTATE, MXP, NSA1S, NSMFS, NT1, NT2, NT3, NT4, NPN1, NDP1, NT
1N1, NT5, NT6, NT7, NT8, NT9, NT10, NT11

INTEGER QR, QT, QCNT, GCNT1, CR, PCR, PMR, RMR, RDA, OPEXF, OPCL1, OP1, OP2, OP
13, OP5, OP6, OP7, OP8, OP4, BIGI

REAL NFP, IACL1, MPDM, MMFC, MPDPM, IPPM, NCL2Q, NCL2Q, MCL3Q, NRPR, NCL1P,
1MFGS, MFGM, IACL1S, MTPSQ, MCL1Q, ICHG, MOCC, MOPC, MPPC, MRTLP, NCL2P, MLEN
2D, MCL1P, MCL1U, MIACL1, MMDM, ICP, NCL3, NCL3P, NTPSQ, MMFG
EQUIVALENCE (SPACE1(1), WORK3(1))
```

... (RE) INITIALIZE QUARTERLY SUMMATION VARIABLES
START3=CTIME(T1)

TRACESUBSCRIPTS

```
DO 1 I=1, NSA1
1 IRR(I)=IR(I)
DO 2 I=NT5, NT3
2 IRR(I)=I
STRMC=0.0
STCCWT=0.0
SMPC=0.0
MCL1U=0.0
SMUCP=0.0
```

```
STRNSR=0.0
XCSDA=0.0
XCTNS=0.
CSTRNS=0.
CSOACI=0.
SOACIT=0.0
SNFP=0.0
SGRCI=0.0
MOCC=0.0
MPPC=0.0
MOPC=0.0
SSOPC=0.0
SCL1D=0.0
SCL2D=0.0
SCL3D=0.0
SPRREC=0.0
SPCL1=0.0
UBLEND=0.0
SVP=0.0
SVRODC=0.0
SVFRPS=0.0
SMSP=0.0
SMSPD=0.0
SMDS=0.0
SMDSD=0.0
CEXP=0.0
TMOP=0.0
DO 3 I=1,NSPM
3 TMMC(I)=0.0
```

...(RE)INITIALIZE ANNUAL SUMMATION VARIABLES

```
IF (GT.NE.6) GO TO 8
DO 4 I=1,NT9
  ATACS(I,2)=0.0
  APCL1(I,2)=0.0
  APRREC(I,2)=0.0
  ATCCNT(I)=0.0
  AUTCP(I)=0.0
  ABLEND(I)=0.
  ACL1P(I)=0.
  ACL1UT(I)=0.
  ACL2D(I,2)=0.
  ACL3D(I,2)=0.
  ARPR(I)=0.0
  AFRPS(I)=0.0
  IF (I.GT.NSA1) GO TO 4
  ATRANS(I)=0.0
  ADACIT(I)=0.0
  ANFP(I)=0.
  ARDDC(I)=0.
  ATRND(I)=0.
  AGRC1(I)=0.0
4 CONTINUE
DO 5 I=1,NT1
DO 5 J=1,NT10
5 AMMFG(I,J)=0.0
DO 6 I=1,NT9
DO 6 J=1,NT9
  ARMMB(I,J)=0.
```

```
APMM(I,J)=0.  
IF (I.GT.NSP) GO TO 6  
ARMMS(I,J)=0.  
6 CONTINUE  
AMC1U(2)=0.  
AS1AC1(2)=0.  
ASPRRC(2)=0.  
ASPL1(2)=0.  
ASCL2D(2)=0.  
ASCL3D(2)=0.  
AMC1P(2)=0.  
AMC2P(2)=0.  
AMC3P(2)=0.  
AMBLND(2)=0.  
AUP(2)=0.  
ASNFP(2)=0.  
ASRODC(2)=0.  
ASRPS(2)=0.  
ACEXP(2)=0.  
AMRTL(2)=0.  
AMRPS(2)=0.  
AMUCP(2)=0.  
ASMFGU(2)=0.  
ASTRNS(2)=0.  
ASDAC1(2)=0.  
XCHATR=0.  
XCHASO=0.  
CHASTR=0.  
CHASOA=0.  
ASHPC=0.  
AMCC=0.  
AMOP=0.  
AMPPC=0.  
AMOPC=0.  
ACSPC=0.  
AMCRG=0.  
ASTRMC=0.  
ASTCCW=0.  
ASTRNO=0.  
ASDACC=0.  
ASDRC1=0.0  
ASMSP=0.0  
ASNSPD=0.0  
ASIDS=0.0  
ASMSDB=0.0  
ATMFGW=0.0  
ATMCST=0.0  
ATMFGU=0.0  
DO 7 I=1,NSPM  
  ATMTC(I)=0.0  
7 AMCP(I)=0.0  
8 CONTINUE  
SFR1=0.0  
VEL1=0.0  
SVP1=0.0  
SUR1=0.0  
CEX1=0.0
```

...CALCULATE QUARTERLY INFORMATION ON MOVEMENTS AND COSTS


```
DO 11 I=1,NT9
WCL1P=0.0
DO 9 J=1,NT9
9 WCL1P=WCL1P+NCL1P(J)*PMM(J,I)
  CL1D1=IFIX(CQC1D(I)+0.5)
  IF (CL1D1.NE.0.0) WCL1P=WCL1P/(CL1D1*23.2558)
  IF (CL1D1.EQ.0.0) WCL1D=WCL1D/23.2558
  FRPS(I)=NRPR(I)-WCL1P
  AFRPS(I)=AFRPS(I)+FRPS(I)/4.
  ARPR(I)=ARPR(I)+NRPR(I)/4.

...PROCESSING CENTER REPORT

  IF (GRC1(I).EQ.0.0) TRANS0(I)=0.0
  IF (GRC1(I).NE.0.0) TRANS0(I)=TRANS0(I)/GRC1(I)
  IF (GRC1(I).EQ.0.0) OACL1T(I)=0.0
  IF (GRC1(I).NE.0.0) OACL1T(I)=OACL1T(I)/GRC1(I)
  XRS=PRCST(I)+OCST
  IF (IPCEF.EQ.1) XRS=CHGPRC(NYCNT)*(PRCST(I)+OCST)
  TCCWT(I)=TRMC(I)+OACL1T(I)+XRS
  IF (GRC1(I).EQ.0.0) TCCWT(I)=0.0
  UTCP(I)=0.0
  IF (CL1CP(I).NE.0.0) UTCP(I)=((GRC1(I)*(1.+CL1RR(I)))/CL1CP(I))
  *100.
1 CL1UT(I)=0.0
  IF (PRREC(I).NE.0.0) CL1UT(I)=(PCL1(I)/PRREC(I))*100.
  AIAC1S(I,2)=AIAC1S(I,2)+CL1D1
  CL2D2=PRREC(I)-(TCL3(I)+PCL1(I))
  ACL2D(I,2)=ACL2D(I,2)+CL2D2
  ACL3D(I,2)=ACL3D(I,2)+TCL3(I)
  ATCCWT(I)=ATCCWT(I)+TCCWT(I)/4.
  AUTCP(I)=AUTCP(I)+UTCP(I)/4.
  ABLEND(I)=ABLEND(I)+BLEND(I,6)/4.
  ACL1P(I)=ACL1P(I)+NCL1P(I)/4.
  ACL1UT(I)=ACL1UT(I)+CL1UT(I)/4.
  APCL1(I,2)=APCL1(I,2)+PCL1(I)
  APRREC(I,2)=APRREC(I,2)+PRREC(I)

...PACKAGE MILK MOVEMENTS

  SMPC=SMPC+IACL1S(I)
  MOCC=MOCC+GRC1(I)-IACL1S(I)

... MILK MOVEMENTS FROM SUPPLY PLANTS

  IF (I.GT.NW.AND.I.LT.NT6) SSOPC=SSOPC+PRRS(I-NW)

... MILK MOVEMENTS FROM SUPPLY AREAS

  MPPC=MPPC+PRRD(I)
  MOPC=MOPC+SPOPD(I)
  IF (I.GT.NSA1) GO TO 10
  CEXP=CEXP+NRPR(I)*CL1D1*23.2558
  SUFRPS=SUFRPS+FRPS(I)*CL1D1*23.2558
  GO TO 11
10 XGH=1.0
  IF (IHCEF.EQ.1) XGH=CHGHC(NYCNT)
  XPR=1.0
  IF (IPDEF.EQ.1) XPR=CHGPD(QCNT)
  XRODC=BLEND(I,6)-XGH*(ERCST(I+NSP)+ACST)-PMCHG-XPR*RMPC(I+NSP)
```

```

IF (BLEND(I,6).EQ.0.0) XRCDC=0.0
SUR1=SUR1+XRCDC
SPR1=SPR1+PRREC(I)
UBL1=UBL1+PRREC(I)*BLEND(I,6)
SUP1=SUP1+UP(I)
CEX1=CEX1+NRPR(I)*CL1D1*23.2558
11 CONTINUE
DO 14 I=1,NSA1
  CHGIAC(I)=0.0
  CL1D1=IFIX(CGC1D(I)+0.5)
  IF (IACL1(I,QT-5).NE.0.0) CHGIAC(I)=((CL1D1-IACL1(I,QT-5))/IACL
1 (I,QT-5))*100.

```

...CONSUMPTION REPORT

```

CL31=TPSQ(I,QT-5)-(CL1Q(I,QT-5)+NCL2Q(I,QT-5))
CL2D2=PRREC(I)-(TCL3(I)+PCL1(I))
SCL2D=SCL2D+CL2D2
SCL3Q=SCL3Q+TCL3(I)
SCL1D=SCL1D+CL1D1
STRMC=STRMC+TRMC(I)*GRC1(I)
STCCT=STCCT+TCCT(I)*GRC1(I)

```

...MILK PRODUCTION REPORT

```

XGH=ERCST(I)+ACST
IF (I.LE.NW) GO TO 12
XGH=SPRC(I-NW)*ERCST(I+NSP)+(1.-SPRC(I-NW))*(ERCST(I)+ACST)
12 IF (IMCEF.EQ.1) XGH=CHGMC(NYCNT)*XGH
NFP(I)=BLEND(I,6)-XGH-PMCHG
IF (BLEND(I,6).EQ.0.0) NFP(I)=0.0
XPR=RMPC(I)
IF (I.LE.NW) GO TO 13
XPR=SPRC(I-NW)*RMPC(I+NSP)+(1.-SPRC(I-NW))*RMPC(I)
13 IF (IPDEF.EQ.1) XPR=CHGPD(QCNT)*XPR
RODC(I)=NFP(I)-XPR
IF (BLEND(I,6).EQ.0.0) RODC(I)=0.0

```

...FEDERAL ORDER REPORT

```

CHGCL0(I)=0.0
IF (CL1Q(I,QT-5).GT.0.0) CHGCL0(I)=((PCL1(I)-CL1Q(I,QT-5))/CL1Q
1 (I,QT-5))*100.
CHGCL2(I)=0.0
IF (NCL2Q(I,QT-5).GT.0.0) CHGCL2(I)=((CL2D2-NCL2Q(I,QT-5))/NCL2
1 Q(I,QT-5))*100.
CHGCL3(I)=0.0
IF (CL31.GT.0.0) CHGCL3(I)=((TCL3(I)-CL31)/CL31)*100.
CHGPRR(I)=0.0
IF (TPSQ(I,QT-5).NE.0.0) CHGPRR(I)=((PRREC(I)-TPSQ(I,QT-5))/TPS
1 Q(I,QT-5))*100.

```

...ADDITIONAL COMPUTATIONS REQUIRED FOR ANNUAL REPORT

```

ATTRMC(I)=ATTRMC(I)+TRMC(I)/4.
ATRANS(I)=ATRANS(I)+TRANSO(I)/4.
ADAC1T(I)=ADAC1T(I)+OACL1T(I)/4.
ANFP(I)=ANFP(I)+NFP(I)/4.
ARODC(I)=ARODC(I)+RODC(I)/4.
AGRC1(I)=AGRC1(I)+GRC1(I)

```

...SUMMARY REPORT

```
SOAC1T=SOAC1T+OACL1T(I)*GRC1(I)
SPRREC=SPRREC+PRREC(I)
SPCL1=SPCL1+PCL1(I)
VBLEND=VBLEND+PRREC(I)*BLEND(I,6)
SUP=SUP+UP(I)
SURODC=SURODC+RODC(I)*PRREC(I)
SGRC1=SGRC1+GRC1(I)
SNFP=SNFP+PRREC(I)*NFP(I)
STRNSR=STRNSR+TRANS0(I)*GRC1(I)
IF (QT.NE.9) GO TO 14
ACHGC2(I)=0.0
IF (ACL2D(I,1).GT.0.0) ACHGC2(I)=((ACL2D(I,2)-ACL2D(I,1))/ACL2D
1 (I,1))*100.
ACHGC3(I)=0.0
IF (ACL3Q(I,1).GT.0.0) ACHGC3(I)=((ACL3Q(I,2)-ACL3Q(I,1))/ACL3Q
1 (I,1))*100.
ACHGC1(I)=0.0
IF (APCL1(I,1).GT.0.0) ACHGC1(I)=((APCL1(I,2)-APCL1(I,1))/APCL1
1 (I,1))*100.0
IF (APRREC(I,1).NE.0.0) ACHGPR(I)=((APRREC(I,2)-APRREC(I,1))/AP
1 RREC(I,1))*100.
IF (AIAC1S(I,1).NE.0.0) ACHIAC(I)=((AIAC1S(I,2)-AIAC1S(I,1))/AI
1 AC1S(I,1))*100.
14 CONTINUE
ASGRC1=ASGRC1+SGRC1
IF (SPRREC+SPR1.EQ.0.0) GO TO 15
SUR1=(SURODC+SUR1)/(SPRREC+SPR1)
VBL1=(VBLEND+VBL1)/(SPRREC+SPR1)
GO TO 16
15 SUR1=0.0
VBL1=0.0
16 CEX1=CEX1+CEXP
SUP1=SUP1+SUP
```

...MANUFACTURING MOVEMENTS AND CENTER REPORT

```
SMFGU=SCL3Q
DO 24 I=1,NT1
  II=I
  IF (I.GT.NSA1.AND.I.LT.NT6) II=I-NSP
  DO 17 J=1,NSMF
    IF (IDMG1(J).EQ.II) TMMC(J)=TMMC(J)+MMFG(I,J)
```

...ADDITIONAL COMPUTATIONS FOR REPORT WRITER

```
17 CONTINUE
DO 18 K=1,NDM
  IF (IDMG2(K).EQ.II) GO TO 19
18 CONTINUE
GO TO 21
19 DO 20 L=1,NMMF
20 TMMC(L+NSMF)=TMMC(L+NSMF)+MMFG(I,L)
21 DO 24 J=1,NMFGT
  IF (J.LE.NSA1.OR.J.GT.NT) GO TO 22
  IF (J.GT.NSA1.AND.J.LT.NT6) GO TO 23
  GO TO 24
22 IF (J.LT.NMFGT) SMDS=SMDS+MMFG(I,J)
```

```
IF (J.EQ.NMFGT) SMDS=SMDS+MMFG(I,J)
GO TO 24
23 IF (J.LT.NMFGT) SMSP=SMSP+MMFG(I,J)
IF (J.EQ.NMFGT) SMSP=SMSP+MMFG(I,J)
24 CONTINUE
DO 25 I=1,NT9
DO 25 J=1,NSP
IF ((I-NW).EQ.J) GO TO 25
TMOP=TMOP+RMMS(J,I)
25 CONTINUE
TMFGM=SMDS+SMSP
TMFGU=(TMFGM/TSMFC)*100.
ATMFGM=ATMFGM+TMFGM
ATMFGU=ATMFGU+TMFGU/4.
DO 26 I=1,NSPM
UMCP(I)=0.0
IF (TMMC(I).LE.0.0) GO TO 25
IF (I.LE.NSMF) UMCP(I)=(TMMC(I)/SMFC(I))*100.
IF (I.GT.NSMF) UMCP(I)=(TMMC(I)/MMFC(I-NSMF))*100.
26 CONTINUE
DO 27 I=1,NSPM
AUMCP(I)=AUMCP(I)+UMCP(I)/4.
ATMMC(I)=ATMMC(I)+TMMC(I)
27 CONTINUE
IF (TSMFC.EQ.0.0) SMFGU=0.0
IF (TSMFC.NE.0.0) SMFGU=(SMFGU/TSMFC)*100.
IF (SGRC1.EQ.0.0) STRNSD=0.0
IF (SGRC1.NE.0.0) STRNSD=STRNSR/SGRC1
IF (SPREC.EQ.0.0) SURDC=0.0
IF (SPREC.NE.0.0) SURDC=SURDC/SPREC
IF (SGRC1.EQ.0.0) STRMC=0.0
IF (SGRC1.NE.0.0) STRMC=STRMC/SGRC1
IF (SGRC1.EQ.0.0) STCCW=0.0
IF (SGRC1.NE.0.0) STCCW=STCCW/SGRC1
ASTRC=ASTRC+STRMC/4.
ASTCCW=ASTCCW+STCCW/4.
IF (SCL1CP.EQ.0.0) SMUCP=0.0
IF (SCL1CP.NE.0.0) SMUCP=(SGRC1/SCL1CP)*100.
IF (SPREC.EQ.0.0) SNFP=0.0
IF (SPREC.NE.0.0) SNFP=SNFP/SPREC
IF (SCL1D.EQ.0.0) WFRPS=0.0
IF (SCL1D.NE.0.0) WFRPS=WFRPS/(SCL1D*23.2558)
IF (SGRC1.EQ.0.0) SOAC1D=0.0
IF (SGRC1.NE.0.0) SOAC1D=SOAC1/SGRC1
IF (SPREC.EQ.0.0) MCL1U=0.0
IF (SPREC.NE.0.0) MCL1U=(SPCL1/SPREC)*100.
IF (SPCL1.EQ.0.0) MCL1P=0.0
IF (SPCL1.NE.0.0) MCL1P=SUCL1/SPCL1
IF (SPREC.EQ.0.0) MBLND=0.0
IF (SPREC.NE.0.0) MBLND=VBLND/SPREC
IF (SCL1D.EQ.0.0) MRTLP=0.0
IF (SCL1D.NE.0.0) MRTLP=CEXP/(SCL1D*23.2558)
CMCL2=0.0
IF (MCL2Q(QT-4).NE.0.0) CMCL2=((SCL2D-MCL2Q(QT-4))/MCL2Q(QT-4))*10
10.
CMCL3=0.0
IF (MCL3Q(QT-4).NE.0.0) CMCL3=((SCL3D-MCL3Q(QT-4))/MCL3Q(QT-4))*10
10.
CIAC1=0.0
IF (MIACL1(QT-5).NE.0.0) CIAC1=((SCL1D-MIACL1(QT-5))/MIACL1(QT-5))
```

```
1*100.  
CSPREC=0.0  
IF (MTPSQ(QT-4).NE.0.0) CSPREC=((SPRREC-MTPSQ(QT-4))/MTPSQ(QT-4))*  
100.  
CSPCL1=0.0  
IF (MCL1Q(QT-4).NE.0.0) CSPCL1=((SPCL1-MCL1Q(QT-4))/MCL1Q(QT-4))*1  
100.  
CMCL1U=0.0  
IF (BMCL1U(QT-4).NE.0.0) CMCL1U=((MCL1U-BMCL1U(QT-4))/BMCL1U(QT-4)  
1)*100.  
CMCL1P=0.0  
IF (BMCL1P(QT-4).NE.0.0) CMCL1P=((MCL1P-BMCL1P(QT-4))/BMCL1P(QT-4)  
1)*100.  
CCL2P=0.0  
IF (BNCL2P(QT-4).NE.0.0) CCL2P=((NCL2P-BNCL2P(QT-4))/BNCL2P(QT-4))  
1*100.  
CCL3P=0.0  
IF (BNCL3P(QT-4).NE.0.0) CCL3P=((NCL3P-BNCL3P(QT-4))/BNCL3P(QT-4))  
1*100.  
CMLEN=0.0  
IF (BMLEN(QT-4).NE.0.0) CMLEN=((MBLEN-BMLEN(QT-4))/BMLEN(QT-4)  
1)*100.  
CSUP=0.0  
IF (BSUP(QT-4).NE.0.0) CSUP=((SUP-BSUP(QT-4))/BSUP(QT-4))*100.  
CSNFP=0.0  
IF (BSNFP(QT-4).NE.0.0) CSNFP=((SNFP-BSNFP(QT-4))/BSNFP(QT-4))*100  
1.  
CSURDD=0.0  
IF (BSURDD(QT-4).NE.0.0) CSURDD=((SURDD-BSURDD(QT-4))/BSURDD(QT-4)  
1)*100.  
CSVFRP=0.0  
IF (BSVFRP(QT-4).NE.0.0) CSVFRP=((SVFRP-BSVFRP(QT-4))/BSVFRP(QT-4)  
1)*100.  
CCEXP=0.0  
IF (BCEXP(QT-4).NE.0.0) CCEXP=((CEXP-BCEXP(QT-4))/BCEXP(QT-4))*100  
1.  
CMRTLTP=0.0  
IF (BMRTLTP(QT-4).NE.0.0) CMRTLTP=((MRTLTP-BMRTLTP(QT-4))/BMRTLTP(QT-4)  
1)*100.  
CWFRPS=0.0  
IF (BWFRPS(QT-4).NE.0.0) CWFRPS=((WFRPS-BWFRPS(QT-4))/BWFRPS(QT-4)  
1)*100.  
CMUCP=0.0  
IF (BSMUCP(QT-4).NE.0.0) CMUCP=((SMUCP-BSMUCP(QT-4))/BSMUCP(QT-4))  
1*100.  
CSMFGU=0.0  
IF (BSMFGU(QT-4).NE.0.0) CSMFGU=((SMFGU-BSMFGU(QT-4))/BSMFGU(QT-4)  
1)*100.  
IF (QCNT.LE.4) GO TO 28  
CSTRNS=0.0  
IF (BSTRNS(QT-4).NE.0.0) CSTRNS=((STRNS-BSTRNS(QT-4))/BSTRNS(QT-4)  
1)*100.  
CSOAC1=0.0  
IF (BSOAC1(QT-4).NE.0.0) CSOAC1=((SOAC1T-BSOAC1(QT-4))/BSOAC1(QT-4)  
1)*100.  
28 CONTINUE  
XCSPRC=SPRREC-MTPSQ(QT-4)  
XCSPC1=SPCL1-MIACL1(QT-5)  
XCMC2=SCL2Q-MCL2Q(QT-4)  
XCMC3=SCL3Q-MCL3Q(QT-4)
```

```
      IF (J.EQ.NMFGT) SMDSO=SMDSO+MMFG(I,J)
      GO TO 24
23     IF (J.LT.NMFGT) SMSP=SMSP+MMFG(I,J)
      IF (J.EQ.NMFGT) SMSP=SMSP+MMFG(I,J)
24 CONTINUE
      DO 25 I=1,NTS
      DO 25 J=1,NSP
      IF ((I-NW).EQ.J) GO TO 25
      TMOP=TMOP+RMMS(J,I)
25 CONTINUE
      TMFGM=SMDS+SMSP
      TMFGU=(TMFGM/TSMFC)*100.
      ATMFGM=ATMFGM+TMFGM
      ATMFGU=ATMFGU+TMFGU/4.
      DO 26 I=1,NSPM
      UMCP(I)=0.0
      IF (TMMC(I).LE.0.0) GO TO 25
      IF (I.LE.NSMF) UMCP(I)=(TMMC(I)/SMFC(I))*100.
      IF (I.GT.NSMF) UMCP(I)=(TMMC(I)/MMFC(I-NSMF))*100.
26 CONTINUE
      DO 27 I=1,NSPM
      AUMCP(I)=AUMCP(I)+UMCP(I)/4.
      ATMMC(I)=ATMMC(I)+TMMC(I)
27 CONTINUE
      IF (TSMFC.EQ.0.0) SMFGU=0.0
      IF (TSMFC.NE.0.0) SMFGU=(SMFGU/TSMFC)*100.
      IF (SGRC1.EQ.0.0) STRNSD=0.0
      IF (SGRC1.NE.0.0) STRNSD=STRNSR/SGRC1
      IF (SPREC.EQ.0.0) SURODC=0.0
      IF (SPREC.NE.0.0) SURODC=SURODC/SPREC
      IF (SGRC1.EQ.0.0) STRMC=0.0
      IF (SGRC1.NE.0.0) STRMC=STRMC/SGRC1
      IF (SGRC1.EQ.0.0) STCCWT=0.0
      IF (SGRC1.NE.0.0) STCCWT=STCCWT/SGRC1
      ASTRMC=ASTRMC+STRMC/4.
      ASTCCW=ASTCCW+STCCWT/4.
      IF (SCL1CP.EQ.0.0) SMUCP=0.0
      IF (SCL1CP.NE.0.0) SMUCP=(SGRC1/SCL1CP)*100.
      IF (SPREC.EQ.0.0) SNFP=0.0
      IF (SPREC.NE.0.0) SNFP=SNFP/SPREC
      IF (SCL1D.EQ.0.0) WFRPS=0.0
      IF (SCL1D.NE.0.0) WFRPS=SUFRPS/(SCL1D*23.2558)
      IF (SGRC1.EQ.0.0) SOAC1D=0.0
      IF (SGRC1.NE.0.0) SOAC1D=SOAC1T/SGRC1
      IF (SPREC.EQ.0.0) MCL1U=0.0
      IF (SPREC.NE.0.0) MCL1U=(SPCL1/SPREC)*100.
      IF (SPCL1.EQ.0.0) MCL1P=0.0
      IF (SPCL1.NE.0.0) MCL1P=SUCL1/SPCL1
      IF (SPREC.EQ.0.0) MBLND=0.0
      IF (SPREC.NE.0.0) MBLND=UCLND/SPREC
      IF (SCL1D.EQ.0.0) MRTLP=0.0
      IF (SCL1D.NE.0.0) MRTLP=CEXP/(SCL1D*23.2558)
      CMCL2=0.0
      IF (MCL2Q(QT-4).NE.0.0) CMCL2=((SCL2D-MCL2Q(QT-4))/MCL2Q(QT-4))*10
10.
      CMCL3=0.0
      IF (MCL3Q(QT-4).NE.0.0) CMCL3=((SCL3D-MCL3Q(QT-4))/MCL3Q(QT-4))*10
10.
      CIAC1=0.0
      IF (MIACL1(QT-5).NE.0.0) CIAC1=((SCL1D-MIACL1(QT-5))/MIACL1(QT-5))
```

```
1*100.  
CSPREC=0.0  
IF (MTPSQ(QT-4).NE.0.0) CSPREC=((SPREC-MTPSQ(QT-4))/MTPSQ(QT-4))*  
1100.  
CSPCL1=0.0  
IF (MCL1Q(QT-4).NE.0.0) CSPCL1=((SPCL1-MCL1Q(QT-4))/MCL1Q(QT-4))*1  
100.  
CMCL1U=0.0  
IF (BMCL1U(QT-4).NE.0.0) CMCL1U=((MCL1U-BMCL1U(QT-4))/BMCL1U(QT-4)  
1)*100.  
CMCL1P=0.0  
IF (BMCL1P(QT-4).NE.0.0) CMCL1P=((MCL1P-BMCL1P(QT-4))/BMCL1P(QT-4)  
1)*100.  
CCL2P=0.0  
IF (BNCL2P(QT-4).NE.0.0) CCL2P=((NCL2P-BNCL2P(QT-4))/BNCL2P(QT-4))  
1*100.  
CCL3P=0.0  
IF (BNCL3P(QT-4).NE.0.0) CCL3P=((NCL3P-BNCL3P(QT-4))/BNCL3P(QT-4))  
1*100.  
CMLEN=0.0  
IF (BMBLEN(QT-4).NE.0.0) CMBLEN=((MBLEND-BMBLEN(QT-4))/BMBLEN(QT-4)  
1)*100.  
CSUP=0.0  
IF (BSUP(QT-4).NE.0.0) CSUP=((SUP-BSUP(QT-4))/BSUP(QT-4))*100.  
CSNFP=0.0  
IF (BSNFP(QT-4).NE.0.0) CSNFP=((SNFP-BSNFP(QT-4))/BSNFP(QT-4))*100  
1.  
CSURDD=0.0  
IF (BSURDD(QT-4).NE.0.0) CSURDD=((SURDDC-BSURDD(QT-4))/BSURDD(QT-4)  
1)*100.  
CSUFRP=0.0  
IF (BSUFRP(QT-4).NE.0.0) CSUFRP=((SUFRPS-BSUFRP(QT-4))/BSUFRP(QT-4)  
1)*100.  
CCEXP=0.0  
IF (BCEXP(QT-4).NE.0.0) CCEXP=((CEXP-BCEXP(QT-4))/BCEXP(QT-4))*100  
1.  
CMRTLPL=0.0  
IF (BMRTLPL(QT-4).NE.0.0) CMRTLPL=((MRTLPL-BMRTLPL(QT-4))/BMRTLPL(QT-4)  
1)*100.  
CWFRPS=0.0  
IF (BWFRPS(QT-4).NE.0.0) CWFRPS=((WFRPS-BWFRPS(QT-4))/BWFRPS(QT-4)  
1)*100.  
CMUCP=0.0  
IF (BSMUCP(QT-4).NE.0.0) CMUCP=((SMUCP-BSMUCP(QT-4))/BSMUCP(QT-4))  
1*100.  
CSMFGU=0.0  
IF (BSMFGU(QT-4).NE.0.0) CSMFGU=((SMFGU-BSMFGU(QT-4))/BSMFGU(QT-4)  
1)*100.  
IF (QCNT.LE.4) GO TO 28  
CSTRNS=0.0  
IF (BSTRNS(QT-4).NE.0.0) CSTRNS=((STRNSR-BSTRNS(QT-4))/BSTRNS(QT-4)  
1)*100.  
CSOAC1=0.0  
IF (BSOAC1(QT-4).NE.0.0) CSOAC1=((SOAC1T-BSOAC1(QT-4))/BSOAC1(QT-4)  
1)*100.  
28 CONTINUE  
XCSPRC=SPREC-MTPSQ(QT-4)  
XCSPC1=SPCL1-MIACL1(QT-5)  
XCMC2=SCL2D-MCL2Q(QT-4)  
XCMC3=SCL3Q-MCL3Q(QT-4)
```

XCMC1U=MCL1U-BMCL1U(QT-4)
XCMC1P=MCL1P-BMCL1P(QT-4)
XCHC2P=NCL2P-BNCL2P(QT-4)
XCHC3P=NCL3P-BNCL3P(QT-4)
XCHMB=MBLEND-BMBLEN(QT-4)
XCSUP=SUP-BSUP(QT-4)
XCNFP=SNFP-BSNFP(QT-4)
XCRDC=SURDC-BSURDC(QT-4)
XCFRP=SUFRRPS-BSUFRRPS(QT-4)
XCEXP=CEXP-BCEXP(QT-4)
XCMRTP=MRTLTP-BMRTLTP(QT-4)
XCMFRP=WFRPS-DWFRPS(QT-4)
XCMUCP=SMUCP-BSMUCP(QT-4)
XCMFGU=SMFGU-BSMFGU(QT-4)
IF (QCNT.LE.4) GO TO 29
XCTRNS=STRNSR-DSTRNS(QT-4)
XCSOA=SOAC1T-BSOAC1(QT-4)

29 CONTINUE

ATMCST=ATMCST+TMCST
ASTIAC1(2)=ASTIAC1(2)+SCL1D
ASPRRC(2)=ASPRRC(2)+SPRREC
ASPCL1(2)=ASPCL1(2)+SPCL1
ASCL2D(2)=ASCL2D(2)+SCL2D
ASCL3D(2)=ASCL3D(2)+SCL3D
ASNFP(2)=ASNFP(2)+SNFP/4.
AWFRPS(2)=AWFRPS(2)+WFRPS/4.
ASMFGU(2)=ASMFGU(2)+SMFGU/4.
ASTRNS(2)=ASTRNS(2)+STRNSR
ASTRNO=ASTRNO+STRNSD/4.
ASOAC1(2)=ASOAC1(2)+SOAC1T
ASOACO=ASOACO+SOAC1D/4.
AMUCP(2)=AMUCP(2)+SMUCP/4.
AMC1U(2)=AMC1U(2)+MCL1U/4.
AMC1P(2)=AMC1P(2)+MCL1P/4.
AMBLND(2)=AMBLND(2)+MBLEND/4.
AUP(2)=AUP(2)+SUP
ASRODC(2)=ASRODC(2)+SURDC/4.
ASFRPS(2)=ASFRPS(2)+SUFRRPS
ACEXP(2)=ACEXP(2)+CEXP
AMRTL(2)=AMRTL(2)+MRTLTP/4.
AMC2P(2)=AMC2P(2)+NCL2P/4.
AMC3P(2)=AMC3P(2)+NCL3P/4.
AMC2D=AMC2D+SMC2D
IF (QT.NE.9) GO TO 32
CHASCO=0.0
IF (AMC1U(1).NE.0.0) CHASCO=((AMC1U(2)-AMC1U(1))/AMC1U(1))*100.
CHAMC1=0.0
IF (AMC1P(1).NE.0.0) CHAMC1=((AMC1P(2)-AMC1P(1))/AMC1P(1))*100.
CHAMC2=0.0
IF (AMC2P(1).NE.0.0) CHAMC2=((AMC2P(2)-AMC2P(1))/AMC2P(1))*100.
CHAMC3=0.0
IF (AMC3P(1).NE.0.0) CHAMC3=((AMC3P(2)-AMC3P(1))/AMC3P(1))*100.
CHABLND=0.0
IF (AMBLND(1).NE.0.0) CHABLND=((AMBLND(2)-AMBLND(1))/AMBLND(1))*100
1.
CHAUP=0.0
IF (AUP(1).NE.0.0) CHAUP=((AUP(2)-AUP(1))/AUP(1))*100.
CHASNFP=0.0
IF (ASNFP(1).NE.0.0) CHASNFP=((ASNFP(2)-ASNFP(1))/ASNFP(1))*100.
CHASRD=0.0


```
IF (ASRODC(1).NE.0.0) CHASRD=((ASRODC(2)-ASRODC(1))/ASRODC(1))*100
1.
CHASFG=0.0
IF (ASMFGU(1).NE.0.0) CHASFG=((ASMFGU(2)-ASMFGU(1))/ASMFGU(1))*100
1.
CHAFRP=0.0
IF (ASFRPS(1).NE.0.0) CHAFRP=((ASFRPS(2)-ASFRPS(1))/ASFRPS(1))*100
1.
CHACEX=0.0
IF (ACEXP(1).NE.0.0) CHACEX=((ACEXP(2)-ACEXP(1))/ACEXP(1))*100.
CHAMRT=0.0
IF (AMRTL(1).NE.0.0) CHAMRT=((AMRTL(2)-AMRTL(1))/AMRTL(1))*100.
CHAWRP=0.0
IF (AWFRPS(1).NE.0.0) CHAWRP=((AWFRPS(2)-AWFRPS(1))/AWFRPS(1))*100
1.
CHAUCP=0.0
IF (AMUCP(1).NE.0.0) CHAUCP=((AMUCP(2)-AMUCP(1))/AMUCP(1))*100.
IF (QCNT.LE.4) GO TO 30
CHASTR=0.0
IF (ASTRNS(1).NE.0.0) CHASTR=((ASTRNS(2)-ASTRNS(1))/ASTRNS(1))*100
1.
CHASDA=0.0
IF (ASOAC1(1).NE.0.0) CHASDA=((ASOAC1(2)-ASOAC1(1))/ASOAC1(1))*100
1.
30 CONTINUE
XCHAPR=ASPRRC(2)-ASPRRC(1)
XCHAPC=ASIAC1(2)-ASIAC1(1)
XCHAC2=ASCL2D(2)-ASCL2D(1)
XCHAC3=ASCL3D(2)-ASCL3D(1)
XCHAU=AMC1U(2)-AMC1U(1)
XCHAC1=AMC1P(2)-AMC1P(1)
XCHA2=AMC2P(2)-AMC2P(1)
XCHA3=AMC3P(2)-AMC3P(1)
XCHABL=AMBLND(2)-AMBLND(1)
XCHAUP=AUP(2)-AUP(1)
XCHANF=ASNFP(2)-ASNFP(1)
XCHARD=ASRODC(2)-ASRODC(1)
XCHAFR=ASFRPS(2)-ASFRPS(1)
XCHACE=ACEXP(2)-ACEXP(1)
XCHAMR=AMRTL(2)-AMRTL(1)
XCHAMF=AWFRPS(2)-AWFRPS(1)
XCHACP=AMUCP(2)-AMUCP(1)
XCHAGU=ASMFGU(2)-ASMFGU(1)
IF (QCNT.LE.4) GO TO 31
XCHATR=ASTRNS(2)-ASTRNS(1)
XCHASD=ASOAC1(2)-ASOAC1(1)
31 CONTINUE
CHASPR=0.0
IF (ASPRRC(1).NE.0.0) CHASPR=((ASPRRC(2)-ASPRRC(1))/ASPRRC(1))*100
1.
CHASPC=0.0
IF (ASPCL1(1).NE.0.0) CHASPC=((ASPCL1(2)-ASPCL1(1))/ASPCL1(1))*100
1.
CHASC1=0.0
IF (ASIAC1(1).NE.0.0) CHASC1=((ASIAC1(2)-ASIAC1(1))/ASIAC1(1))*100
1.
CHASC2=0.0
IF (ASCL2D(1).NE.0.0) CHASC2=((ASCL2D(2)-ASCL2D(1))/ASCL2D(1))*100
1.
CHASC3=0.0
```

```
IF (ASCL3Q(1).NE.0.0) CHASC3=((ASCL3Q(2)-ASCL3Q(1))/ASCL3Q(1))*100
1.
32 CONTINUE
...SUM INDIVIDUAL MOVEMENTS FOR ANNUAL REPORT
DO 33 I=1,NT9
DO 33 J=1,NT9
ARMMD(I,J)=ARMMD(I,J)+RMMD(I,J)
APMM(I,J)=APMM(I,J)+PMM(I,J)
IF (I.GT.NSP) GO TO 33
ARMMS(I,J)=ARMMS(I,J)+RMMS(I,J)
33 CONTINUE
DO 36 I=1,NT1
DO 36 J=1,NT10
IF (I.LE.NSA1.OR.I.GT.NT) GO TO 34
IF (I.GT.NSA1.AND.I.LT.NT6) GO TO 35
GO TO 36
34 IF (J.LT.NMFCT) ASMDS=ASMDS+MMFG(I,J)
IF (J.EQ.NMFCT) ASMDS=ASMDS+MMFG(I,J)
GO TO 36
35 IF (J.LT.NMFCT) ASMSP=ASMSP+MMFG(I,J)
IF (J.EQ.NMFCT) ASMSP=ASMSP+MMFG(I,J)
36 AMMFG(I,J)=AMMFG(I,J)+MMFG(I,J)
ASOPC=ASOPC+SSOPC
ATMOP=ATMOP+TMOP
AMPPC=AMPPC+MPPC
ANOPC=ANOPC+MOPC
ASMPC=ASMPC+SMPC
AMOCC=AMOCC+MOCC
DO 37 I=1,NT9
37 PCL2(I)=PRREC(I)-(TCL3(I)+PCL1(I))

REWIND OP6
WRITE (OP6) WORK3
CALL TCOST

...WRITE TO GRADE B REPORT WRITER - DRW2
IF (ISTATE.NE.0) WRITE (OP8) SUP1,SUR1,UBL1,CEX1,NCL3P

REWIND OP6
READ (OP6) WORK3

TIME3=START3-CTIME(T1)
...STORE QUARTERLY DATA FOR VARIABLES TO BE USED IN COMPARATIVE
REPORTS IN WORK1 ARRAY

IF (NIC.NE.0) CALL OSET (QT,NYCNT,NIC,ICR,NT9,PRREC,PCL1,PCL2,TCL3
1,CL1UT,NCL1P,BLEND(1,6),UTCP,TCNT,CBC1D,SPACE1,NT9)

...STORE QUARTERLY DATA TO BE USED IN COMPARATIVE SUMMARY REPORTS

WRITE (OP2) SPRREC,SPCL1,SCL2D,SCL3Q,MCL1U,MCL1P,NCL2P,NCL3P,MLEN
1D,SUP,SNFP,SURDDC,SURRPS,CEXP,MRTLP,WFRPS,SMUCP,SMFGU,STRNSR,SFPCS
2T,DTCST,DFPCST,SOAC1T,TCST,TRMPC

N1=25+(NYCNT-1)*25+(QT-6)*150
REWIND OP2
READ (OP2) (SPACE1(N1+IJK),IJK=1,25)
```

```
REWIND OP2
START7=CTIME(T1)
...PRINT QUARTERLY REPORT IF REQUESTED
  IF (QR(QCNT).LT.1) GO TO 40
  IF (QR(QCNT).EQ.3) GO TO 39
...MERGED ORDER REPORT
  IF (IMR.LE.0) GO TO 38
  WRITE (OP1) (PRRECM(I),PRRECM(I+10),BLNDMR(I+10),BLNDMR(I),I=1,IMR
  1)
...SUMMARY REPORT
38 WRITE (OP1) QT,NYRBC,SPRREC,XCSPRC,CSPREC,SPCL1,XCSPC1,CIAC1,SCL2D
  1,XCMC2,CMCL2,SCL3Q,XCMC3,CMCL3,MCL1U,XCMC1U,CMCL1U,MCL1P,XCMC1P,CM
  2CL1P,NCL2P,XCHC2P,CCL2P,NCL3P,XCHC3P,CCL3P,MBLEND,XCHMB,CMBLEN,SUP
  3,XCSUP,CSUP,SNFP,XCNFP,CSNFP,SURODC,XCRDC,CSURD,SUFRPS,XCFRP,CSUF
  4RP,CEXP,XCEXP,CCEXP,MRTL P,XCMRTP,CMRTP,WFRPS,XCWRP,CWRPS,SMUCP,X
  5CUCP,CMUCP,SMFGU,XCMFU,CSMFGU,STRNSR,XCTRNS,CSTRNS,SOAC1T,XCSOA,CS
  6OAC1
  IF (QR(QCNT).EQ.4) GO TO 40
...FEDERAL ORDER REPORT
  WRITE (OP1) (PRREC(IR(I)),CHGPRR(IR(I)),CL1UT(IR(I)),NCL1P(IR(I)),
  1NCL2P,NCL3P,BLEND(IR(I),6),I=1,NSA1),SPRREC,CSPREC,MCL1U,MCL1P,NCL
  2P,NCL3P,MBLEND
  WRITE (OP1) (PCL1(IR(I)),CHGCL0(IR(I)),PCL2(IR(I)),CHGCL2(IR(I)),T
  1CL3(IR(I)),CHGCL3(IR(I)),I=1,NSA1),SPCL1,SCL2D,SCL3Q,CSPCL1,CMCL2,
  2CMCL3,SMC20
...MILK PRODUCTION REPORT
  WRITE (OP1) (PRREC(IR(I)),CHGPRR(IR(I)),BLEND(IR(I),6),NFP(IR(I)),
  1RODC(IR(I)),I=1,NSA1),SPRREC,CSPREC,MBLEND,SNFP,SURODC
...PROCESSING CENTER REPORT
  WRITE (OP1) (GRC1(IR(I)),UTCP(IR(I)),TRMC(IR(I)),TRANSO(IR(I)),OAC
  1L1T(IR(I)),TCCNT(IR(I)),I=1,NSA1),SMUCP,STRMC,STRNSD,SOAC1D,STCCWT
  2,SGRC1
...MANUFACTURING CENTER REPORT
  WRITE (OP1) (TMMC(I),UMCP(I),I=1,NSPM),TMFGM,TMFGU
...CONSUMPTION REPORT
  WRITE (OP1) (CCO1D(IR(I)),CHGIAC(IR(I)),NRPR(IR(I)),FRPS(IR(I)),I=
  11,NSA1),SCL1D,CIAC1,MRTL P,WFRPS
  IF (QR(QCNT).GT.1) GO TO 40
... MILK MOVEMENTS TO PROCESSING CENTERS FROM DIRECT SHIPPERS
```

```
39 WRITE (OP1) ((RMMD(IRR(I),IRR(J)),J=1,NT9),I=1,NT9),MPPC,MOPC
```

```
...MILK MOVEMENTS TO PROCESSING CENTERS FROM SUPPLY PLANTS
```

```
WRITE (OP1) ((RMMS(I,IRR(J)),J=1,NT9),I=1,NSP),SSOPC,TMOP
```

```
...MILK MOVEMENTS TO MANUFACTURING CENTERS FROM DIRECT SHIPPERS
```

```
WRITE (OP1) ((MMFG(I,J),J=1,NT10),I=1,NSA1),((MMFG(I,J),J=1,NT10),  
II=NT6,NT1),SMDS,SMDS
```

```
...MILK MOVEMENTS TO MANUFACTURING CENTERS FROM SUPPLY PLANTS
```

```
WRITE (OP1) ((MMFG(I+NSA1,J),J=1,NT10),I=1,NSP),SMSP,SMSP
```

```
...PACKAGED MILK MOVEMENTS
```

```
WRITE (OP1) ((PMM(IRR(I),IRR(J)),J=1,NT9),I=1,NT9),SMPC,MOCC  
40 CONTINUE
```

```
...NEW DATA FOR NEXT SOLUTION
```

```
TIME7=START7-CTIME(T1)
```

```
IF (QCNT.EQ.NQ) GO TO 42
```

```
DO 41 I=1,NT9
```

```
IF (ISUP1.NE.1.AND.ISUP7.LT.1) NCLIP(I)=CLIP(I,1)+CHGCL1(I)*FLO
```

```
1 AT(QCNT)
```

```
IF (ISUP7.GE.1) NCLIP(I)=QCLIP(I,QCNT+1)
```

```
IF (ISUP1.EQ.1.AND.ISUP7.LT.1) NCLIP(I)=(1.+FLOAT(QCNT)*CHGCL1(  
1 I))*CLIP(I,1)
```

```
IF (NCLIP(I).LT.0.0) NCLIP(I)=0.0
```

```
IF (I.LE.NW.OR.I.GT.NSA1) GO TO 41
```

```
IF (ISUP1.NE.1.AND.ISUP7.LT.1) CLISP(I-NW)=CLIP(I-NW,2)+CHGCL1(  
1 I)*FLOAT(QCNT)
```

```
IF (ISUP7.GE.1) CLISP(I-NW)=QCLIP(I,QCNT+1)
```

```
IF (ISUP1.EQ.1.AND.ISUP7.LT.1) CLISP(I-NW)=(1.+FLOAT(QCNT)*CHGCL1(  
1 I))*CLIP(I-NW,2)
```

```
IF (CLISP(I-NW).LT.0.0) CLISP(I-NW)=0.0
```

```
41 CONTINUE
```

```
42 CONTINUE
```

```
...REINITIALIZE DATA BASE WITH CURRENT QUARTER DATA
```

```
SUMXX=0.0
```

```
SUMCP=0.0
```

```
DO 45 I=1,NT9
```

```
QSPL1(I)=CGTSP(I)
```

```
IABCD=IFIX(CGTSPL(I)+0.5)
```

```
XX=PRECD(I)-FLOAT(IABCD)
```

```
IF (I.LE.NW.OR.I.GT.NSA1) GO TO 43
```

```
X1=IFIX((1.0-SPRC(I-NW))*CGTSP(I)+0.5)
```

```
X2=IFIX(SPRC(I-NW)*CGTSP(I)+0.5)
```

```
SPRC(I-NW)=X2/PRREC(I)
```

```
43 XX=PRREC(I)-(X1+X2)
CONTINUE

SUMXX=SUMXX+XX

IF (BTPSQ(I,QT-5).GT.0.0) CPRCHG(I)=CPRCHG(I)+XX/BTPSQ(I,QT-5)
IF (BTPSQ(I,QT-5).EQ.0.0.AND.PRREC(I).GT.0.0) BTPSQ(I,QT-4)=FRR
1 EC(I)

SUMCP=SUMCP+CPRCHG(I)

IACL1(I,QT-5)=CQC1D(I)
TPSQ(I,QT-5)=PRREC(I)
CL1Q(I,QT-5)=PCL1(I)
NCL2Q(I,QT-5)=PRREC(I)-(TCL3(I)+PCL1(I))
BLEND(I,5)=0.0
XPR1=0.0
XPR=RMPC(I)
IF (I.GT.NSA1) XPR=RMPC(I+NSP)
IF (I.LE.NW.OR.I.GT.NSA1) GO TO 44
XPR=SPRC(I-NW)*RMPC(I+NSP)+(1.-SPRC(I-NW))*RMPC(I)
44 IF (IPDEF.EQ.1) XPR1=CHGPD(QCNT)*XPR-XPR
IF (BLEND(I,6).GT.XPR1) BLEND(I,5)=BLEND(I,6)-XPR1
IF (PRREC(I).EQ.0.0.OR.BLEND(I,5).EQ.0.0) CPRCHG(I)=0.0
45 CONTINUE

MTPSQ(QT-4)=SPREC
MCL1Q(QT-4)=SPCL1
MIACL1(QT-5)=SCL1D
MCL2Q(QT-4)=SCL2D
MCL3Q(QT-4)=SCL3Q
BMCL1U(QT-4)=MCL1U
BMCL1P(QT-4)=MCL1P
BNCL2P(QT-4)=NCL2P
BNCL3P(QT-4)=NCL3P
BMBLEN(QT-4)=MBLEND
BSUP(QT-4)=SUP
BSNFP(QT-4)=SNFP
BSURDD(QT-4)=SURDDC
BSUFRP(QT-4)=SUFRPS
BCEXP(QT-4)=CEXP
BMRTLQ(QT-4)=MRTLQ
BWFRPS(QT-4)=WFRPS
BSMUCP(QT-4)=SMUCP
BSMFCU(QT-4)=SMFCU
BSTRNS(QT-4)=STRNSR
BSOAC1(QT-4)=SOAC1T

...CALCULATE NEW CLASS 2 , CLASS 3 PRICES

XYN=QCST2
IF (IPDEF.EQ.1) XYN=CHGPRC(NYCNT)*QCST2
XCL2P=C2MGN*(NCL2P+XYN)
IF (ISUP1.EQ.1) NCL2P=(1.+FLOAT(QCNT)*CHGCL4)*XBP
IF (ISUP1.EQ.1) NCL3P=(1.+FLOAT(QCNT)*CHGCL5)*XBP3
IF (ISUP1.NE.1.AND.ISUP7.NE.1.AND.ISUP7.NE.3) NCL2P=XBP+CHGCL4*FLO
1AT(QCNT)
IF (ISUP1.NE.1.AND.ISUP7.NE.1.AND.ISUP7.NE.3) NCL3P=XBP3+CHGCL5*FL
10AT(QCNT)
IF (ISUP7.EQ.1.OR.ISUP7.EQ.3) NCL2P=QCL2P(QCNT+1)
```

```
IF (ISUP7.EQ.1.OR.ISUP7.EQ.3) NCL3P=QCL3P(QCNT+1)
IF (NCL2P.LT.0.0) NCL2P=0.0
IF (NCL3P.LT.0.0) NCL3P=0.0
```

...INCREMENT QUARTER COUNTERS

```
QCNT=QCNT+1
QCNT1=QCNT
QT=QT+1
IF (QT.NE.10) GO TO 50
```

...STORE DATA (ANNUAL) FOR VARIABLES TO BE USED IN COMPARATIVE REPORTS IN WORK1 ARRAY

```
IF (NIC.NE.0) CALL OSET (QT, NYCNT, NIC, ICR, NTS, AFRREC(1,2), APCL1(1,2), ACL2D(1,2), ACL3Q(1,2), ACL1UT, ACL1P, ABLEND, AUTCP, ATCCWT, AIACIS(1,2), SPACE1, NTS)
```

...STORE (ANNUAL) DATA TO BE USED IN COMPARATIVE SUMMARY REPORTS

```
WRITE (OP2) ASPRR(2), ASPCL1(2), ASCL2D(2), ASCL3Q(2), AMC1U(2), AMC1P(2), AMC2P(2), AMC3P(2), AMBLND(2), AUP(2), ASNFP(2), ASRODC(2), ASFRPS(2), ACEXP(2), AMRTL(2), AMFRPS(2), AMUCP(2), ASMFGU(2), ASTRNS(2), ASFP(2), 3ADTCST, ADFPCS, ASDAC1(2), ATMCST, ATRMPC
REWIND OP2
N1=N1+150
READ (OP2) (SPACE1(N1+IJK), IJK=1,25)
REWIND OP2
```

...PRINT ANNUAL REPORT IF REQUESTED

```
IF (MR(NYRBC-NYBASE).LT.1) GO TO 48
IF (MR(NYRBC-NYBASE).EQ.3) GO TO 47
```

...MERGED ORDER REPORT

```
IF (IMR.LE.0) GO TO 46
WRITE (OP1) (PRECAM(I), VPAMRG(I), BLAMR(I+10), BLAMR(I), I=1, IMR)
```

...SUMMARY REPORT

```
46 WRITE (OP1) NYRBC, ASPRR(2), XCHAPR, CHASPR, ASPCL1(2), XCHAPC, CHASC1, 1ASCL2D(2), XCHAC2, CHASC2, ASCL3Q(2), XCHAC3, CHASC3, AMC1U(2), XCHAU, CHA2SC0, AMC1P(2), XCHAC1, CHAMC1, AMC2P(2), XCHAC2, CHAMC2, AMC3P(2), XCHAC3, CHAMC3, AMBLND(2), XCHABL, CHABL, AUP(2), XCHAUP, CHAUP, ASNFP(2), XCHANF, C4HASNF, ASRODC(2), XCHARD, CHASRD, ASFRPS(2), XCHAFR, CHAFRP, ACEXP(2), XCH5ACE, CHACEX, AMRTL(2), XCHAMR, CHAMRT, AMFRPS(2), XCHAWF, CHAWRP, AMUCP(2) 6, XCHACP, CHAUCP, ASMFGU(2), XCHAGU, CHASFG, ASTRNS(2), XCHATR, CHASTR, ASD7AC1(2), XCHASQ, CHASQA
IF (MR(NYRBC-NYBASE).EQ.4) GO TO 48
```

...FEDERAL ORDER REPORT

```
WRITE (OP1) (APRREC(IR(I),2), ACHGPR(IR(I)), ACL1UT(IR(I)), ACL1P(IR(I)), AMC2P(2), AMC3P(2), ABLEND(IR(I)), I=1, NSAI), ASPRR(2), CHASPR, AMC21U(2), AMC1P(2), AMC2P(2), AMC3P(2), AMBLND(2)
WRITE (OP1) (APCL1(IR(I),2), ACHGC1(IR(I)), ACL2D(IR(I),2), ACHGC2(IR(I)), ACL3Q(IR(I),2), ACHGC3(IR(I)), I=1, NSAI), ASPCL1(2), ASCL2D(2), AS2CL3Q(2), CHASPC, CHASC2, CHASC3, AMC2D
```

...MILK PRODUCTION REPORT

```
WRITE (OP1) (APRREC(IR(I),2),ACHGPR(IR(I)),ABLEND(IR(I)),ANFP(IR(I),1),ARDC(IR(I)),I=1,NSA1),ASPRRC(2),CHASPR,AMBLND(2),ASNFP(2),ASRO2DC(2))
```

...PROCESSING CENTER REPORT

```
WRITE (OP1) (AGRC1(IR(I)),AUTCP(IR(I)),ATRMC(IR(I)),ATRANS(IR(I)),1ADAC1T(IR(I)),ATCCWT(IR(I)),I=1,NSA1),AMUCP(2),ASTRMC,ASTRNO,ASOAC20,ASTCCW,ASGRCl
```

...MANUFACTURING CENTER REPORT

```
WRITE (OP1) (ATMMC(I),AUMCP(I),I=1,NSPM),ATMFCM,ATMFCU
```

...CONSUMPTION REPORT

```
WRITE (OP1) (AIACIS(IR(I),2),ACHIAC(IR(I)),ARPR(IR(I)),AFRPS(IR(I),1),I=1,NSA1),ASIAC1(2),CHASC1,AMRTL(2),AMFRPS(2)IF (MR(NYRBC-NYBASE).GT.1) GO TO 48
```

...MILK MOVEMENTS TO PROCESSING CENTERS FROM DIRECT SHIPPERS

```
47 WRITE (OP1) ((ARMMD(IRR(I),IRR(J)),J=1,NT9),I=1,NT9),AMPPC,AMOPC
```

...MILK MOVEMENTS TO PROCESSING CENTERS FROM SUPPLY PLANTS

```
WRITE (OP1) ((ARMMS(I,IRR(J)),J=1,NT9),I=1,NSP),ASOPC,ATMOP
```

...MILK MOVEMENTS TO MANUFACTURING CENTERS FROM DIRECT SHIPPERS

```
WRITE (OP1) ((AMMFG(I,J),J=1,NT10),I=1,NSA1),((AMMFG(I,J),J=1,NT10),I=NT6,NT1),ASMDS,ASMDSO
```

...MILK MOVEMENTS TO MANUFACTURING CENTERS FROM SUPPLY PLANTS

```
WRITE (OP1) ((AMMFG(I+NSA1,J),J=1,NT10),I=1,NSP),ASMSP,ASMSPD
```

...PACKAGE MILK MOVEMENTS

```
WRITE (OP1) ((APMM(IRR(I),IRR(J)),J=1,NT9),I=1,NT9),ASMPC,AMOCC48 CONTINUE
```

...REINITIALIZE ANNUAL VARIABLES

```
NYRBC=NYRBC+1  
NYCNT=NYCNT+1  
DO 49 I=1,NT9  
  AIACIS(I,1)=AIACIS(I,2)  
  APCL1(I,1)=APCL1(I,2)  
  APRREC(I,1)=APRREC(I,2)  
  ACL2D(I,1)=ACL2D(I,2)
```

```
ACL3G(I,1)=ACL3G(I,2)
49 CONTINUE
AMC1U(1)=AMC1U(2)
AMC1P(1)=AMC1P(2)
AMC2P(1)=AMC2P(2)
AMC3P(1)=AMC3P(2)
AMBLND(1)=AMBLND(2)
AUP(1)=AUP(2)
ASNFP(1)=ASNFP(2)
ASRODC(1)=ASRODC(2)
ASFRPS(1)=ASFRPS(2)
ACEXP(1)=ACEXP(2)
AMRTL(1)=AMRTL(2)
AWFRPS(1)=AWFRPS(2)
AMUCP(1)=AMUCP(2)
ASMFGU(1)=ASMFGU(2)
ASTRNS(1)=ASTRNS(2)
ASDAC1(1)=ASDAC1(2)
ASAC1(1)=ASAC1(2)
ASPRRC(1)=ASPRRC(2)
ASPCL1(1)=ASPCL1(2)
ASCL2D(1)=ASCL2D(2)
ASCL3G(1)=ASCL3G(2)
QT=6
50 CONTINUE
RETURN

END
SUBROUTINE TCOST
```

...CALCULATE AGGREGATE TRANSPORTATION AND PRODUCTION COSTS

```
COMMON DSPMI(59,59),RMMA(59,59)
```

```
DSPMI=SIZE OF RMMD
RMMA =SIZE OF RMMD
```

```
COMMON /BLK1/ RMMD(59,59),SPOPD(59),PRROD(59),TRANSR(59),TRMC(59),
1RMMS(16,59),PRRD(59),CL3SP(16),CL3DS(59),PRRS(16),CL1SLS(59),IACL1
2S(59),PMM(59,59),OACL1S(59),MMFG(75,42)
COMMON /BLK2/ RDA(5),NSP,NSA1,NSMF,NDM,NMMF,NMFGT,BIGI
COMMON /BLK5/ CPRCHG(59),BTPO(59,4),BLEND(59,6),SE(59),GRBIND(4),
1GRBCNV(59),IGBEF,CQTS(59),QSPL1(59)
COMMON /BLK9/ NW,N3N,NT,NPN,NNN,NMN,NMM,NMD,NDP,NMS,NTN
COMMON /BLK8/ NYCNT
COMMON /BLK10/ MMFC(27),CL1CP(59),CL1SP(16),IR(61),CHGHC(5),CHGPD(
120),ERCST(75),SPRC(16),IDMG1(26),IDMG2(19),SMFC(40),SSMFC,SMMFC,SM
2XC,NNDD,NNNE,NNND,ISTPS,NYRBC,NYBASE,QT,QCNT,NQ,NIC,ISUP1,ISUP7,NS
3PM,NSC,NSK,IHCEF,IPDEF,ECHG,NCL2P,NCL3P,CHGCL4,ISC1CP,CHGCL5,ACST,
4PMCHG,RMGN,OCST2,C2MGN,RMR,PCR,PMR,MCR,OPCL1,LBCHK,IPNARC,SCL1CP,T
5SMFC,XBP,XBP3,MFDC,OP1,OP2,OP6,OP7,OP8
COMMON /BLK11/ NCL1P(59),CL1RR(59),IWORK2,START4,OCST,ILI(45),CHGP
1RC(5),TRANSO(59),OACL1T(59),CINSP(59),PMCST(59),PRCST(59),ICST(5,1
20),HCHG2(16),MXN,IPCEF,COC2D(59),COC1D(59),NRPR(59),XCL2P,DTCST,IW
3ORR3,SFPCST,DFPCST,TRMPC,TMCST,RMFC(75),EFRR(59),ADTCST,ASFPCS,ADF
4PCS,ATMCST,ATRMPC,ECHBT(5),SMC2D,SUCL1
COMMON /BLK12/ ISTATE,MXP,NSA1S,NSMFS,NT1,NT2,NT3,NT4,NPN1,NDP1,NT
1N1,NT5,NT6,NT7,NT8,NT9,NT10,NT11
```


INTEGER RDA,OP1,OP2,OP5,OP6,OP7,OPS,QT,QCNT,RMR,PCR,PMR,OPCL1
REAL IACL1S,MMFG

```
IF (QT.NE.6) GO TO 1
ADTCST=0.0
ASFPCS=0.0
ADFFPCS=0.0
ATRMPC=0.0
1 DTCST=0.0
SFFPCS=0.0
DFPCST=0.0
TRMPC=0.0
REWIND OP7
IF (QCNT.LE.1) GO TO 2
READ (OP7) ((DSPMI(I,J),I=1,NT9),J=1,NT9)
READ (OP7) ((RMMA(I,J),I=1,NT9),J=1,NT9)
GO TO 5
2 DO 3 J=1,NT9
3 READ (OP7) (DSPMI(I,J),I=1,NT9)
DO 4 I=1,NT9
DO 4 J=1,NT9
4 RMMA(J,I)=RMMD(J,I)
5 DO 8 I=1,NT9
IF (I.GT.NW) SFFPCS=SFFPCS+CQTSP(I)*SPRC(I-NW)*(ERCST(I+NSP)*CH
1 GHC(NYCNT))
IJK=I
IF (I.GT.NSA1) IJK=I+NSP
DFPCST=DFPCST+RMMD(I,I)*(ERCST(IJK)+ACST)*CHGHC(NYCNT)
DO 6 K=1,MMFGT
TRMPC=TRMPC+MMFG(IJK,K)*RMPC(IJK)*CHGPD(QCNT)
6 DFPCST=DFPCST+MMFG(IJK,K)*(ERCST(IJK)+ACST)*CHGHC(NYCNT)
QPCS1=1.0
IF (QSPL1(I).GT.0.0) QPCS1=1.0+((CQTSP(I)-QSPL1(I))/QSPL1(I))
IF (QCNT.EQ.1) GO TO 8
RMMA(I,I)=RMMD(I,I)
DO 7 J=1,NT9
IF (J.EQ.I) GO TO 7
ADJ1=0.0
ADJ1=QPCS1*RMMA(J,I)
IF (ADJ1.GT.0.0) RMMA(I,I)=RMMA(I,I)-ADJ1
RMMA(J,I)=ADJ1+RMMD(J,I)
7 CONTINUE
8 CONTINUE
JCOL=1
DO 9 I=1,NT9
IF (I.EQ.NT9) GO TO 10
JCOL=JCOL+1
DO 9 J=JCOL,NT9
DIFF1=RMMA(I,J)-RMMA(J,I)
RMMA(J,I)=0.0
RMMA(I,J)=0.0
IF (DIFF1.GT.0.0) RMMA(I,J)=DIFF1
IF (DIFF1.LT.0.0) RMMA(J,I)=ABS(DIFF1)
9 CONTINUE
10 CONTINUE
DO 11 I=1,NT9
IF (I.GT.NW.AND.I.LE.NSA1) TRMPC=TRMPC+RMPC(I+NSP)*CQTSP(I)*SPR
1 C(I-NW)*CHGPD(QCNT)
IJK=I
IF (I.GT.NSA1) IJK=I+NSP
```

```
DO 11 J=1,NT9
  IF (I.NE.J) DTCST=DTCST+(DSPMI(I,J)+ACST)*RMMA(I,J)*ECHBT(NYCNT
1 )
11 TRMPC=TRMPC+RMPC(IJK)*RMMA(I,J)*CHGPD(GCNT)
  ADTCST=ADTCST+DTCST
  ASFPCS=ASFPCS+SFPCST
  ADFPCS=ADFPCS+DFPCST
  ATRMPC=ATRMPC+TRMPC
  REWIND OP7
  WRITE (OP7) ((DSPMI(I,J),I=1,NT9),J=1,NT9)
  WRITE (OP7) ((RMMA(I,J),I=1,NT9),J=1,NT9)
  RETURN

END
SUBROUTINE OSET (QT,NYR,NIC,ICR,NSA1,PRREC,PCL1,PCL2,TCL3,CL1UT,NC
1L1P,BLEND,UTCP,TCCWT,COC1D,WORK1,NT9)
```

...SUBROUTINE OSET

...COMPARATIVE REPORT DATA PLACED IN WORK1 ARRAY

```
DIMENSION ICR(10), NO(5), PRREC(1), PCL1(1), PCL2(1), TCL3(1), CL1
1UT(1), NCL1P(1), BLEND(1), UTCP(1), TCCWT(1), COC1D(1), WORK1(1)
```

```
REAL NCL1P
INTEGER QT
```

.....N1,NO(1),NO(2),NO(3),NO(4),NO(5)/1,751,2101,3451,
4801,6151/

```
N=750+(NYR*NT9)+((QT-6)*(6*NT9))
DO 1 I=1,NIC
1 NO(I)=N+(I-1)*(30*NT9)
  KJ=0
  DO 13 M=1,10
    IF (ICR(M).EQ.0) GO TO 13
    KJ=KJ+1
    DO 12 I=1,NSA1
      NN=NO(KJ)+I
      GO TO (2,3,4,5,6,7,8,9,10,11), M
2     WORK1(NN)=PRREC(I)
      GO TO 12
3     WORK1(NN)=PCL1(I)
      GO TO 12
4     WORK1(NN)=PCL2(I)
      GO TO 12
5     WORK1(NN)=TCL3(I)
      GO TO 12
6     WORK1(NN)=CL1UT(I)
      GO TO 12
7     WORK1(NN)=NCL1P(I)
      GO TO 12
8     WORK1(NN)=BLEND(I)
      GO TO 12
9     WORK1(NN)=UTCP(I)
      GO TO 12
10    WORK1(NN)=TCCWT(I)
      GO TO 12
11    WORK1(NN)=COC1D(I)
```

```
12 CONTINUE  
13 CONTINUE  
RETURN
```

```
END  
FUNCTION CTIME(T1)
```

```
...CTIME ROUTINE FOR 6000, 7000 SERIES CDC
```

```
CALL SECOND (T1)  
CTIME=-T1  
RETURN
```

```
END
```

DRW1

DRW1, listed on the following pages, reports results from stage one of DAMPS. Some of these results are aggregated and used in DRW2, as well. DRW1 will list the input form and print the following reports, both quarterly and annually, as requested:

1. Summary Report
2. Comparative Summary Report
3. Federal Order Report
4. Federal Order Report - Sales
5. Milk Production Report
6. Processing Center Report
7. Manufacturing Center Report
8. Consumption Report
9. Raw Milk Movements from Direct Ship Supply Areas to Processing Centers
10. Raw Milk Movements from Supply Plants to Processing Centers
11. Raw Milk Movements from Direct Ship Supply Areas to Manufacturing Centers
12. Raw Milk Movements from Supply Plants to Processing Centers
13. Packaged Milk Movements from Processing Centers to Consumption Centers
14. Comparative Reports
 - a. Producers Receipts
 - b. Class I Sales
 - c. Class II Sales
 - d. Class III Sales
 - e. Class I Utilization
 - f. Class I Price
 - g. Blend Price
 - h. Processing Capacity Utilization
 - i. Total Processing Cost
 - j. In-Area Class I Sales
15. Merged Order Report
16. Unregulated Grade A Report

Reports 1 through 8 and 14h through 14j are exclusively for Federal Orders. Movements reports 9 through 13 are for state regulated areas as well as Federal Orders. Comparative Reports 14a through 14g can be obtained for state regulated areas and Federal Orders. Details on the contents of these reports and how they are obtained are available in Novakovic, et al. (8).

PROGRAM DRW1 (INPUT, OUTPUT, TAPE4, TAPE8, TAPE6=OUTPUT, TAPE2=INPUT, TAPE16, TAPE17)

.... DRW1 IS THE FOURTH PROGRAM OF DAMPS
DRW1 IS THE GRADE A REPORT WRITER FOR DAMPS. DRW1 WAS
WRITTEN BY D.R. MARTELLA, J.E. PRATT, AND A.M. NOVAKOVIC.

INTEGER QR, CR, QT, YEND, QCNT

DIMENSION A(75), B(59), C(59), D(59), E(59), F(59), G(59), H(59),
1AA(59,59), NOR(75), ONAME(75), II(59), QR(20), MR(5), MFG(33), IRC(261),
ICR(10), ICRS(10), MFID(70), MFNAME(70), BB(53), CC(53), NY2(36),
IMRN(11), IMRG(59), JPLT(10), III(59), CMP(2,2), QRB(5), ICRB(414),
MFN(14), DGSC(20), UNREG(9), UNC1S(9), UNGBP(9), UNRODC(9), U5NNBP(9),
IXT(3), NAME(9), AUNREG(6,9), AUNC1S(6,9), AUNGBP(6,9), A6UNNBP(6,9),
AUNROD(6,9)

DATA JPLT/1,2,4,5,9,12,15,5,17,0/

DATA KN1,KN2,KN3,KN4/1,7,8,9/

DATA CMP/10HFEDERAL OR, 4HDEFS, 10H STATE ORD, 4HERS /

DATA MFN/3HMAL, 3HMCA, 3HMME, 3HMMA, 3HMMT, 3HMNU, 3HMNY, 3HMNC, 3HMND, 3HM1PA, 3HMSC, 3HMUT, 3HMUA, 3HMUY/

DATA NAME/3HNORTHEAST, 9MCORN BELT, 9HLAKE, 9HSOUTHEAST, 9HS.CENTRAL, 9HPRAIRIE, 9HMOUNTAIN, 9HSOUTHWEST, 9HNORTHWEST/

EQUIVALENCE (A(1),AA(1,44)),(B(1),AA(1,43)),(C(1),AA(1,42)),
1(D(1),AA(1,41)),(E(1),AA(1,40)),(F(1),AA(1,39)),
1(G(1),AA(1,38)),(BB(1),AA(1,36)),(CC(1),AA(1,34))

NSA1=45 ; NSP=16 ; NT=61 ; NSMF=26 ; NMMF=27 ; NMFCT=28 ; NDM=19

MS=9

NIN1=8

NIN2=17

NIN3=16

NOUT1=6

NOUT2=4

REWIND NOUT2

YEND=0

NULD=0

IPLT=0

QCNT=1

QT=1

IPAG=0

SCALE1=0.001

SCALE2=0.01

SCALE3=0.0001

READ PARAMETERS FOR PLOT VARIABLES

READ (2,149) (II(J),J=1,14)

IF (EOF(2)) 3,1,3

1 DO 2 I=5,14

2 JPLT(I-4)=II(I)

KN1=II(1)

KN2=II(2)

KN3=II(3)
KN4=II(4)
3 CONTINUE

P A R T I - INPUT FROM GAPSIN

READ (NIN1,END=999) DNUM,NY,OR,MR,CR,ISTATE,NSA1,NSA1S,NSP,ICR,ICR
1S,NIC,ISUP7,NSMF,NMFM,NT,NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10,
2NT11,NDM,NMFGT,NY2(1),IMR,MODL,IMFGB

IF (IMR.GT.0) READ (NIN1) (IMRN(I),I=1,IMR),IMRT,(IMRG(J),J=1,IMRT
1),IMRN(11),BLWT,(AK(K),K=1,NSA1)

ISUP7=ISUP7-1
NQ=NY*4
NSPM=NSMF+NMFM
NSPM2=NSPM+2
NSPM3=NSPM+3
NSPM4=NSPM+4
NSPMA=NSPM3+NSA1S
NSMF1=NSMF+1
NSMF2=NSMF+2
NDM1=NDM+(NT9-NSA1)
NMFG1=NMFGT+(NT9-NSA1)
IF (MR(1).LT.1.OR.MR(1).GT.2) NULD=1
IF (NY.EQ.5) IPLT=1

PRINT INPUT FORM

READ (NIN1,END=999) (IR(I),NOR(I),ONAME(I),I=1,NT),(MFG(J),J=1,NDM
1),(MFID(I),MFNAME(I),I=1,NSPM3)
DO 4 I=NSPM4,NSPMA
IJ=I-NSPM3
MFID(I)=MFN(IJ)
MFG(IJ+19)=NSA1+IJ

4 CONTINUE
IF (ISTATE.GT.0) READ (NIN1,END=999) (NOR(I),ONAME(I),I=NT6,NT1)
IF (IMR.LE.0) GO TO 9
IPAG=IPAG+1
WRITE (NOUT1,150) MODL,IPAG,DNUM
WRITE (NOUT1,151) IMR
MRN=0
MRS=0
DO 5 I=1,IMR
MRS=MRN+1
MRNN=IMRN(I)
MRN=MRN+MRNN
5 WRITE (NOUT1,152) I,MRNN,(IMRG(J),J=MRS,MRN)
IF (IMRN(11).LE.0) GO TO 9
IF (IMRN(11).EQ.1) WRITE (NOUT1,153) BLWT
IF (IMRN(11).NE.1) WRITE (NOUT1,154)
DO 8 I=1,IMRT
KK=0
DO 6 K=1,NSA1
IF (NOR(K).NE.IMRG(I)) GO TO 6
KK=K

```
        GO TO 7
6      CONTINUE
      IF (KK.EQ.0) GO TO 999
7      IF (ABS(A(KK)).EQ.0.0) A(KK)=0.0
        A(KK)=A(KK)*SCALE2
8      WRITE (NOUT1,155) NOR(KK),ONAME(KK),A(KK)
9      READ (NIN1,END=999) (A(I),B(I),C(I),I=1,NSA1)
      IF (ISTATE.GT.0) READ (NIN1,END=999) (A(I),B(I),C(I),I=NT5,NT3),(I
      IXT(I),I=1,3)
      ISTD=1+IXT(1)+2*IXT(2)+3*IXT(3)
      READ (NIN1,END=999) (H(J),J=1,6),(II(I),I=1,2)
      DO 10 I=2,6
        H(I)=H(I)*SCALE2
10     NY2(I)=NY2(I)+I-1
        NY1=NY+NY2(I)
        H(I)=H(I)*SCALE2
      DO 11 I=1,4
        IF (QR(I).LT.1.OR.QR(I).GT.2) NULD=1
        IF (ABS(H(I)).EQ.0.0) H(I)=0.0
        IF (I.GT.2) GO TO 11
        IF (IABS(II(I)).EQ.0) II(I)=0
11     CONTINUE
      DO 12 I=1,NSA1
        IF (ABS(A(I)).EQ.0.0) A(I)=0.0
        IF (ABS(B(I)).EQ.0.0) B(I)=0.0
        IF (ABS(C(I)).EQ.0.0) C(I)=0.0
        A(I)=A(I)*SCALE2
        C(I)=C(I)*SCALE2
        IF (ISUP7.GT.0) B(I)=0.0
        IF (ISUP7.GT.0) C(I)=0.0
        IF (ISTATE.EQ.0) GO TO 12
        IF (I.GT.NSA15) GO TO 12
        IF (ABS(A(I+NSA1)).EQ.0.0) A(I+NSA1)=0.0
        IF (ABS(B(I+NSA1)).EQ.0.0) B(I+NSA1)=0.0
        IF (ABS(C(I+NSA1)).EQ.0.0) C(I+NSA1)=0.0
        A(I+NSA1)=A(I+NSA1)*SCALE2
        C(I+NSA1)=C(I+NSA1)*SCALE2
        IF (ISTT.GT.1) B(I+NSA1)=0.0
        IF (ISTT.GT.1) C(I+NSA1)=0.0
        B(I+NSA1)=B(I+NSA1)*SCALE2
12     B(I)=B(I)*SCALE2
        IPAG=IPAG+1
        WRITE (NOUT1,150) MODL,IPAG,DNUM
        WRITE (NOUT1,156)
        WRITE (NOUT1,157) (NOR(I),ONAME(I),A(I),B(I),C(I),I=1,NSA1)
        IPAG=IPAG+1
        IF (ISTATE.EQ.0) GO TO 13
        WRITE (NOUT1,150) MODL,IPAG,DNUM
        WRITE (NOUT1,158)
        WRITE (NOUT1,157) (NOR(I+NSP),ONAME(I+NSP),A(I),B(I),C(I),I=NT5,NT
13)
        IPAG=IPAG+1
13     WRITE (NOUT1,159) NY2(1),MODL,IPAG,DNUM,(H(I),I=1,6)
        WRITE (NOUT1,160) (NY2(I),I=2,6),MR
        WRITE (NOUT1,161) (QR(I),QR(I+4),QR(I+8),QR(I+12),QR(I+16),I=1,4)
        WRITE (NOUT1,162) NY1,(II(I),I=1,2),NY2(1)
```

```
*****
PRINT INPUT FORM - PART II
*****
```

```
READ (NIN1,END=999) (A(I),B(I),C(I),D(I),E(I),F(I),I=1,NSA1)
READ (NIN1,END=999) (H(I),I=1,4),(II(I),I=10,14),(II(I),I=1,6),(II
1(I),I=15,18)
DO 14 I=7,9
14 II(I)=0
IF (ISUP7.EQ.0) GO TO 15
II(ISUP7+6)=1
15 DO 16 I=1,NSA1
IF (ABS(A(I)).EQ.0.0) A(I)=0.0
IF (ABS(B(I)).EQ.0.0) B(I)=0.0
IF (ABS(C(I)).EQ.0.0) C(I)=0.0
IF (ABS(D(I)).EQ.0.0) D(I)=0.0
IF (ABS(E(I)).EQ.0.0) E(I)=0.0
IF (ABS(F(I)).EQ.0.0) F(I)=0.0
IF (ISUP7.GT.0) A(I)=0.0
D(I)=D(I)*SCALE2
E(I)=E(I)*SCALE2
F(I)=F(I)*100.0
IF (I.GT.14) GO TO 16
IF (IABS(II(I)).EQ.0) II(I)=0
IF (I.GT.10) GO TO 16
IF (IABS(ICR(I)).EQ.0) ICR(I)=0
IF (I.GT.4) GO TO 16
IF (ABS(H(I)).EQ.0.0) H(I)=0.0
16 CONTINUE
IPAG=IPAG+1
WRITE (NOUT1,150) MODL,IPAG,DNUM
WRITE (NOUT1,163) (II(I),I=1,6)
IF (ISTATE.GT.0) WRITE (NOUT1,164) (II(I),I=15,18)
WRITE (NOUT1,165) (II(I),I=7,9)
IF (ISTATE.GT.0) WRITE (NOUT1,166) (IXT(I),I=1,3)
IF (ISTATE.GT.0) IPAG=IPAG+1
IF (ISTATE.GT.0) WRITE (NOUT1,150) MODL,IPAG,DNUM
WRITE (NOUT1,167) H(3),H(4),(II(I),I=10,14),(ICR(J),J=1,10)
IF (ISTATE.GT.0) WRITE (NOUT1,168) (ICRS(J),J=1,10)
```

```
*****
PRINT INPUT FORM - PART III
*****
```

```
IF (IMFGB.EQ.0) GO TO 17
READ (NIN1,END=999) ORB,ICRB,ISS,DCSC,IMPHLD
IPAG=IPAG+1
WRITE (NOUT1,150) MODL,IPAG,DNUM
WRITE (NOUT1,169) (DCSC(I),I=1,20)
WRITE (NOUT1,170) ISS,IMPHLD,(ORB(I),I=1,5),(ICRB(J),J=1,14)
```

```
*****
PRINT OPTIONAL FEDERAL INPUT FORMS
*****
```

```
IPAG=IPAG+1
WRITE (NOUT1,171) MODL,IPAG,DNUM
WRITE (NOUT1,172) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),I
1=1,NSA1)
WRITE (NOUT1,173) H(1),H(2)
```

```
*****
PRINT 20 QUARTERS OF PRICES - F.O.
*****
```

```
17 IF (ISUP7.LT.1) GO TO 25
   NQ1=NQ+1
   NQ3=NY+35
   NQ4=NY+15
   IF (ISUP7.GT.1) GO TO 20
   NQ2=NQ1+NQ-1
   READ (NIN1,END=999) ((AA(I,J),I=1,NSA1),J=1,NQ),(H(K),H(K+5),H(K+10),H(K+15),K=1,NY),(H(J+20),H(J+25),H(J+30),H(J+35),J=1,NY)
```

SCALE TO MIL LBS AND \$

```
   CALL SCLE (NSA1,NQ,AA,SCALE2)
   DO 18 I=1,NQ3
18  H(I)=H(I)*SCALE2
   DO 19 I=1,NY
      IJK=(I-1)*4+1
      KLM=NY2(1)+I
      IPAG=IPAG+1
      WRITE (NOUT1,174) MODL,IPAG,DNUM,KLM
   DO 19 J=1,NSA1
19  WRITE (NOUT1,175) NDR(J),ONAME(J),AA(J,IJK),AA(J,IJK+1),AA(J,IJK+2),AA(J,IJK+3)
      IPAG=IPAG+1
      WRITE (NOUT1,176) MODL,IPAG,DNUM
      WRITE (NOUT1,177) (NY2(I+1),H(I),H(I+5),H(I+10),H(I+15),I=1,NY)
      WRITE (NOUT1,178) (NY2(I+1),H(I+20),H(I+25),H(I+30),H(I+35),I=1,NY)
1)
   GO TO 25
```

PRINT GENERATED PRICE SURFACE - F.O.

```
20 READ (NIN1,END=999) ((AA(I,J),I=1,NSA1),J=1,NQ)
   CALL SCLE (NSA1,NQ,AA,SCALE2)
   IF (ISUP7.EQ.3) GO TO 21
   READ (NIN1,END=999) (H(I),I=1,3)
   H(1)=H(1)*SCALE2
   IPAG=IPAG+1
   WRITE (NOUT1,179) MODL,IPAG,DNUM,H(3),H(2),H(1)
   GO TO 23
21 READ (NIN1,END=999) (H(I),H(I+5),H(I+10),H(I+15),I=1,NY),H(NQ4),(A(1(I),A(I+5),A(I+10),A(I+15),I=1,NY),(A(I+20),A(I+25),A(I+30),A(I+35),I=1,NY)
   DO 22 I=1,NQ3
      IF (I.LT.NQ4) H(I)=H(I)*SCALE2
22  A(I)=A(I)*SCALE2
      IPAG=IPAG+1
      WRITE (NOUT1,180) MODL,IPAG,DNUM,H(NQ4),(NY2(I+1),H(I),H(I+5),H(I+10),H(I+15),I=1,NY)
      WRITE (NOUT1,177) (NY2(I+1),A(I),A(I+5),A(I+10),A(I+15),I=1,NY)
      WRITE (NOUT1,178) (NY2(I+1),A(I+20),A(I+25),A(I+30),A(I+35),I=1,NY)
1)
23 DO 24 I=1,NY
      IJK=(I-1)*4+1
      KLM=NY2(1)+I
      IPAG=IPAG+1
      WRITE (NOUT1,181) MODL,IPAG,DNUM,KLM
```

```
DO 24 J=1,NSA1
24 WRITE (NOUT1,175) NOR(J),ONAME(J),AA(J,IJK),AA(J,IJK+1),AA(J,IJK+2
1),AA(J,IJK+3)
```

```
*****
PRINT OPTIONAL STATE INPUT FORMS
*****
```

```
25 IF (ISTATE.LT.1) GO TO 34
READ (NIN1,END=999) (A(I),B(I),C(I),F(I),I=NT5,NT3)
IPAG=IPAG+1
DO 26 I=NT5,NT3
  IF (ABS(A(I)).EQ.0) A(I)=0.0
  IF (ABS(B(I)).EQ.0) B(I)=0.0
  IF (ABS(C(I)).EQ.0) C(I)=0.0
  IF (ABS(F(I)).EQ.0) F(I)=0.0
26 CONTINUE
WRITE (NOUT1,182) MODL,IPAG,DNUM
WRITE (NOUT1,183) (NOR(I+NSP),ONAME(I+NSP),A(I),B(I),C(I),F(I),I=N
1T5,NT3)
```

```
*****
PRINT 20 QUARTERS OF PRICES - S.O.
*****
```

```
IF (ISTT.LE.1) GO TO 33
NQ1=NQ+1
NQ3=NY+35
NQ4=NY+16
NQ2=NQ1+NQ-1
READ (NIN1,END=999) ((AA(I,J),I=NT5,NT3),J=1,NQ)
CALL SCLE (NT3,NQ,AA,SCALE2)
IF (ISTT.GT.2) GO TO 28
DO 27 I=1,NY
  IJK=(I-1)*4+1
  KLM=NY2(1)+I
  IPAG=IPAG+1
  WRITE (NOUT1,184) MODL,IPAG,DNUM,KLM
DO 27 J=NT5,NT3
27 WRITE (NOUT1,185) NOR(J+NSP),ONAME(J+NSP),AA(J,IJK),AA(J,IJK+1),AA
1(J,IJK+2),AA(J,IJK+3)
IF (ISTT.LT.3) GO TO 33
```

```
*****
PRINT GENERATED PRICE SURFACE - S.O.
*****
```

```
28 IF (I14B.NE.1) GO TO 29
READ (NIN1,END=999) (H(I),I=1,3)
H(1)=H(1)*SCALE2
IPAG=IPAG+1
WRITE (NOUT1,186) MODL,IPAG,DNUM,H(3),H(2),H(1)
GO TO 31
29 READ (NIN1,END=999) (H(I),H(I+5),H(I+10),H(I+15),I=1,NY),H(NQ4)
DO 30 I=1,NQ4-1
  H(I)=H(I)*SCALE2
30 CONTINUE
IPAG=IPAG+1
WRITE (NOUT1,187) MODL,IPAG,DNUM,H(NQ4),(NY2(I+1),H(I),H(I+5),H(I+
110),H(I+15),I=1,NY)
```

```
31 DO 32 I=1,NY
    IJK=(I-1)*4+1
    KLM=NY2(1)+I
    IPAG=IPAG+1
    WRITE (NOUT1,188) MODL,IPAG,DNUM,KLM
    DO 32 J=NT5,NT3
32 WRITE (NOUT1,185) NOR(J+NSP),ONAME(J+NSP),AA(J,IJK),AA(J,IJK+1),AA
    1(J,IJK+2),AA(J,IJK+3)
```

```
*****
READ UNREGULATED BASE DATA
*****
```

```
33 IF (ISTATE.GT.0) READ (NIN1,END=999) UNUT
```

```
*****
PRINT QUARTERLY REPORT IF REQUESTED
*****
```

```
34 YEND=YEND+1
    NYRBG=YEND+NY2(1)
35 IF (QR(QCNT).LT.1) GO TO 66
```

```
*****
CHECK TO SEE IF ONLY MOVEMENTS ARE WANTED
*****
```

```
IF (QR(QCNT).EQ.3) GO TO 52
```

```
*****
MERGED ORDER REPORT
*****
```

```
IF (IMR.LE.0) GO TO 38
```

```
READ(NIN2)(PRREC(I),PRREC(I+10),BLNDR(I+10)),BLNDR(I),I=1,IMR)
```

```
READ (NIN2,END=999) (A(I),B(I),C(I),D(I),I=1,IMR)
DO 36 I=1,IMR
    A(I)=A(I)*SCALE1
    B(I)=B(I)*SCALE1
```

```
36 D(I)=D(I)*SCALE2
    IPAG=IPAG+1
    WRITE (NOUT1,189) DNUM,IPAG,QT,NYRBG
    N1=0
    N2=0
    DO 37 I=1,IMR
        N1=N2+1
        N2=N2+IMRN(I)
        WRITE (NOUT1,190) I,A(I),B(I),C(I),D(I)
        WRITE (NOUT1,191) (IMRG(J),J=N1,N2)
37 WRITE (NOUT1,192)
```

```
*****
SUMMARY REPORT
*****
```

```
READ(NIN2)QT,NYRBC,SPREC,XCSPRC,CSPREC,SPCL1,XCSPC1,CIAC1,
1SCL2D,XCMC2,CMCL2,SCL3D,XCMC3,CMCL3,MCL1U,XCMC1U,CMCL1U,
2MCL1P,XCMC1P,CMCL1P,NCL2P,XCHC2P,CCL2P,NCL3P,XCHC3P,CCL3P,
3MBLEND,XCHMB,CMBLEND,SUP,XCSUP,CSUP,SNFP,XCNFP,CSNFP,
```

4SURDDC, XCRDC, CSURDDC, SUFRPS, XCFRP, CSUFRPS, CEXP, XCEXP, CCEXP,
SMRTLP, XCMRTP, CMRTLP, WFRPS, XCMFR, CWFRPS, SMUCP, XCUCP, CMUCP,
GSMFGU, XCMFU, GSMFGU, STRNSR, XCTRNS, CSTRNSR, SOAC1T, XCSOA, CSOAC1T

38 READ (NIN2,END=999) QT, NYRBG, (A(I),B(I),C(I), I=1,20)
QT=QT-5

SCALE TO MIL LBS AND \$

DO 39 I=1,4
A(I)=A(I)*SCALE1
39 B(I)=B(I)*SCALE1
DO 41 I=6,14
IF (I.EQ.10) GO TO 40
IF (I.GT.12) GO TO 40
A(I)=A(I)*SCALE2
B(I)=B(I)*SCALE2
GO TO 41
40 A(I)=A(I)*SCALE3
B(I)=B(I)*SCALE3
41 CONTINUE
DO 42 I=19,20
A(I)=A(I)*SCALE3
42 B(I)=B(I)*SCALE3
IPAG=IPAG+1
WRITE (NOUT1,193) DNUM, IPAG, QT, NYRBG
WRITE (NOUT1,194) (A(I),B(I),C(I), I=1,20)

CHECK TO SEE IF SUMMARY IS ALL THAT IS WANTED

IF (QR(QCNT).EQ.4) GO TO 66

FEDERAL ORDER REPORT - 1

READ(NIN2)(PRREC(IR(I)), CHGPRR(IR(I)), CL1UT(IR(I)),
1NCL1P(IR(I)), NCL2P, NCL3P, BLEND(IR(I),6), I=1,NSA1),
2SFREC, CSPRES, MCL1U, MCL1P, NCL2P, NCL3P, MBLEND

READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),F(I),G(I), I=1,NSA1),
1(H(I), I=1,7)

SCALE TO MIL LBS AND \$

DO 43 I=1,NSA1
A(I)=A(I)*SCALE1
D(I)=D(I)*SCALE2
E(I)=E(I)*SCALE2
F(I)=F(I)*SCALE2
43 G(I)=G(I)*SCALE2
H(I)=H(I)*SCALE1
DO 44 I=4,7
44 H(I)=H(I)*SCALE2
IPAG=IPAG+1
WRITE (NOUT1,195) DNUM, IPAG, QT, NYRBG
WRITE (NOUT1,196) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),G
1(I), I=1,NSA1)
WRITE (NOUT1,197) (H(I), I=1,7)

FEDERAL ORDER REPORT - 2 (SALES)

READ(NIN2)(PCL1(IR(I)),CHGCL1S(IR(I)),PCL2(IR(I)),
1CHGCL2D(IR(I)),TCL3(IR(I)),CHGCL3Q(IR(I)),I=1,NSA1),
2SPCL1,SCL2D,SCL3Q,CSPCL1,CMCL2,CMCL3,SMC20

READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),F(I),I=1,NSA1),(H(I)
1,I=1,7)

SCALE TO MIL LBS AND \$

DO 45 I=1,NSA1
A(I)=A(I)*SCALE1
C(I)=C(I)*SCALE1
45 E(I)=E(I)*SCALE1
H(1)=H(1)*SCALE1
H(3)=H(3)*SCALE1
H(2)=H(2)*SCALE1
H(7)=H(7)*SCALE1
IPAG=IPAG+1
WRITE (NOUT1,198) DNUM,IPAG,QT,NYRBC
WRITE (NOUT1,199) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),I
1=1,NSA1)
WRITE (NOUT1,200) (H(I),I=1,7)

MILK PRODUCTION REPORT

READ(NIN2)(PPEC(IR(I)),CHGPRR(IR(I)),BLEND(IR(I),6),
1NFP(I),RODC(IR(I)),I=1,NSA1),SPREC
2CSPREC,MBLEND,SNFP,SURODC

READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),I=1,NSA1),(H(I),I=1,
15)

SCALE TO MIL LBS AND \$

DO 46 I=1,NSA1
C(I)=C(I)*SCALE2
D(I)=D(I)*SCALE2
E(I)=E(I)*SCALE2
46 A(I)=A(I)*SCALE1
H(1)=H(1)*SCALE1
DO 47 I=3,5
47 H(I)=H(I)*SCALE2
IPAG=IPAG+1
WRITE (NOUT1,201) DNUM,IPAG,QT,NYRBC
WRITE (NOUT1,202) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),I=1,NS
1A1)
WRITE (NOUT1,203) (H(I),I=1,5)

PROCESSING REPORT

READ(NIN2)(GRC1(IR(I)),UTCP(IR(I)),TRMC(IR(I)),TRANSRP(IR(I)),
1GALC1T(I),TCWT(IR(I)),I=1,NSA1),
2SMUCP,STRMC,STRNSD,SOAC1D,STCCWT,SGRC1

```
READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),F(I),I=1,NSA1),(H(I),I=1,6)
```

SCALE TO MIL LBS AND \$

```
DO 48 I=1,NSA1
  A(I)=A(I)*SCALE1
  C(I)=C(I)*SCALE2
  D(I)=D(I)*SCALE2
  E(I)=E(I)*SCALE2
48 F(I)=F(I)*SCALE2
  H(2)=H(2)*SCALE2
  H(3)=H(3)*SCALE2
  H(4)=H(4)*SCALE2
  H(5)=H(5)*SCALE2
  H(6)=H(6)*SCALE1
  IPAG=IPAG+1
  WRITE (NOUT1,204) DNUM,IPAG,QT,NYRBC
  WRITE (NOUT1,205) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),I=1,NSA1)
  WRITE (NOUT1,206) (H(I),I=1,6)
```

MANUFACTURING CENTER REPORT

```
READ(NIN2)(TMMC(I),UMCP(I),I=1,NSPM),SCL30,SMFGU
```

```
READ (NIN2,END=999) (BB(I),CC(I),I=1,NSPM),H(1),H(2)
```

SCALE TO MIL LBS

```
DO 49 I=1,NSPM
49 BB(I)=BB(I)*SCALE1
  H(1)=H(1)*SCALE1
  IPAG=IPAG+1
  WRITE (NOUT1,207) DNUM,IPAG,QT,NYRBC
  DO 50 I=1,NSPM2
    IA=I
    IF (I.EQ.NSMF1.OR.I.EQ.NSMF2) GO TO 50
    IF (I.GT.NSMF2) IA=I-2
    IF (BB(IA).EQ.C.0) GO TO 50
    WRITE (NOUT1,208) MFID(I),MFNAME(I),BB(IA),CC(IA)
50 CONTINUE
  WRITE (NOUT1,209) H(1),H(2)
```

CONSUMPTION REPORT

```
READ(NIN2)(CIACL1(IR(I)),CHGCIAC1(IR(I)),INRPR(I),FRPS(IR(I)),I=1,NSA1),SCL1D,CIAC1,MRTLP,WFRPS
```

```
READ (NIN2,END=999) (A(I),B(I),C(I),D(I),I=1,NSA1),(H(I),I=1,4)
```

SCALE TO MIL LBS AND \$

```
DO 51 I=1,NSA1
51 A(I)=A(I)*SCALE1
  H(1)=H(1)*SCALE1
  IF (NY.EQ.1) WRITE (NOUT2) (C(I),I=1,NSA1)
  IPAG=IPAG+1
```

```
WRITE (NOUT1,210) DNUM,IPAG,QT,NYRBG
WRITE (NOUT1,211) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),I=1,NSA1)
WRITE (NOUT1,212) (H(I),I=1,4)
```

```
*****
CHECK TO SEE IF MOVEMENTS ARE WANTED
*****
```

```
IF (QR(QCNT).GT.1) GO TO 66
```

```
*****
RAW MILK MOVEMENTS FROM SUPPLY AREAS 1
*****
```

```
READ(NIN2)((RMMD(IR(I),IR(J)),J=1,NT9),I=1,NT9),MPPC,MOPC
```

```
52 READ (NIN2,END=999) ((AA(I,J),J=1,NT9),I=1,NT9),(H(I),I=1,2)
```

```
SCALE TO MIL LBS AND $
```

```
DO 53 I=1,2
```

```
53 H(I)=H(I)*SCALE1
```

```
IPAG=IPAG+1
```

```
WRITE (NOUT1,213) DNUM,IPAG,QT,NYRBG
```

```
DO 55 I=1,NT9
```

```
IJ=I
```

```
IF (I.GT.NSA1) IJ=I+NSP
```

```
IK=0
```

```
DO 54 K=1,NT9
```

```
AA(I,K)=AA(I,K)*SCALE1
```

```
IF (AA(I,K).LE.0.0) GO TO 54
```

```
IK=IK+1
```

```
II(IK)=K
```

```
III(IK)=K
```

```
IF (K.GT.NSA1) III(IK)=K+NSP
```

```
54 CONTINUE
```

```
IF (IK.LE.0) GO TO 55
```

```
WRITE (NOUT1,214) NOR(IJ),(NOR(III(J)),AA(I,II(J)),J=1,IK)
```

```
55 CONTINUE
```

```
WRITE (NOUT1,215) (H(I),I=1,2)
```

```
*****
RAW MILK MOVEMENTS TO SUPPLY PLANTS 1
*****
```

```
READ(NIN2)((RMMS(I,IR(J)),J=1,NT9),I=1,NSP),SSGPC,TMOP
```

```
READ (NIN2,END=999) ((AA(I,J),J=1,NT9),I=1,NSP),H(1),H(2)
```

```
SCALE TO MIL LBS
```

```
H(1)=H(1)*SCALE1
```

```
H(2)=H(2)*SCALE1
```

```
IPAG=IPAG+1
```

```
WRITE (NOUT1,216) DNUM,IPAG,QT,NYRBG
```

```
DO 57 I=1,NSP
```

```
IK=0
```

```
DO 56 K=1,NT9
```

```
AA(I,K)=AA(I,K)*SCALE1
```

```
IF (AA(I,K).LE.0.0) GO TO 56
```

```
IK=IK+1
```

```
II(IK)=K
```

```
      III(IK)=K
      IF (K.GT.NSA1) III(IK)=K+NSP
56  CONTINUE
      IF (IK.LE.0) GO TO 57
      WRITE (NOUT1,214) NOR(I+NSA1),(NOR(III(J)),AA(I,II(J)),J=1,IK)
57  CONTINUE
      WRITE (NOUT1,217) H(1),H(2)
```

RAW MILK MOVEMENTS TO MANUFACTURING CENTERS 2

```
READ(NIN2)((MMFG(I,J),J=1,NT10),I=1,NSA1),  
1((MMFG(I,J),J=1,NT10),I=NT6,NT1),SMDS,SMDS  
READ (NIN2,END=999) ((AA(I,J),J=1,NT10),I=1,NSA1),((AA(I,J),J=1,NT  
110),I=NT5,NT3),H(1),H(2)
```

SCALE TO MIL LBS AND \$

```
IPAG=IPAG+1  
WRITE (NOUT1,218) DNUM,IPAG,GT,NYRBC  
H(1)=H(1)*SCALE1  
H(2)=H(2)*SCALE1  
DO 60 I=1,NT9  
  IJK=IR(I)  
  IF (I.GT.NSA1) IJK=I  
  IJ=I  
  IF (I.GT.NSA1) IJ=I+NSP  
  IK=0  
  IKNT=0  
  DO 58 J=1,NOM1  
    IF (IJK.NE.MFG(J)) GO TO 58  
    IKNT=NSMF2  
58  CONTINUE  
    DO 59 K=1,NMFG1  
      AA(IJK,K)=AA(IJK,K)*SCALE1  
      IF (AA(IJK,K).LE.0.0) GO TO 59  
      IF (IKNT.EQ.K) IKNT=NSMF2  
      IK=IK+1  
      II(IK)=K  
      KKK=K+IKNT  
      IF (I.GT.NSA1.AND.K.LE.NMFGT) KKK=K  
      II(IK+NSMF2)=KKK  
59  CONTINUE  
      IF (IK.LE.0) GO TO 60  
      WRITE (NOUT1,219) NOR(IJ),(MFD(II(J+NSMF2)),AA(IJK,II(J)),J=1,  
1  IK)  
60  CONTINUE  
      WRITE (NOUT1,220) H(1),H(2)
```

SUPPLY PLANT MILK MOVEMENTS TO MANUFACTURING CENTERS

```
READ(NIN2)((MMFG(I+NSA1,J),J=1,NT10),I=1,NSP),SMSP,SMSP  
READ (NIN2,END=999) ((AA(I,J),J=1,NT10),I=1,NSP),H(1),H(2)
```

SCALE TO MIL LBS

IPAG=IPAG+1


```
WRITE (NOUT1,221) DNUM,IPAG,QT,NYRBG
H(1)=H(1)*SCALE1
H(2)=H(2)*SCALE1
DO 63 I=1,NSP
  IK=0
  IKNT=0
  DO 61 J=1,NT9
    JJ=J
    IF (J.GT.NSA1) JJ=J+NSP
    IJK=J
    IF (J.LE.NSA1) IJK=IR(J)
    DO 61 KK=1,NDM1
      IF (NOR(I+NSA1).EQ.NOR(JJ).AND.IJK.EQ.MFG(KK)) IKNT=NSMF2
61  CONTINUE
    DO 62 K=1,NT10
      AA(I,K)=AA(I,K)*SCALE1
      IF (AA(I,K).LE.0.0) GO TO 62
      IF (IKNT.EQ.K) IKNT=NSMF2
      IK=IK+1
      II(IK)=K
      KKK=K+IKNT
      II(IK+NSMF2)=KKK
62  CONTINUE
      IF (IK.LE.0) GO TO 63
      WRITE (NOUT1,219) NOR(I+NSA1),(MFID(II(J+NSMF2)),AA(I,II(J)),J=
1      1,IK)
63 CONTINUE
  WRITE (NOUT1,220) H(1),H(2)

      *****
      PACKAGED MILK MOVEMENTS
      *****
READ(NIN2)((PMM(I,J),J=1,NT9),I=1,NT9),SMPC,MOCC
READ (NIN2,END=999) ((AA(I,J),J=1,NT9),I=1,NT9),H(1),H(2)

      SCALE TO MILL LBS

H(1)=H(1)*SCALE1
H(2)=H(2)*SCALE1
IPAG=IPAG+1
WRITE (NOUT1,222) DNUM,IPAG,QT,NYRBG
DO 65 I=1,NT9
  IJ=I
  IF (I.GT.NSA1) IJ=I+NSP
  IK=0
  DO 64 K=1,NT9
    AA(I,K)=AA(I,K)*SCALE1
    IF (AA(I,K).LE.0.0) GO TO 64
    IK=IK+1
    II(IK)=K
    III(IK)=K
    IF (K.GT.NSA1) III(IK)=K+NSP
64  CONTINUE
    IF (IK.LE.0) GO TO 65
    WRITE (NOUT1,214) NOR(IJ),(NOR(III(J)),AA(I,II(J)),J=1,IK)
65 CONTINUE
  WRITE (NOUT1,223) H(1),H(2)
66 CONTINUE
```

READ UNREGULATED DATA: SCALE AND SUM DATA

```
IF (ISTATE.EQ.0.OR.IMFCB.EQ.0) GO TO 69
READ (NIN3,END=999) UNREG,UNC1S,UNGBP,UNNBP,UNRODC
DO 67 K=1,MS
  UNREG(K)=UNREG(K)*SCALE1
  UNC1S(K)=UNC1S(K)*SCALE1
  UNGBP(K)=UNGBP(K)*SCALE2
  UNNBP(K)=UNNBP(K)*SCALE2
  UNRODC(K)=UNRODC(K)*SCALE2
  AUNREG(YEND+1,K)=AUNREG(YEND+1,K)+UNREG(K)
  AUNC1S(YEND+1,K)=AUNC1S(YEND+1,K)+UNC1S(K)
  AUNGBP(YEND+1,K)=AUNGBP(YEND+1,K)+UNGBP(K)/4.
  AUNNBP(YEND+1,K)=AUNNBP(YEND+1,K)+UNNBP(K)/4.
  AUNROD(YEND+1,K)=AUNROD(YEND+1,K)+UNRODC(K)/4.
67 CONTINUE
IF (QR(QCNT).EQ.0.OR.QR(QCNT).GE.3) GO TO 69
```

UNREGULATED QTRLY GRADE A REPORT

```
W3=0.
W2=W3
W1=W2
S2=W1
S1=S2
IPAG=IPAG+1
WRITE (NOUT1,224) DNUM,IPAG,QT,NYRBC
WRITE (NOUT1,225)
DO 68 I=1,MS
  WRITE (NOUT1,226) NAME(I),UNREG(I),UNC1S(I),UNGBP(I),UNNBP(I),U
  1  NRODC(I)
  S1=S1+UNREG(I)
  S2=S2+UNC1S(I)
  W1=W1+UNGBP(I)
  W2=W2+UNNBP(I)
68 W3=W3+UNRODC(I)
  W1=W1/FLOAT(MS)
  W2=W2/FLOAT(MS)
  W3=W3/FLOAT(MS)
  WRITE (NOUT1,227) S1,S2,W1,W2,W3,UNUT
69 QT=QT+1
  QCNT=QCNT+1
  IF (QT.NE.5) GO TO 35
  QT=1
```

PRINT ANNUAL REPORT IF REQUESTED

```
IF (MR(YEND).LT.1) GO TO 101
```

CHECK TO SEE IF ONLY MOVEMENTS ARE WANTED

```
IF (MR(YEND).EQ.3) GO TO 86
```

MERGED ORDER REPORT

```
IF (IMR.LE.0) GO TO 72
READ(NIN2)(PRECAM(I),UPAMRG(I),BLAMR(I+10),BLAMR(I),I=1,IMR)
READ (NIN2,END=999) (A(I),B(I),C(I),D(I),I=1,IMR)
DO 70 I=1,IMR
  A(I)=A(I)*SCALE1
  B(I)=B(I)*SCALE1
70 D(I)=D(I)*SCALE2
  IPAG=IPAG+1
  WRITE (NOUT1,228) DNUM,IPAG,NYRBC
  N1=0
  N2=0
  DO 71 I=1,IMR
    N1=N2+1
    N2=N2+IMRN(I)
    WRITE (NOUT1,190) I,A(I),B(I),C(I),D(I)
    WRITE (NOUT1,191) (IMRG(J),J=N1,N2)
71 WRITE (NOUT1,192)
```

SUMMARY REPORT

```
READ(NIN2)NYRBC,ASPRRC(2),XCHAPR,CHASPR,ASPCL1(2),XCHAPC,CHASC1,
1ASCL2(2),XCHAC2,CHASC2,ASCL3(2),XCHAC3,CHASC3,AMC1U(2),
2XCHAU,CHASC1U,AMC1P(2),XCHAC1,CHAMC1P,AMC2P(2),XCHAC2,CHAMC2P,
3AMC3P(2),XCHAC3,CHAMC3P,AMBLND(2),XCHABLND,CHABLND,AUP(2),
4XCHAUP,CHAUP,ASNFP(2),ACHANFP,CHASNFP,ASRODC(2),XCHARDC,
5CHASRDC,ASFRPS(2),XCHAFRP,CHAFRPS,ACEXP(2),XCHACEX,CHACEXP,
6AMRTL(2),XCHAMRT,CHAMRTL,AMFRPS(2),XCHAWF,CHAWRPS,AMUCP(2),
7XCHACP,CHAUCP,AMFGU(2),XCHACU,CHASFGU,ASTRNSR(2),XCHATR,
8CHASTRN,ASOACIT(2),XCHASO,CHASOA
72 READ (NIN2,END=999) NYRBC,(A(I),B(I),C(I),I=1,20)
SCALE TO MIL LBS AND $
DO 73 I=1,4
  A(I)=A(I)*SCALE1
73 B(I)=B(I)*SCALE1
  DO 75 I=8,14
    IF (I.EQ.10) GO TO 74
    IF (I.GT.12) GO TO 74
    A(I)=A(I)*SCALE2
    B(I)=B(I)*SCALE2
    GO TO 75
74 A(I)=A(I)*SCALE3
  B(I)=B(I)*SCALE3
75 CONTINUE
  DO 76 I=19,20
    A(I)=A(I)*SCALE3
76 B(I)=B(I)*SCALE3
  IPAG=IPAG+1
  WRITE (NOUT1,229) DNUM,IPAG,NYRBC
  WRITE (NOUT1,194) (A(I),B(I),C(I),I=1,20)
```

CHECK TO SEE IF SUMMARY IS ALL THAT IS WANTED

IF (MR(YEND).EQ.4) GO TO 101

FEDERAL ORDER REPORT - 1

READ(NIN2)(APREC(IR(I),2),ACHCPRR(IR(I)),ACLIUT(IR(I)),
1ACLIP(IR(I)),AMC2P(2),AMC3P(2),ABLEND(IR(I)),I=1,NSA1),
2ASPRRC(2),CHASPR,AMC1U(2),AMC1P(2),AMC2P(2),AMC3P(2),AMBLND(2))

READ (NIN2,END=995) (A(I),B(I),C(I),D(I),E(I),F(I),G(I),I=1,NSA1),
1(H(I),I=1,7))

SCALE TO MIL LBS AND \$

DO 77 I=1,NSA1
A(I)=A(I)*SCALE1
D(I)=D(I)*SCALE2
E(I)=E(I)*SCALE2
F(I)=F(I)*SCALE2
77 G(I)=G(I)*SCALE2
H(I)=H(I)*SCALE1
DO 78 I=4,7
78 H(I)=H(I)*SCALE2
IPAG=IPAG+1
WRITE (NOUT1,230) DNUM,IPAG,NYRBC
WRITE (NOUT1,196) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),G
1(I),I=1,NSA1)
WRITE (NOUT1,197) (H(I),I=1,7)

FEDERAL ORDER REPORT - 2 (SALES)

READ(NIN2)(APCL1(IR(I),2),ACHGC1S(IR(I)),ACL2D(IR(I),2),
1ACHGC2D(IR(I)),ACL3D(IR(I),2),ACHGC3D(IR(I)),I=1,NSA1),
2ASPCL1(2),ASCL2D(2),ASCL3D(2),CHASPC,CHASC2,CHASC3,AMC2D

READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),F(I),I=1,NSA1),(H(I)
1,I=1,7)

SCALE TO MIL LBS AND \$

DO 79 I=1,NSA1
A(I)=A(I)*SCALE1
C(I)=C(I)*SCALE1
79 E(I)=E(I)*SCALE1
H(1)=H(1)*SCALE1
H(3)=H(3)*SCALE1
H(2)=H(2)*SCALE1
H(7)=H(7)*SCALE1
IPAG=IPAG+1
WRITE (NOUT1,231) DNUM,IPAG,NYRBC
WRITE (NOUT1,199) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),I
1=1,NSA1)
WRITE (NOUT1,200) (H(I),I=1,7)

```
*****  
MILK PRODUCTION REPORT  
*****  
READ(NIN2)(APREC(IR(I)),ACHGFR(IR(I)),  
1ABLEND(IR(I)),ANFP(IR(I)),ARODC(IR(I)),I=1,NSA1),  
2ASPRC(2),CHASPR,AMBLND(2),ASNFP(2),ASRODC(2)  
  
READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),I=1,NSA1),(H(I),I=1,  
15)  
  
SCALE TO MIL LBS AND $  
  
DO 80 I=1,NSA1  
C(I)=C(I)*SCALE2  
D(I)=D(I)*SCALE2  
E(I)=E(I)*SCALE2  
80 A(I)=A(I)*SCALE1  
H(I)=H(I)*SCALE1  
DO 81 I=3,5  
81 H(I)=H(I)*SCALE2  
IPAG=IPAG+1  
WRITE (NOUT1,232) DNUM,IPAG,NYRBG  
WRITE (NOUT1,202) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),I=1,NS  
1A1)  
WRITE (NOUT1,203) (H(I),I=1,5)
```

```
*****  
PROCESSING REPORT  
*****  
READ(NIN2)(AGRC1(IR(I)),AUTCP(IR(I)),ATRCM(IR(I)),  
1ATRANSR(IR(I)),ADAC1T(IR(I)),ATCCWT(IR(I)),I=1,NSA1),  
2AMUCP(2),ASTRMC,ASTRMSD,ASOAC1D,ASTCCWT,ASGRC1  
  
READ (NIN2,END=999) (A(I),B(I),C(I),D(I),E(I),F(I),I=1,NSA1),(H(I)  
1,I=1,6)
```

```
SCALE TO MIL LBS AND $  
  
DO 82 I=1,NSA1  
A(I)=A(I)*SCALE1  
C(I)=C(I)*SCALE2  
D(I)=D(I)*SCALE2  
E(I)=E(I)*SCALE2  
82 F(I)=F(I)*SCALE2  
H(2)=H(2)*SCALE2  
H(3)=H(3)*SCALE2  
H(4)=H(4)*SCALE2  
H(5)=H(5)*SCALE2  
H(6)=H(6)*SCALE1  
  
IPAG=IPAG+1  
WRITE (NOUT1,233) DNUM,IPAG,NYRBG  
WRITE (NOUT1,205) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),E(I),F(I),I  
1=1,NSA1)  
WRITE (NOUT1,206) (H(I),I=1,6)
```

```
*****  
MANUFACTURING CENTER REPORT  
*****  
READ(NIN2)(ATMDC(I),AUMCP(I),I=1,NSFM),ASCL3Q(2),ASMFGU(2)
```

READ (NIN2,END=999) (BB(I),CC(I),I=1,NSPM),H(1),H(2)

SCALE TO MIL LBS

```
DO 83 I=1,NSPM
83 BB(I)=BB(I)*SCALE1
   H(1)=H(1)*SCALE1
   IPAG=IPAG+1
   WRITE (NOUT1,234) DNUM,IPAG,NYRBG
   DO 84 I=1,NSPM2
     IA=I
     IF (I.EQ.NSMF1.OR.I.EQ.NSMF2) GO TO 84
     IF (I.GT.NSMF2) IA=I-2
     IF (BB(IA).EQ.0.0) GO TO 84
     WRITE (NOUT1,208) MFID(I),MFNAME(I),BB(IA),CC(IA)
84 CONTINUE
   WRITE (NOUT1,209) H(1),H(2)
```

CONSUMPTION REPORT

```
READ(NIN2)(AIAC1S(IR(I)),ACH1AC1(IR(I)),ARPR(IR(I)),
IAFRPS(IR(I)),I=1,NSA1),AS1AC1(2),CHASC1,AMRTL,AWFRPS(2)
```

READ (NIN2,END=999) (A(I),B(I),C(I),D(I),I=1,NSA1),(H(I),I=1,4)

SCALE TO MIL LBS AND \$

```
DO 85 I=1,NSA1
85 A(I)=A(I)*SCALE1
   H(1)=H(1)*SCALE1
   IF (NY.EQ.1) WRITE (NOUT2) (C(I),I=1,NSA1)
   IPAG=IPAG+1
   WRITE (NOUT1,235) DNUM,IPAG,NYRBG
   WRITE (NOUT1,211) (NOR(I),ONAME(I),A(I),B(I),C(I),D(I),I=1,NSA1)
   WRITE (NOUT1,212) (H(I),I=1,4)
```

CHECK TO SEE IF MOVEMENTS ARE WANTED

IF (MR(YEND).GT.1) GO TO 101

RAW MILK MOVEMENTS FROM SUPPLY AREAS 1

```
READ(NIN2)((ARMMD(IR(I),IR(J)),J=1,NT9),I=1,NT9),AMPPC,MOPC
```

86 READ (NIN2,END=999) ((AA(I,J),J=1,NT9),I=1,NT9),(H(I),I=1,2)

SCALE TO MIL LBS AND \$

```
DO 87 I=1,2
87 H(I)=H(I)*SCALE1
   IPAG=IPAG+1
   WRITE (NOUT1,236) DNUM,IPAG,NYRBG
   DO 89 I=1,NT9
     IJ=I
     IF (I.GT.NSA1) IJ=I+NSP
```

```
IK=0
DO 88 K=1,NT9
  AA(I,K)=AA(I,K)*SCALE1
  IF (AA(I,K).LE.0.0) GO TO 88
  IK=IK+1
  II(IK)=K
  III(IK)=K
  IF (K.GT.NSA1) III(IK)=K+NSP
88 CONTINUE
  IF (IK.LE.0) GO TO 88
  WRITE (NOUT1,214) NOR(IJ),(NOR(III(J)),AA(I,II(J)),J=1,IK)
89 CONTINUE
  WRITE (NOUT1,215) (H(I),I=1,2)

  *****
  RAW MILK MOVEMENTS TO SUPPLY PLANTS 1
  *****
  READ(NIN2)((ARIMS(IR(I),IR(J)),J=1,NT9),I=1,NSP),ASOPC,ATMOP
  READ (NIN2,END=999) ((AA(I,J),J=1,NT9),I=1,NSP),H(1),H(2)

  SCALE TO MIL LBS

  H(1)=H(1)*SCALE1
  H(2)=H(2)*SCALE1
  IPAG=IPAG+1
  WRITE (NOUT1,237) DNUM,IPAG,NYRBC
  DO 91 I=1,NSP
    IK=0
    DO 90 K=1,NT9
      AA(I,K)=AA(I,K)*SCALE1
      IF (AA(I,K).LE.0.0) GO TO 90
      IK=IK+1
      II(IK)=K
      III(IK)=K
      IF (K.GT.NSA1) III(IK)=K+NSP
90 CONTINUE
      IF (IK.LE.0) GO TO 91
      WRITE (NOUT1,214) NOR(I+NSA1),(NOR(III(J)),AA(I,II(J)),J=1,IK)
91 CONTINUE
      WRITE (NOUT1,217) H(1),H(2)

      *****
      RAW MILK MOVEMENTS TO MANUFACTURING CENTERS 2
      *****
      READ(NIN2)((AMMFG(I,J),J=1,NT10),I=1,NSA1),
      1((AMMFG(I,J),J=1,NT10),I=NT5,NT1),ASMDS,ASMDS

      READ (NIN2,END=999) ((AA(I,J),J=1,NT10),I=1,NSA1),((AA(I,J),J=1,NT
      110),I=NT5,NT3),H(1),H(2)

      SCALE TO MIL LBS AND $

      IPAG=IPAG+1
      WRITE (NOUT1,238) DNUM,IPAG,NYRBC
      H(1)=H(1)*SCALE1
      H(2)=H(2)*SCALE1
      DO 94 I=1,NT9
        IJK=I
        IF (I.LE.NSA1) IJK=IR(I)
```

```
IJ=I
IF (I.GT.NSA1) IJ=I+NSP
IK=0
IKNT=0
DO 92 J=1,NDM1
  IF (IJK.EQ.MFG(J)) IKNT=NSMF2
92 CONTINUE
DO 93 K=1,NMFG1
  AA(IJK,K)=AA(IJK,K)*SCALE1
  IF (AA(IJK,K).LE.0.0) GO TO 93
  IF (IKNT.EQ.K) IKNT=NSMF2
  IK=IK+1
  II(IK)=K
  KKK=K+IKNT
  IF (I.GT.NSA1.AND.K.LE.NMFGT) KKK=K
  II(IK+NSMF2)=KKK
93 CONTINUE
IF (IK.LE.0) GO TO 94
WRITE (NOUT1,219) NOR(IJ),(MFD(II(J+NSMF2)),AA(IJK,II(J)),J=1,
1 IK)
94 CONTINUE
WRITE (NOUT1,220) H(1),H(2)
```

SUPPLY PLANT MOVEMENTS TO MANUFACTURING CENTERS

```
READ(NIN2)((AMMFG(I+NSA1,J),J=1,NT10),I=1,NSP),ASMSP,ASMSPD
READ (NIN2,END=999) ((AA(I,J),J=1,NT10),I=1,NSP),H(1),H(2)
SCALE TO MIL LBS
```

```
IPAG=IPAG+1
WRITE (NOUT1,239) DNUM,IPAG,NYR8G
H(1)=H(1)*SCALE1
H(2)=H(2)*SCALE1
DO 97 I=1,NSP
```

```
  IK=0
  IKNT=0
  DO 95 J=1,NT9
    JJ=J
    IF (J.GT.NSA1) JJ=J+NSP
    IJK=J
    IF (J.LE.NSA1) IJK=IR(J)
    DO 95 KK=1,NDM1
      IF (NOR(I+NSA1).EQ.NOR(JJ).AND.IJK.EQ.MFG(KK)) IKNT=NSMF2
95 CONTINUE
  DO 96 K=1,NT10
    AA(I,K)=AA(I,K)*SCALE1
    IF (AA(I,K).LE.0.0) GO TO 96
    IF (IKNT.EQ.K) IKNT=NSMF2
    IK=IK+1
    II(IK)=K
    KKK=K+IKNT
    II(IK+NSMF2)=KKK
96 CONTINUE
  IF (IK.LE.0) GO TO 97
  WRITE (NOUT1,219) NOR(I+NSA1),(MFD(II(J+NSMF2)),AA(I,II(J)),J=
1 I,IK)
97 CONTINUE
```



```
WRITE (NOUT1,220) H(1),H(2)

*****
PACKAGED MILK MOVEMENTS
*****
READ(NIN2)((APMM(IR(I), IR(J)), J=1, NT9), I=1, NT9), ASMPC, AMOCC
READ (NIN2,END=999) ((AA(I,J), J=1, NT9), I=1, NT9), H(1), H(2)

SCALE TO MILL LBS

DO 98 I=1, NT9
DO 98 J=1, NT9
98 AA(I,J)=AA(I,J)*SCALE1
H(1)=H(1)*SCALE1
H(2)=H(2)*SCALE1
IPAG=IPAG+1
WRITE (NOUT1,240) DNUM, IPAG, NYRBG
DO 100 I=1, NT9
IJ=I
IF (I.GT.NSA1) IJ=I+NSP
IK=0
DO 99 K=1, NT9
IF (AA(I,K).LE.0.0) GO TO 99
IK=IK+1
II(IK)=K
III(IK)=K
IF (K.GT.NSA1) III(IK)=K+NSP
99 CONTINUE
IF (IK.LE.0) GO TO 100
WRITE (NOUT1,214) NOR(IJ), (NOR(III(J)), AA(I, II(J)), J=1, IK)
100 CONTINUE
WRITE (NOUT1,223) H(1), H(2)
101 CONTINUE

*****
READ UNREGULATED BASE DATA: SCALE AND SUM DATA
*****

IF (ISTATE.EQ.0.OR.IMFGB.EQ.0) GO TO 106
IF (YEND.GT.1) GO TO 104
DO 102 J=1, 4
DO 102 K=1, MS
1 K)
READ (NIN1,END=999) UNREG(K), UNC1S(K), UNGBP(K), UNNBP(K), UNRODC(
UNREG(K)=UNREG(K)*SCALE1
UNC1S(K)=UNC1S(K)*SCALE1
UNGBP(K)=UNGBP(K)*SCALE2
UNNBP(K)=UNNBP(K)*SCALE2
UNRODC(K)=UNRODC(K)*SCALE2
AUNREG(1,K)=AUNREG(1,K)+UNREG(K)
AUNC1S(1,K)=AUNC1S(1,K)+UNC1S(K)
AUNGBP(1,K)=AUNGBP(1,K)+UNGBP(K)/4.
AUNNBP(1,K)=AUNNBP(1,K)+UNNBP(K)/4.
AUNROD(1,K)=AUNROD(1,K)+UNRODC(K)/4.
102 CONTINUE

*****
UNREGULATED GRADE A BASE REPORT
*****
```

```
W3=0.
W2=W3
W1=W2
S2=W1
S1=S2
IPAG=IPAG+1
NYBASE=NYRBC-1
WRITE (NOUT1,241) DNUM,IPAG,NYBASE
WRITE (NOUT1,225)
DO 103 I=1,MS
  WRITE (NOUT1,226) NAME(I),AUNREG(1,I),AUNC1S(1,I),AUNGBP(1,I),A
1  UNNBP(1,I),AUNROD(1,I)
  S1=S1+AUNREG(1,I)
  S2=S2+AUNC1S(1,I)
  W1=W1+AUNGBP(1,I)
  W2=W2+AUNNBP(1,I)
103 W3=W3+AUNROD(1,I)
  W1=W1/FLOAT(MS)
  W2=W2/FLOAT(MS)
  W3=W3/FLOAT(MS)
  WRITE (NOUT1,227) S1,S2,W1,W2,W3,UNUT
```

```
*****
UNREGULATED ANNUAL GRADE A REPORT
*****
```

104 CONTINUE

IF(MR(YEND).EQ.0.OR.MR(YEND).GE.3) GO TO 203

```
IPAG=IPAG+1
WRITE (NOUT1,241) DNUM,IPAG,NYRBC
WRITE (NOUT1,223)
W3=0.
W2=W3
W1=W2
S2=W1
S1=S2
DO 105 I=1,MS
  WRITE (NOUT1,226) NAME(I),AUNREG(YEND+1,I),AUNC1S(YEND+1,I),AUN
1  GBP(YEND+1,I),AUNNBP(YEND+1,I),AUNROD(YEND+1,I)
  S1=S1+AUNREG(YEND+1,I)
  S2=S2+AUNC1S(YEND+1,I)
  W1=W1+AUNGBP(YEND+1,I)
  W2=W2+AUNNBP(YEND+1,I)
105 W3=W3+AUNROD(YEND+1,I)
  W1=W1/FLOAT(MS)
  W2=W2/FLOAT(MS)
  W3=W3/FLOAT(MS)
  WRITE (NOUT1,227) S1,S2,W1,W2,W3,UNUT
106 IF (YEND.NE.NY) GO TO 34
```

```
*****
RESET DATA FOR VALIDATION PROCESS
*****
```

```
IF (NY.NE.1) GO TO 108
IF (NULD.EQ.1) GO TO 108
REWIND NOUT2
```

```
DO 107 J=1,5
107 READ (NOUT2,END=999) (AA(I,J+5),I=1,NSA1)
CALL SCLE (NSA1,10,AA,10.0)
REWIND NOUT2
WRITE (NOUT2,242) DNUM
CALL OUTX (NSA1,AA,IR,NOR,NT,NOUT2,0)
108 CALL SCLE (NT9,NT9,AA,0.0)
IF (NY.NE.1) WRITE (NOUT2,242) DNUM

*****
READ DATA FOR COMPARATIVE SUMMARY
*****

READ (NIN2,END=999) (((AA(I1,J),I1=1,25), (AA(I2,J+5),I2=1,25), (AA(
113,J+10),I3=1,25), (AA(I4,J+15),I4=1,25), (AA(I5,J+20),I5=1,25), (AA(
216,J+25),I6=1,25)),J=1,5)
DO 109 I=1,4
DO 109 J=1,30
109 AA(I,J)=AA(I,J)*SCALE1
DO 113 I=6,14
IF (I.EQ.10) GO TO 111
IF (I.GT.12) GO TO 111
DO 110 J=1,30
110 AA(I,J)=AA(I,J)*SCALE2
GO TO 113
111 DO 112 J=1,30
112 AA(I,J)=AA(I,J)*SCALE3
113 CONTINUE
DO 114 I=19,25
DO 114 J=1,30
114 AA(I,J)=AA(I,J)*SCALE3
IF (IPLT.NE.1) GO TO 116
DO 115 I=KN1,KN2
CALL OUTY (NSA1,AA,NOUT2,JPLT(I))
115 CONTINUE
116 DO 117 I=1,5
IPAG=IPAG+1
IF (I.LT.5) WRITE (NOUT1,243) DNUM,IPAG,I,(NY2(KN),KN=1,6)
IF (I.EQ.5) WRITE (NOUT1,244) DNUM,IPAG,(NY2(KN),KN=1,6)
117 WRITE (NOUT1,245) (AA(K,I),AA(K,I+5),AA(K,I+10),AA(K,I+15),AA(K,I+
120),AA(K,I+25),K=1,25)

*****
READ DATA FOR VARIABLES IN COMPARATIVE REPORTS
*****

IF (ISTATE.GT.0) ISTATE=2
IF (ISTATE.LE.0) ISTATE=1
IF (NIC.EQ.0) GO TO 148
IF (KN3.EQ.0.OR.KN4.EQ.0) IPLT=0
KJ=0
DO 146 I,K=1,NIC
N1=KJ+1
DO 118 M=N1,10
IF (ICR(M).EQ.0.AND.ICRS(M).EQ.0) GO TO 118
KJ=M
GO TO 119
118 CONTINUE
119 READ (NIN2,END=999) (((AA(I1,J),I1=1,NT9), (AA(I2,J+5),I2=1,NT9)
1 (AA(I3,J+10),I3=1,NT9), (AA(I4,J+15),I4=1,NT9), (AA(I5,J+20),I5=
```

```
2 1,NT9),(AA(16,J*25),I6=1,NT9)),J=1,5)
GO TO (120,124,128,130,132,134,136,140,142,144), KJ

*****
PRODUCER RECEIPTS
*****

120 IF (NY.EQ.1) CALL OUTX (NSA1,AA,IR,NOR,NT,NOUT2,1)
CALL SCLE (NT9,30,AA,SCALE1)
IF (IPLT.NE.1) GO TO 122
DO 121 I=KN3,KN4
KPLT=IR(JPLT(I))
CALL OUTY (NSA1,AA,NOUT2,KPLT)

121 CONTINUE
122 DO 123 K=1,ISTATE
IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 123
IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 123
NSA2=1+(K-1)*NSA1
NSA3=NSA1+(K-1)*NSA1S
DO 123 I=1,5
IPAG=IPAG+1
IF (I.LT.5) WRITE (NOUT1,246) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
1 IF (I.EQ.5) WRITE (NOUT1,247) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
2(KN),KN=1,6)
CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)

123 CONTINUE
GO TO 146

*****
CLASS I SALES
*****

124 IF (NY.EQ.1) CALL OUTX (NSA1,AA,IR,NOR,NT,NOUT2,1)
CALL SCLE (NT9,30,AA,SCALE1)
IF (IPLT.NE.1) GO TO 126
DO 125 I=KN3,KN4
KPLT=IR(JPLT(I))
CALL OUTY (NSA1,AA,NOUT2,KPLT)

125 CONTINUE
126 DO 127 K=1,ISTATE
IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 127
IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 127
NSA2=1+(K-1)*NSA1
NSA3=NSA1+(K-1)*NSA1S
DO 127 I=1,5
IPAG=IPAG+1
IF (I.LT.5) WRITE (NOUT1,248) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
1 IF (I.EQ.5) WRITE (NOUT1,249) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
2(KN),KN=1,6)
CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)

127 CONTINUE
GO TO 146

*****
CLASS II SALES
*****

128 IF (NY.EQ.1) CALL OUTX (NSA1,AA,IR,NOR,NT,NOUT2,1)
```

```
CALL SCLE (NT9,30,AA,SCALE1)
DO 129 K=1,ISTATE
  IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 129
  IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 129
  NSA2=1+(K-1)*NSA1
  NSA3=NSA1+(K-1)*NSA1S
DO 129 I=1,5
  IPAG=IPAG+1
  IF (I.LT.5) WRITE (NOUT1,252) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
  IF (I.EQ.5) WRITE (NOUT1,253) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
  CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)
129 CONTINUE
GO TO 146
```

```
*****
CLASS III SALES
*****
```

```
130 IF (NY.EQ.1) CALL OUTX (NSA1,AA,IR,NOR,NT,NOUT2,1)
CALL SCLE (NT9,30,AA,SCALE1)
DO 131 K=1,ISTATE
  IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 131
  IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 131
  NSA2=1+(K-1)*NSA1
  NSA3=NSA1+(K-1)*NSA1S
DO 131 I=1,5
  IPAG=IPAG+1
  IF (I.LT.5) WRITE (NOUT1,254) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
  IF (I.EQ.5) WRITE (NOUT1,255) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
  CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)
131 CONTINUE
GO TO 146
```

```
*****
CLASS I UTILIZATION
*****
```

```
132 DO 133 K=1,ISTATE
  IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 133
  IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 133
  NSA2=1+(K-1)*NSA1
  NSA3=NSA1+(K-1)*NSA1S
DO 133 I=1,5
  IPAG=IPAG+1
  IF (I.LT.5) WRITE (NOUT1,256) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
  IF (I.EQ.5) WRITE (NOUT1,257) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
  CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)
133 CONTINUE
IF (NY.NE.1) GO TO 146
CALL SCLE (NT9,10,AA,10.0)
CALL OUTX (NSA1,AA,IR,NOR,NT,NOUT2,1)
GO TO 146
```

```
*****
```

CLASS I PRICE

```
134 CALL SCLE (NT9,30,AA,SCALE2)
DO 135 K=1, ISTATE
  IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 135
  IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 135
  NSA2=1+(K-1)*NSA1
  NSA3=NSA1+(K-1)*NSA1S
DO 135 I=1,5
  IPAG=IPAG+1
  IF (I.LT.5) WRITE (NOUT1,258) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
  IF (I.EQ.5) WRITE (NOUT1,259) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
  CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,1,NOUT1)
135 CONTINUE
GO TO 146
```

BLEND PRICE

```
136 CALL SCLE (NT9,30,AA,SCALE2)
IF (IPLT.NE.1) GO TO 138
DO 137 I=KN3,KN4
  KPLT=IR(JPLT(I))
  CALL OUTY (NSA1,AA,NOUT2,KPLT)
137 CONTINUE
138 DO 139 K=1, ISTATE
  IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 139
  IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 139
  NSA2=1+(K-1)*NSA1
  NSA3=NSA1+(K-1)*NSA1S
DO 139 I=1,5
  IPAG=IPAG+1
  IF (I.LT.5) WRITE (NOUT1,260) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
  IF (I.EQ.5) WRITE (NOUT1,261) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
  CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,1,NOUT1)
139 CONTINUE
GO TO 146
```

PROCESSING CAPACITY UTILIZATION

```
140 DO 141 K=1, ISTATE
  IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 141
  IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 141
  NSA2=1+(K-1)*NSA1
  NSA3=NSA1+(K-1)*NSA1S
DO 141 I=1,5
  IPAG=IPAG+1
  IF (I.LT.5) WRITE (NOUT1,262) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
  IF (I.EQ.5) WRITE (NOUT1,263) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
  CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)
```

141 CONTINUE
GO TO 146

TOTAL PROCESSING COST

142 CALL SCLE (NT9,30,AA,SCALE2)
DO 143 K=1,ISTATE
IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 142
IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 142
NSA2=1+(K-1)*NSA1
NSA3=NSA1+(K-1)*NSA1S
DO 143 I=1,5
IPAG=IPAG+1
IF (I.LT.5) WRITE (NOUT1,264) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
IF (I.EQ.5) WRITE (NOUT1,265) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,1,NOUT1)
143 CONTINUE
GO TO 146

IN AREA CLASS I SALES

144 CALL SCLE (NT9,30,AA,SCALE1)
DO 145 K=1,ISTATE
IF (K.EQ.1.AND.ICR(KJ).LE.0) GO TO 145
IF (K.GT.1.AND.ICRS(KJ).LE.0) GO TO 145
NSA2=1+(K-1)*NSA1
NSA3=NSA1+(K-1)*NSA1S
DO 145 I=1,5
IPAG=IPAG+1
IF (I.LT.5) WRITE (NOUT1,250) DNUM,IPAG,(CMP(J,K),J=1,2),I,(
1 NY2(KN),KN=1,6)
IF (I.EQ.5) WRITE (NOUT1,251) DNUM,IPAG,(CMP(J,K),J=1,2),(NY
1 2(KN),KN=1,6)
CALL RITE (I,NSA2,NSA3,K,IR,NOR,ONAME,AA,0,NOUT1)
145 CONTINUE
146 CONTINUE
GO TO 148
999 WRITE (NOUT1,266)
148 CONTINUE

STOP

FORMATS

149 FORMAT (14I3)
150 FORMAT (1H1,20X,2SHINPUT FORM FOR DAMPS - MODEL ,A2,12X,5HPAGE ,I3
1,1H./21X,31(1H-)/25X,15MDECISION NAME -,1X,A8/25X,22(1H=)//)
151 FORMAT (2X,32HTHE NUMBER OF PROPOSED MERGERS =,I3,1H.)
152 FORMAT (/2X,6HMERGER, I3,19H - NUMBER OF ORDERS,12H IN MERGER =,I3,
11H./7X,13HORDER NUMBERS/7X,13(1H-)/(7X,15I4))
153 FORMAT (/23X,25HBLEND PRICE DIFFERENTIALS/23X,25(1H-)/2X,34HOPTION
1 1 - LOCATION DIFFERENTIAL =,F8.4//2X,5HORDER,19X,5HBLEND/3X,3HND.
2,8X,4HNAME,5X,12HDIFFERENTIAL/3X,3H---,5X,10(1H-),2X,12(1H-))
154 FORMAT (/23X,25HBLEND PRICE DIFFERENTIALS/23X,25(1H-)/2X,40HOPTION

- 1 2 - USER SPECIFIED DIFFERENTIALS.//2X,5HORDER,19X,5HBLEND/3X,3HNO
2.,2X,4HNAME,5X,12HDIFFERENTIAL/3X,3H---,5X,10(1H-),2X,12(1H-))
- 155 FORMAT (3X,I3,5X,A10,F11.4)
156 FORMAT (2X,45H3. CHANGE CLASS I PRICE LEVEL AND STRUCTURE./42X,5H
1PRICE/33X,23(1H-)/9X,5HORDER,19X,4HLAST,3X,5HPROP.,4X,6HQUART./10X
2,3HNO.,3X,4HNAME,6X,2(2X,5HPRICE),2X,5HABS.CHGE./10X,3(1H-),5X,10(
31H-),5X,5(1H-),2X,5(1H-),2X,9(1H-))
- 157 FORMAT (10X,I3,5X,A10,4X,F6.2,F7.2,F9.2)
158 FORMAT (2X,47H3-S. CHANGE CLASS I PRICE LEVEL AND STRUCTURE-.7HST
1ATES./42X,5HPRICE/33X,23(1H-)/9X,5HSTATE,19X,4HLAST,3X,5HPROP.,4X,
26HQUART./10X,3HNO.,8X,4HNAME,6X,2(2X,5HPRICE),2X,5HABS.CHGE./10X,3
3(1H-),5X,10(1H-),5X,5(1H-),2X,5(1H-),2X,9(1H-))
- 159 FORMAT (//4X,57HNOTE... LAST PRICE IS MINIMUM FEDERAL ORDER PRICE
1FOR THE/12X,13H4TH QUARTER, ,I4,1H./1H1,19X,30HINPUT FORM FOR DAM
2PS - MODEL ,A2,11X,5HPAGE ,I3,1H./20X,32(1H-)/25X,15HDECISION NAME
3 -,1X,AG/25X,22(1H-)//2X,33H4. CHANGE CLASS II PRICE LEVEL -,15H
4LAST PRICE = \$,F5.2/6X,14HPROPOSED PRICE,F10.2,2X,21HQUARTERLY ABS
5. CHANGE,F10.2//2X,49H5. CHANGE CLASS III PRICE LEVEL - LAST PRIC
6E = \$,F5.2/6X,14HPROPOSED PRICE,F10.2,2X,21HQUARTERLY ABS. CHANGE,
7F10.2/)
- 160 FORMAT (2X,20H6. REPORTS DESIRED.//6X,8HA. YEARS,1X,5(6X,I4)/15X,
15(6X,4H---)/14X,5(8X,I2)/)
- 161 FORMAT (6X,11HB. QUARTERS/11X,3H1ST,5(8X,I2)/11X,3H2ND,5(8X,I2)/11
1X,3H3RD,5(8X,I2)/11X,3H4TH,5(8X,I2)/)
- 162 FORMAT (2X,30H7. LAST YEAR TO BE ANALYZED -,7X,I4//2X,36H8. PRIN
1T REPORT SHOWING BASE DATA -,15//2X,34H9. OPTION FOR EXOGENOUS FA
2CTORS -,17//4X,57HNOTE... LAST PRICE IS MINIMUM FEDERAL ORDER PRI
3CE FOR THE/12X,13H4TH QUARTER, ,I4,1H.)
- 163 FORMAT (1X,49H10. DATA TO BE READ FROM SUPPLEMENTAL INPUT FORM,5H
1 F-A.//9X,45HA. READ PERCENTAGES FOR QUARTERLY CHANGES IN/13X,24H
2CLASS I, II, III PRICES.,22X,I4//9X,29HB. READ DEMAND ELASTICITIE
3S.,21X,I4//9X,29HC. READ SUPPLY ELASTICITIES.,21X,I4//9X,40HD. R
4EAD HANDLING CHARGES ON INTRA-ORDER/13X,23HSUPPLY PLANT SHIPMENTS.
5,23X,I4//9X,40HE. READ HANDLING CHARGES ON INTER-ORDER/13X,23HSUP
6PLY PLANT SHIPMENTS.,23X,I4//9X,30HF. READ RESERVE REQUIREMENTS.,
720X,I4//)
- 164 FORMAT (1X,54H11. DATA TO BE READ FROM SUPPLEMENTAL INPUT FORM S-
1A.//9X,45HA. READ PERCENTAGES FOR QUARTERLY CHANGES IN/13X,24HCLA
2SS I, II, III PRICES.,22X,I4//9X,29HB. READ DEMAND ELASTICITIES.,
321X,I4//9X,29HC. READ SUPPLY ELASTICITIES.,21X,I4//9X,30HD. READ
4 RESERVE REQUIREMENTS.,20X,I4//)
- 165 FORMAT (1X,45H12. SUPPLEMENTAL PRICE INPUT, FEDERAL ORDER.//9X,42
1HA. READ QUARTERLY CLASS I, II, III PRICES/13X,14HFROM FORM F-B.,
232X,I4//9X,30HB. READ A CLASS I BASE PRICE,18H QUARTERLY CHANGE,
3/13X,35HAND LOCATION DIFFERENTIAL FROM FORM,5H F-C.,6X,I4//9X,42HC
4. READ QUARTERLY CLASS I BASE PRICES AND/13X,35HCLASS II, III PRI
5CES FROM FORM F-D.,11X,I4//)
- 166 FORMAT (1X,43H13. SUPPLEMENTAL PRICE INPUT, STATE ORDER.//9X,43HA
1. READ QUARTERLY CLASS I, II, III, PRICES/13X,9HFROM FORM,5H S-B.
2,32X,I4//9X,41HB. READ A CLASS I BASE PRICE, QUARTERLY ,7HCHANGE,
3/13X,40HAND LOCATION DIFFERENTIAL FROM FORM S-C.,6X,I4//9X,42HC.
4READ QUARTERLY CLASS I BASE PRICES AND/13X,38HCLASS II AND III PRI
5CES FROM FORM S-D.,8X,I4//)
- 167 FORMAT (1X,39H14. CHANGE CLASS II PRICE ELASTICITY -,18H BASE ELA
1STICITY =,F6.2/6X,19HPROPOSED ELASTICITY,28X,F10.3//1X,53H15. REMO
2VE CLASS I PROCESSING CAPACITY RESTRICTION -,19//1X,52H16. REMOU
3E MANUFACTURED MILK CAPACITY RESTRICTION -,110//1X,44H17. REMOUE
4PACKAGED MILK FLOW RESTRICTION -,13X,I5//1X,39H18. REMOUE RAW MIL
5K FLOW RESTRICTION -,18X,I5//1X,36H19. OPTION FOR CLASS I SALES B
6ASE -,21X,I5//1X,33H20. COMPARATIVE REPORTS DESIRED.,22H (MAXIMU

7M OF FIVE (5)/6X,27HCAN BE REQUESTED FOR A RUN//9X,22HA. PRODUCER
8 RECEIPTS -,110/9X,18HB. CLASS I SALES -,114/9X,19HC. CLASS II SAL
SES -,113/9X,20HD. CLASS III SALES -,112/9X,24HE. CLASS I UTILIZATI
*GN -,18/9X,18HF. CLASS I PRICE -,114/9X,16HG. BLEND PRICE -,116/9X
*,24HH. PROCESSING CAP. USE -,18/9X,26HI. TOTAL PROCESSING COST -,I
*6/9X,26HJ. IN AREA CLASS I SALES -,16/

168 FORMAT (1X,40H21. COMPARATIVE REPORTS DESIRED-STATES.,22H (MAXIM
UM OF FIVE (5)/6X,27HCAN BE REQUESTED FOR A RUN//9X,22HA. PRODUCE
2R RECEIPTS -,110/9X,18HB. CLASS I SALES -,114/9X,19HC. CLASS II SA
LES -,113/9X,20HD. CLASS III SALES -,112/9X,24HE. CLASS I UTILIZAT
4ION -,18/9X,18HF. CLASS I PRICE -,114/9X,16HG. BLEND PRICE -,116/9
5X,24HH. PROCESSING CAP. USE -,18/9X,26HI. TOTAL PROCESSING COST -,
6/9X,26HJ. IN AREA CLASS I SALES -,16)

169 FORMAT (1X,41H22. DESIRED GOVERNMENT STOCKS OF CHEESE.//26X,1HI,1
10X,2HII,9X,3HIII,10X,2HIU/13X,4H1977,4F12.0/13X,4H1978,4F12.0/,13X
2,4H1979,4F12.0/13X,4H1980,4F12.0/13X,4H1981,4F12.0//)

170 FORMAT (1X,40H23. GOVERNMENT STOCKS RELEASED OPTION -,18X,14//1X,
127H24. IMPORT CHANGE OPTION -,31X,14//1X,35H25. MANUFACTURING RE
2PORTS DESIRED.//9X,18HA. YEARLY REPORTS//13X,4HYEAR,8X,31H 1977
3 1978 1979 1980 1981/28X,5(4(1H-),2X)//25X,5I6//9X,23HB. COMP
4ARATIVE REPORTS//13X,19HA. GRADE B PROD. -,11X,14/13X,21HB. TOTA
5L MFG PROD. -,113/13X,20HC. CLASS II CONS. -,10X,14/13X,18HD. CH
6ESEE CONS. -,12X,14/13X,18HE. BUTTER CONS. -,116/13X,27HF. NONFA
7T DRY MILK CONS. -,17/13X,27HG. MISC. CLASS III CONS. -,17/13X,20
8HM. END COMM CHEESE ,6HSTK. -,18/13X,26HI. END COMM BUTTER STK.
9-,18/13X,24HJ. END COMM NFDM STK. -,110/13X,26HK. END GOVT CHEES
*E STK. -,18/13X,26HL. END GOVT BUTTER STK. -,18/13X,13HM. END GO
*UT ,11HNFDM STK. -,110/13X,2HN.,28X,14/)

171 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM F-A/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,AG/25X,22(1H=)//19X,7HCLASS I,20X,5HHAND.,3X,5HHAND./20
3X,5HPRICE,20X,6HCHARGE,2X,6HCHARGE/2X,5HORDER,11X,25HQUARTERLY DE
4MAND SUPPLY,4X,3HOWN,4X,14HOTHER RESERVE/3X,3HNO.,4X,4HNAME,4X,2
55HPCT.CHGE. ELAST. ELAST.,2X,14HMARKET MARKET,4X,4HREQ./3X,3(1H
6-),4X,4(1H-),4X,9(1H-),4(2X,6H-----),2X,7(1H-))

172 FORMAT (3X,I3,1X,A10,F8.2,F9.3,3F8.3,F9.1)

173 FORMAT (1H0,2X,39HCLASS II PRICE QUARTERLY PERCENT CHANGE,F8.2/3X,
140HCLASS III PRICE QUARTERLY PERCENT CHANGE,F7.2)

174 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM F-B/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,AG/22X,25(1H=)//29X,14HCLASS I PRICES/29X,14(1H-)/30X,6
3HYEAR -,1X,14/30X,11(1H-)/37X,7HQUARTER/1X,5HORDER,17X,35(1H-)/2X,
43HNO.,6X,4HNAME,9X,3H(1),7X,3H(2),7X,3H(3),7X,3H(4)/2X,3(1H-),3X,1
50(1H-),4(5X,5H-----))

175 FORMAT (2X,I3,3X,A10,4F10.2)

176 FORMAT (1H1,15X,37HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5HP
1AGE ,I3,1H./16X,40(1H-)/32X,8HFORM F-B/32X,8(1H=)/25X,15HDECISION
2NAME -,1X,AG/22X,25(1H=)//

177 FORMAT (31X,19H- CLASS II PRICES -/31X,19(1H-)/37X,7HQUARTER/23X,3
15(1H-)/9X,4HYEAR,11X,3H(1),7X,3H(2),7X,3H(3),7X,3H(4)/9X,4(1H-),5X
2,4(5X,5H-----)/9X,14,5X,4F10.2/9X,14,5X,4F10.2/9X,14,5X,4F10.2/9X,
314,5X,4F10.2/9X,14,5X,4F10.2)

178 FORMAT (//30X,20H- CLASS III PRICES -/30X,20(1H-)/37X,7HQUARTER/23
1X,35(1H-)/9X,4HYEAR,11X,3H(1),7X,3H(2),7X,3H(3),7X,3H(4)/9X,4(1H-)
2,5X,4(5X,5H-----)/9X,14,5X,4F10.2/9X,14,5X,4F10.2/9X,14,5X,4F10.2/
39X,14,5X,4F10.2/9X,14,5X,4F10.2)

- 179 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM F-C/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,A6/25X,22(1H=)//1X,28H17. LOCATION DIFFERENTIAL -,10X,
3F10.4//1X,37H18. QUARTERLY CHANGE IN BASE PRICE -,1X,F10.2//1X,25
4H19. CLASS I BASE PRICE -,13X,F10.2)
- 180 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM F-D/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,A6/25X,22(1H=)//1X,28H17. LOCATION DIFFERENTIAL -,10X,
3F10.4//1X,29H18. CLASS I, II, III PRICES./29X,23H- CLASS I BASE
4PRICES -/29X,23(1H-)/37X,7HQUARTER/23X,35(1H-)/9X,4HYEAR,11X,3H(1)
5,7X,3H(2),7X,3H(3),7X,3H(4)/9X,4(1H-),5X,4(5X,5H-----)/9X,14,5X,4F
610.2/9X,14,5X,4F10.2/9X,14,5X,4F10.2/9X,14,5X,4F10.2/9X,14,5X,4F10
7.2//)
- 181 FORMAT (1H1,13X,43HGENERATED PRICE SURFACE FOR DAMPS - MODEL ,A2,
15X,5HPAGE ,I3,1H./14X,45(1H-)/25X,15HDECISION NAME -,1X,A6/22X,25(
21H=)//29X,14HCLASS I PRICES/29X,14(1H-)/30X,6HYEAR -,1X,I4/30X,11(
31H-)/37X,7HQUARTER/1X,5HORDER,17X,35(1H-)/2X,3HNO.,6X,4HNAME,9X,3H
4(1),7X,3H(2),7X,3H(3),7X,3H(4)/2X,3(1H-),3X,10(1H-),4(5X,5H-----))
- 182 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM S-A/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,A6/25X,22(1H=)//19X,7HCLASS I/20X,5HPRICE/2X,5HSTATE,11
3X,25HQUARTERLY DEMAND SUPPLY,2X,7HRESERVE/3X,3HNO.,4X,4HNAME,4X,
425HPCT.CHGE. ELAST. ELAST.,4X,4HREQ./3X,3(1H-),4X,4(1H-),4X,9(1H
5-),2(2X,6H-----),2X,7(1H-))
- 183 FORMAT (3X,I3,1X,A10,F8.2,F9.3,F8.3,F9.1)
- 184 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM S-B/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,A6/22X,25(1H=)//29X,14HCLASS I PRICES/29X,14(1H-)/30X,6
3HYEAR -,1X,I4/30X,11(1H-)/37X,7HQUARTER/1X,5HORDER,17X,35(1H-)/2X,
43HNO.,6X,4HNAME,9X,3H(1),7X,3H(2),7X,3H(3),7X,3H(4)/2X,3(1H-),3X,1
50(1H-),4(5X,5H-----))
- 185 FORMAT (2X,I3,3X,A10,4F10.2)
- 186 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM S-C/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,A6/25X,22(1H=)//1X,28H17. LOCATION DIFFERENTIAL -,10X,
3F10.4//1X,37H18. QUARTERLY CHANGE IN BASE PRICE -,1X,F10.2//1X,25
4H19. CLASS I BASE PRICE -,13X,F10.2)
- 187 FORMAT (1H1,15X,38HSUPPLEMENTAL INPUT FOR DAMPS - MODEL ,A2,7X,5H
1PAGE ,I3,1H./16X,40(1H-)/32X,8HFORM S-D/32X,8(1H=)/25X,15HDECISION
2 NAME -,1X,A6/25X,22(1H=)//1X,28H17. LOCATION DIFFERENTIAL -,10X,
3F10.4//1X,20H18. CLASS I PRICES./29X,23H- CLASS I BASE PRICES -/
429X,23(1H-)/37X,7HQUARTER/23X,35(1H-)/9X,4HYEAR,11X,3H(1),7X,3H(2)
5,7X,3H(3),7X,3H(4)/9X,4(1H-),5X,4(5X,5H-----)/9X,14,5X,4F10.2/9X,I
64,5X,4F10.2/9X,14,5X,4F10.2/9X,14,5X,4F10.2/9X,14,5X,4F10.2//)
- 188 FORMAT (1H1,13X,43HGENERATED PRICE SURFACE FOR DAMPS - MODEL ,A2,
15X,5HPAGE ,I3,1H./14X,45(1H-)/25X,15HDECISION NAME -,1X,A6/22X,25(
21H=)//29X,14HCLASS I PRICES/29X,14(1H-)/30X,6HYEAR -,1X,I4/30X,11(
31H-)/37X,7HQUARTER/1X,5HORDER,17X,35(1H-)/2X,3HNO.,6X,4HNAME,9X,3H
4(1),7X,3H(2),7X,3H(3),7X,3H(4)/2X,3(1H-),3X,10(1H-),4(5X,5H-----))
- 189 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//23X,25HSUMMARY FOR ORDER MERGERS/23X,25(1H-)/27X,7HQUARTER,
2I2,1X,4HYEAR,15/27X,19(1H-)/65X,5HGROSS/34X,8HPRODUCER,3X,7HCLASS
3I,3X,7HCLASS I,3X,5HBLEND/2X,6HMERGER,26X,8HRECEIPTS,4X,5HSALES,6X
4,3HUSE,5X,5HPRICE/2X,6HNUMBER,26X,8HMIL.LBS.,3X,8HMIL.LBS.,4X,4MPC
5T.,4X,5H\$/CWT/3X,4H-----,27X,8(1H-),3X,8(1H-),2X,7(1H-),3X,5(1H-))
- 190 FORMAT (3X,I3,7X,17HORDERS IN MERGER ,F12.1,F10.1,F9.1,F9.2)
- 191 FORMAT (9X,24(1H-)/(9X,6I4))
- 192 FORMAT (9X,24(1H-)/)
- 193 FORMAT (1H1,23X,15HDECISION NAME -,1X,A6,17X,5HPAGE ,I3,1H./24X,22
1(1H=)//28X,14HSUMMARY REPORT/28X,14(1H-)/25X,7HQUARTER,I2,1X,4HYEA
2R,I5/25X,19(1H-)/45X,4HTHIS,3X,20HCHANGE FROM YEAR AGO/44X,16HPERI
30D ABSOLUTE,2X,10HPERCENTAGE/44X,6(1H-),2X,8(1H-),2X,10(1H-))

- 194 FORMAT (1X,31HPRODUCER RECEIPTS (MIL. POUNDS),8X,3F10.1//1X,27HCLA
195 I SALES (MIL. POUNDS),12X,3F10.1//1X,28HCLASS II SALES (MIL. PO
UNDS),11X,3F10.1//1X,29HCLASS III SALES (MIL. POUNDS),10X,3F10.1//
31X,26HCLASS I UTILIZATION (PCT.),13X,3F10.1//1X,28HCLASS I PRICE (4\$ - WT. AVE.),11X,2F10.2,F10.1//1X,29HCLASS II PRICE (\$ - WT. AVE.
5),10X,2F10.2,F10.1//1X,30HCLASS III PRICE (\$ - WT. AVE.),9X,2F10.2
6,F10.1//1X,32HGROSS BLEND PRICE (\$ - WT. AVE.),7X,2F10.2,F10.1//1X
7,24HVALUE OF POOL (THOUS. \$),15X,2F10.0,F10.1//1X,29HNET FARM PRIC
8E (\$ - WT. AVE.),10X,2F10.2,F10.1//1X,36HRETURN OVER DIRECT COST (9\$-WT. AVE.),3X,2F10.2,F10.1//1X,27HMARKETING MARGIN (THOUS. \$),12X
*,2F10.0,F10.1//1X,31HCONSUMER EXPENDITURE (THOUS. \$),8X,2F10.0,F10
*.1//1X,39HRETAIL PRICE (WT. AVE. - CTS. 1/2 GAL.),3F10.1//1X,39HFA
*RM-RETAIL PRICE SPREAD (WT.AVE.-CTS.),3F10.1//1X,38HPROCESSING CAP
*ACITY UTILIZATION (PCT.),1X,3F10.1//1X,32HMFG. CAPACITY UTILIZATIO
*N (PCT.),7X,3F10.1//1X,35HRAW MILK TRANSPORTATION COST (000\$),4X,2
*F10.0,F10.1//1X,39HPACKAGED MILK TRANSPORTATION COST(000\$),2F10.0,
*F10.1)
- 195 FORMAT (1H1,23X,15HDECISION NAME -,1X,A6,17X,5HPAGE ,I3,1H./24X,22
1(1H=)//25X,20HFEDERAL ORDER REPORT/25X,20(1H-)/26X,7HQARTER,I2,1X
2,4HYEAR,I5/26X,19(1H-)/28X,4HPCT.,18X,19HCLASS CLASS GROSS/18X,3
31HPRODUCER CHANGE CLASS I CLASS I,3X,2HII,4X,3HIII,3X,5HBLEND/1X,5
4HORDER,12X,14HRECEIPTS FROM,4X,3HUSE,2X,4(2X,5HPRICE)/2X,3HNO.,5X
5,4HNAME,4X,15HMIL.LBS. YR.AGO,8X,4HPCT.,2X,4(1X,6H\$/CWT.)/2X,3(1H-
6),2X,10(1H-),1X,8(1H-),1X,6(1H-),1X,7(1H-),1X,7(1H-),1X,6(1H-),1X,
76(1H-),1X,6(1H-))
- 196 FORMAT (2X,I3,2X,A10,F9.1,F6.1,F8.1,F8.2,3F7.2)
- 197 FORMAT (18X,8(1H-)/4X,5HTOTAL,7X,F10.1,1X,5(1H-),3X,5(1H-)
1-),2X,5(1H-),2X,5(1H-),2X,5(1H-)/1X,16HWEIGHTED AVERAGE,8X,F7.1,F8
2.1,F8.2,3F7.2)
- 198 FORMAT (1H1,23X,15HDECISION NAME -,1X,A6,17X,5HPAGE ,I3,1H./24X,22
1(1H=)//21X,28HFEDERAL ORDER REPORT - SALES/21X,28(1H-)/25X,7HQART
2ER,I2,1X,4HYEAR,I5/25X,19(1H-)/28X,4HPCT.,14X,4HPCT.,15X,4HPCT./18
3X,33HCLASS I CHANGE CLASS II CHANGE,2X,17HCLASS III CHANGE/1X,
45HORDER,13X,5HSALES,4X,4HFROM,4X,5HSALES,5X,4HFROM,5X,5HSALES,5X,4
5HFROM/2X,3HNO.,5X,4HNAME,4X,33HMIL.LBS. YR.AGO MIL.LBS. YR.AGO,3
6X,16HMIL.LBS. YR.AGO/2X,3(1H-),2X,10(1H-),1X,8(1H-),1X,6(1H-),2X,
78(1H-),2X,6(1H-),2X,9(1H-),2X,6(1H-))
- 199 FORMAT (2X,I3,2X,A10,F8.1,F7.1,F10.1,F8.1,F11.1,F8.1)
- 200 FORMAT (18X,7(1H-),10X,7(1H-),12X,7(1H-)/4X,5HTOTAL,6X,F10.1,2X,5(1
1H-),F10.1,3X,5(1H-),F11.1,4X,5(1H-)/1X,16HWEIGHTED AVERAGE,7X,F8.
21,10X,F8.1,12X,F8.1/1X,38HCLASS II SALES FROM NONORDER SOURCES :,F
311.1)
- 201 FORMAT (1H1,18X,15HDECISION NAME -,1X,A6,22X,5HPAGE ,I3,1H./19X,22
1(1H=)//19X,22HMILK PRODUCTION REPORT/19X,22(1H-)/21X,7HQARTER,I2,
21X,4HYEAR,I5/21X,19(1H-)/57X,6HRETURN/2X,6HSUPPLY,26X,4HPCT.,3X,5H
3GROSS,4X,3HNET,5X,4HOVER/3X,4HAREA,14X,25HPRODUCTION CHANGE BLEN
4D,4X,4HFARM,3X,6HDIRECT/4X,3HNO.,5X,4HNAME,6X,24HMIL. LBS. YR.AGO
5 PRICE,3X,5HPRICE,4X,4HCOST/4X,3(1H-),2X,10(1H-),2X,10(1H-),2X,6(1
61H-),2X,5(1H-),3X,5(1H-),3X,6(1H-))
- 202 FORMAT (4X,I3,2X,A10,F11.1,F8.1,3F8.2)
- 203 FORMAT (21X,9(1H-)/4X,5HTOTAL,10X,F11.1,4(3X,5H-----)/1X,16HWEIGHT
1ED AVERAGE,13X,F8.1,3F8.2)
- 204 FORMAT (1H1,23X,15HDECISION NAME -,1X,A6,17X,5HPAGE ,I3,1H./24X,22
1(1H=)//23X,24HPROCESSING CENTER REPORT/23X,24(1H-)/26X,7HQARTER, I
22,1X,4HYEAR,I5/26X,19(1H-)/41X,3HRAW,5X,3HRAW,3X,8HPACKAGED/1X,10H
3PROCESSING,9X,8HPACKAGED,2X,8HCAPACITY,3X,4HMILK,4X,4HMILK,4X,4HMI
4LK,4X,5HTOTAL/2X,6HCENTER,14X,4HMILK,6X,3HUSE,4X,7HACQUIS.,2X,6HTR
5ANS.,2X,6HTRANS.,4X,4HCOST/4X,3HNO.,5X,4HNAME,6X,5HSALES,5X,4HPCT.
6,5X,4HCOST,4X,4HCOST,4X,4HCOST,4X,8HPER CWT./4X,3(1H-),2X,10(1H-),
72X,6(1H-),4X,5(1H-),4X,6(1H-),2X,6(1H-),2X,6(1H-),3X,6(1H-))

- 205 FORMAT (4X, I3, 2X, A10, F8.1, F9.1, F9.2, 2F8.4, F10.2)
- 206 FORMAT (31X, 5(1H-), 4X, 5(1H-), 3X, 5(1H-), 3X, 5(1H-), 5X, 5(1H-)/1X, 16HW
EIGHTED AVERAGE, 3X, 7(1H-), F9.1, F9.2, F8.4, F8.4, F10.2/4X, SHTOTAL, 8X,
2F10.1)
- 207 FORMAT (1H1, 18X, 15HDECISION NAME -, 1X, A6, 22X, 5HPAGE , I3, 1H./19X, 22
1(1H=)//16X, 27HMANUFACTURING CENTER REPORT/16X, 27(1H-)/20X, 7HQUARTE
2R, I2, 1X, 4HYEAR, I5/20X, 19(1H-)/23X, 12HMANUFACTURED, 5X, 8HCAPACITY/1X
3, 6HCENTER, 20X, 4HMILK, 8X, 11HUTILIZATION/3X, 2HID, 8X, 4HNAME, 7X, 10H(MI
4L.LBS.), 6X, 9H(PERCENT)/3X, 3(1H-), 4X, 10(1H-), 5X, 8(1H-), 8X, 7(1H-))
- 208 FORMAT (3X, A3, 4X, A10, 2X, F10.1, 5X, F10.1)
- 209 FORMAT (23X, 10(1H-)/6X, SHTOTAL, 9X, F12.1, 9X, 7(1H-)/9X, 7HAVERAGE, 21X
1, F10.1)
- 210 FORMAT (1H1, 22X, 15HDECISION NAME -, 1X, A6, 18X, 5HPAGE , I3, 1H./23X, 22
1(1H=)//25X, 18HCONSUMPTION REPORT/25X, 18(1H-)/25X, 7HQUARTER, I2, 1X, 4
2HYEAR, I5/25X, 19(1H-)/64X, 4HFARM/1X, 11HCONSUMPTION, 13X, 7HIN AREA, 4X
3, 4HPCT., 7X, 6HRETAIL, 10X, 6HRETAIL/3X, 6HCENTER, 16X, 7HCLASS I, 3X, 6HCH
4ANGE, 4X, 12HPRICE - CTS., 7X, 5HPRICE/5X, 3HNO., 6X, 4HNAME, 8X, 5HSALES, 4
5X, 22HYR.AGO --- PER 1/2 GAL., 6X, 6HSPREAD/5X, 3(1H-), 3X, 10(1H-), 3X, 9(
61H-), 2X, 6(1H-), 5X, 8(1H-), 7X, 8(1H-))
- 211 FORMAT (5X, I3, 3X, A10, F11.1, F8.1, F13.1, F15.1)
- 212 FORMAT (25X, 7(1H-), 3X, 5(1H-)/4X, SHTOTAL, 11X, F12.1, F8.1, 7X, 6(1H-), 9
1X, 6(1H-)/1X, 16HWEIGHTED AVERAGE, 24X, F12.1, F15.1)
- 213 FORMAT (1H1, 24X, 15HDECISION NAME -, 1X, A6, 16X, 5HPAGE , I3, 1H./25X, 22
1(1H=)//27X, 18HRAW MILK MOVEMENTS/10X, 51HFROM DIRECT SHIP SUPPLY AR
2EAS TO PROCESSING CENTERS/10X, 51(1H-)/26X, 7HQUARTER, I2, 1X, 4HYEAR, I
35/26X, 19(1H-)/3X, 4HFROM/2X, 5HORDER, 20X, 25HMOVEMENTS TO (MIL. LBS.
4)/3X, 3HNO., 1X, 5(2X, 11HNO. AMOUNT)/3X, 3(1H-), 2X, 5(1X, 12H---
5-))
- 214 FORMAT (3X, I3, 1X, 5(I5, F8.1), (/7X, I5, F8.1, I5, F8.1, I5, F8.1, I5, F8.1, I
15, F8.1))
- 215 FORMAT (1H0, 45HTOTAL MOVEMENTS TO PRIMARY PROCESSING CENTERS, 5X, 10
1H(MIL.LBS.), F10.1/1X, 43HTOTAL MOVEMENTS TO OTHER PROCESSING CENTER
2S, 7X, 10H(MIL.LBS.), F10.1)
- 216 FORMAT (1H1, 24X, 15HDECISION NAME -, 1X, A6, 16X, 5HPAGE , I3, 1H./25X, 22
1(1H=)//27X, 18HRAW MILK MOVEMENTS/16X, 40HFROM SUPPLY PLANTS TO PROC
2ESSING CENTERS/16X, 40(1H-)/26X, 7HQUARTER, I2, 1X, 4HYEAR, I5/26X, 19(1H
3-)/3X, 4HFROM/2X, 5HORDER, 20X, 25HMOVEMENTS TO (MIL. LBS.)/3X, 3HNO.,
41X, 5(2X, 11HNO. AMOUNT)/3X, 3(1H-), 2X, 5(1X, 12H---
5-))
- 217 FORMAT (1H0, 45HTOTAL MOVEMENTS TO PRIMARY PROCESSING CENTERS, 11H (MI
L.LBS.), F10.1/1X, 43HTOTAL MOVEMENTS TO OTHER PROCESSING CENTERS,
22X, 11H (MIL.LBS.), F10.1)
- 218 FORMAT (1H1, 24X, 15HDECISION NAME -, 1X, A6, 16X, 5HPAGE , I3, 1H./25X, 22
1(1H=)//27X, 18HRAW MILK MOVEMENTS/9X, 54HFROM DIRECT SHIP SUPPLY ARE
2AS TO MANUFACTURING CENTERS/9X, 54(1H-)/26X, 7HQUARTER, I2, 1X, 4HYEAR,
3I5/26X, 19(1H-)/3X, 4HFROM/2X, 5HORDER, 20X, 25HMOVEMENTS TO (MIL. LBS
4.)/3X, 3HNO., 1X, 5(2X, 11H ID AMOUNT)/3X, 3(1H-), 2X, 5(1X, 12H---
5-))
- 219 FORMAT (3X, I3, 1X, 5(2X, A3, F8.1), (/7X, 2X, A3, F8.1, 2X, A3, F8.1, 2X, A3, F8
1.1, 2X, A3, F8.1, 2X, A3, F8.1))
- 220 FORMAT (1H0, 40HTOTAL MOVEMENTS TO MANUFACTURING CENTERS, 6X, 11H (MI
L.LBS.), F10.1/1X, 39HTOTAL MOVEMENTS TO OTHER MANUFACTURING , 18HCEN
2TERS (MIL.LBS.), F10.1)
- 221 FORMAT (1H1, 24X, 15HDECISION NAME -, 1X, A6, 16X, 5HPAGE , I3, 1H./25X, 22
1(1H=)//27X, 18HRAW MILK MOVEMENTS/14X, 43HFROM SUPPLY PLANTS TO MANU
2FACTURING CENTERS/14X, 43(1H-)/26X, 7HQUARTER, I2, 1X, 4HYEAR, I5/26X, 19
3(1H-)/3X, 4HFROM/2X, 5HORDER, 20X, 25HMOVEMENTS TO (MIL. LBS.)/3X, 3HN
40., 1X, 5(2X, 11H ID AMOUNT)/3X, 3(1H-), 2X, 5(1X, 12H---
5-))
- 222 FORMAT (1H1, 24X, 15HDECISION NAME -, 1X, A6, 16X, 5HPAGE , I3, 1H./25X, 22
1(1H=)//25X, 23HPACKAGED MILK MOVEMENTS/13X, 46HFROM PROCESSING CENTE
2RS TO CONSUMPTION CENTERS/13X, 46(1H-)/26X, 7HQUARTER, I2, 1X, 4HYEAR, I
35/26X, 19(1H-)/3X, 4HFROM/3X, 5HORDER, 20X, 25HMOVEMENTS TO (MIL. LBS.
4)/3X, 3HNO., 1X, 5(2X, 11HNO. AMOUNT)/3X, 3(1H-), 2X, 5(1X, 12H---
5-))

- 223 FORMAT (1H0,51HTOTAL MOVEMENTS TO PRIMARY CONSUMPTION CENTERS (MIL 1,6H.LBS.),F10.1/1X,51HTOTAL MOVEMENTS TO OTHER CONSUMPTION CENTERS 2 (MIL,6H.LBS.),F10.1)
- 224 FORMAT (1H1,24X,15HDECISION NAME -,1X,AG,16X,4HPAGE,14,1H./25X,22(11H=)//23X,26HUNREGULATED GRADE A REPORT/23X,26(1H-)/26X,7HQUARTER, 2I2,5H YEAR,15)
- 225 FORMAT (26X,19(1H-)/65X,6HRETURN/43X,5HGROSS,6X,3HNET,8X,4HOVER/30 1X,7HCLASS I,6X,2(4HFARM,7X),6HDIRECT/3X,6HREGION,7X,53HPRODUCTION 2 CONSUMPTION PRICE PRICE COST/3X,6(1H-),7X,43H----- 3-----,12H -----/18X,2(7H(M LB 45),5X),3(7H(\$/CWT),4X)/)
- 226 FORMAT (3X,AG,2X,2(F12.1),3(4X,F7.2))
- 227 FORMAT (14X,2(4X,8(1H-))/3X,5HTOTAL,6X,2F12.1,3(4X,7(1H-))/3X,8HWT 1D AVE.,27X,3(4X,F7.2)/////////3X,22HCLASS I UTILIZATION = ,F5.2,8H P 2ERCENT)
- 228 FORMAT (1H1,24X,15HDECISION NAME -,1X,AG,16X,5HPAGE ,13,1H./25X,22 1(1H=)//23X,25HSUMMARY FOR ORDER MERGERS/23X,25(1H-)/27X,8HANNUAL - 2,1X,4HYEAR,16/27X,19(1H-)/65X,5HGROSS/34X,8HPRODUCER,3X,7HCLASS I, 33X,7HCLASS I,3X,5HBLEND/2X,6HMERGER,26X,8HRECEIPTS,4X,5HSALES,6X,3 4HUSE,5X,5HPRICE/2X,6HNUMBER,26X,8HMIL.LBS.,3X,8HMIL.LBS.,4X,4HPCT. 5,4X,5H\$/CWT/3X,4H-----,27X,8(1H-),3X,8(1H-),2X,7(1H-),3X,5(1H-))
- 229 FORMAT (1H1,23X,15HDECISION NAME -,1X,AG,17X,5HPAGE ,13,1H./24X,22 1(1H=)//28X,14HSUMMARY REPORT/28X,14(1H-)/25X,8HANNUAL -,1X,4HYEAR, 2I6/25X,19(1H-)/45X,4HTHIS,3X,20HCHANGE FROM YEAR AGO/44X,16HPERIOD 3 ABSOLUTE,2X,10HPERCENTAGE/44X,6(1H-),2X,8(1H-),2X,10(1H-))
- 230 FORMAT (1H1,23X,15HDECISION NAME -,1X,AG,17X,5HPAGE ,13,1H./24X,22 1(1H=)//25X,20HFEDERAL ORDER REPORT/25X,20(1H-)/26X,8HANNUAL -,1X,4 2HYEAR,16/26X,19(1H-)/28X,4HPCT.,18X,19HCLASS CLASS GROSS/18X,31H 3PRODUCER CHANGE CLASS I CLASS I,3X,2HII,4X,3HIII,3X,5HBLEND/1X,5H0 4RDER,12X,14HRECEIPTS FROM,4X,3HUSE,2X,4(2X,5HPRICE)/2X,3HNO.,5X,4 5HNAME,4X,15HMIL.LBS. YR.AGO,3X,4HPCT.,2X,4(1X,6H\$/CWT.)/2X,3(1H-), 62X,10(1H-),1X,8(1H-),1X,6(1H-),1X,7(1H-),1X,7(1H-),1X,6(1H-),1X,6(71H-),1X,6(1H-))
- 231 FORMAT (1H1,23X,15HDECISION NAME -,1X,AG,17X,5HPAGE ,13,1H./24X,22 1(1H=)//21X,28HFEDERAL ORDER REPORT - SALES/21X,28(1H-)/25X,8HANNUA 2L -,1X,4HYEAR,16/25X,19(1H-)/28X,4HPCT.,14X,4HPCT.,15X,4HPCT./18X, 333HCLASS I CHANGE CLASS II CHANGE,2X,17HCLASS III CHANGE/1X,5H 4ORDER,13X,5HSALES,4X,4HFROM,4X,5HSALES,5X,4HFROM,5X,5HSALES,5X,4HF 5ROM/2X,3HNO.,5X,4HNAME,4X,33HMIL.LBS. YR.AGO MIL.LBS. YR.AGO,3X, 616HMIL.LBS. YR.AGO/2X,3(1H-),2X,10(1H-),1X,8(1H-),1X,6(1H-),2X,8(71H-),2X,6(1H-),2X,9(1H-),2X,6(1H-))
- 232 FORMAT (1H1,18X,15HDECISION NAME -,1X,AG,22X,5HPAGE ,13,1H./19X,22 1(1H=)//19X,22HMILK PRODUCTION REPORT/19X,22(1H-)/21X,13HANNUAL - Y 2EAR,16/21X,19(1H-)/57X,6HRETURN/2X,6HSUPPLY,26X,4HPCT.,3X,5HGROSS, 34X,3HNET,5X,4HOVER/3X,4HAREA,14X,25HPRODUCTION CHANGE BLEND,4X,4 4HFARM,3X,6HDIRECT/4X,3HNO.,5X,4HNAME,6X,24HMIL. LBS. YR.AGO PRIC 5E,3X,5HPRICE,4X,4HCOST/4X,3(1H-),2X,10(1H-),2X,10(1H-),2X,6(1H-),2 6X,5(1H-),3X,5(1H-),3X,6(1H-))
- 233 FORMAT (1H1,23X,15HDECISION NAME -,1X,AG,17X,5HPAGE ,13,1H./24X,22 1(1H=)//23X,24HPROCESSING CENTER REPORT/23X,24(1H-)/26X,13HANNUAL - 2 YEAR,16/26X,19(1H-)/36X,2(5X,3HRAW),3X,8HPACKAGED/1X,10HPROCESSIN 3G,9X,8HPACKAGED,2X,8HCAPACITY,3X,4HMILK,4X,4HMILK,4X,4HMILK,4X,5HT 4OTAL/2X,6HCENTER,14X,4HMILK,6X,3HUSE,4X,7HACQUIS.,2X,6HTRANS.,2X,6 5HTRANS.,4X,4HCOST/4X,3HNO.,5X,4HNAME,6X,5HSALES,5X,4HPCT.,5X,4HCOS 6T,4X,4HCOST,4X,4HCOST,4X,8HPER CWT./4X,3(1H-),2X,10(1H-),2X,6(1H-) 7,4X,5(1H-),4X,6(1H-),2X,6(1H-),2X,6(1H-),3X,6(1H-))
- 234 FORMAT (1H1,18X,15HDECISION NAME -,1X,AG,22X,5HPAGE ,13,1H./19X,22 1(1H=)//16X,27HMANUFACTURING CENTER REPORT/16X,27(1H-)/21X,8HANNUAL 2 -,1X,4HYEAR,16/21X,19(1H-)/23X,12HMANUFACTURED,5X,8HCAPACITY/1X,6 3HCENTER,20X,4HMILK,8X,11HUTILIZATION/3X,2HID,8X,4HNAME,7X,10H(MIL. 4LBS.),6X,9H(PERCENT)/2X,4(1H-),4X,10(1H-),5X,8(1H-),8X,7(1H-))

- 235 FORMAT (1H1,22X,15HDECISION NAME -,1X,A6,18X,5HPAGE ,I3,1H./23X,22
1(1H=)//25X,18HCONSUMPTION REPORT/25X,18(1H-)/25X,8HANNUAL -,1X,4HY
2EAR,16/25X,19(1H-)/64X,4HFARM/1X,11HCONSUMPTION,13X,7HIN AREA,4X,4
3HPCT.,7X,6HRETAIL,10X,6HRETAIL/3X,6HCENTER,16X,7HCLASS I,3X,6HCHAN
4GE,4X,12HPRICE - CTS.,7X,5HPRICE/5X,3HNO.,6X,4HNAME,8X,5HSALES,4X,
522HYR.AGO PER 1/2 GAL.,6X,6HSPREAD/5X,3(1H-),3X,10(1H-),3X,9(1H
6-),2X,6(1H-),5X,8(1H-),7X,8(1H-))
- 236 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//27X,18HRAW MILK MOVEMENTS/10X,51HFROM DIRECT SHIP SUPPLY AR
2EAS TO PROCESSING CENTERS/10X,51(1H-)/26X,8HANNUAL -,1X,4HYEAR,16/
326X,19(1H-)/2X,5HORDER,20X,25HMOVEMENTS TO (MIL. LBS.)/3X,3HNO.,1
4X,5(2X,11HNO. AMOUNT)/3X,3(1H-),2X,5(1X,12H-----)
- 237 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//27X,18HRAW MILK MOVEMENTS/16X,40HFROM SUPPLY PLANTS TO PROC
2ESSING CENTERS/16X,40(1H-)/26X,8HANNUAL -,1X,4HYEAR,16/26X,19(1H-)
3/3X,4HFROM/2X,5HORDER,20X,25HMOVEMENTS TO (MIL. LBS.)/3X,3HNO.,1X
4,5(2X,11HNO. AMOUNT)/3X,3(1H-),2X,5(1X,12H-----)
- 238 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//27X,18HRAW MILK MOVEMENTS/9X,54HFROM DIRECT SHIP SUPPLY ARE
2AS TO MANUFACTURING CENTERS/9X,54(1H-)/26X,8HANNUAL -,1X,4HYEAR,16
3/26X,19(1H-)/2X,5HORDER,20X,25HMOVEMENTS TO (MIL. LBS.)/3X,3HNO.,
41X,5(2X,11H ID AMOUNT)/3X,3(1H-),2X,5(1X,12H-----)
- 239 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//27X,18HRAW MILK MOVEMENTS/14X,43HFROM SUPPLY PLANTS TO MANU
2FACTURING CENTERS/14X,43(1H-)/26X,8HANNUAL -,1X,4HYEAR,16/26X,19(1
3H-)/3X,4HFROM/2X,5HORDER,20X,25HMOVEMENTS TO (MIL. LBS.)/3X,3HNO.
4,1X,5(2X,11H ID AMOUNT)/3X,3(1H-),2X,5(1X,12H-----)
- 240 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//25X,23HPACKAGED MILK MOVEMENTS/13X,46HFROM PROCESSING CENTE
2RS TO CONSUMPTION CENTERS/13X,46(1H-)/26X,8HANNUAL -,1X,4HYEAR,16/
326X,19(1H-)/3X,4HFROM/3X,5HORDER,20X,25HMOVEMENTS TO (MIL. LBS.)/
43X,3HNO.,1X,5(2X,11HNO. AMOUNT)/3X,3(1H-),2X,5(1X,12H-----
5))
- 241 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,4HPAGE,I4,1H./25X,22(
11H=)//23X,26HUNREGULATED GRADE A REPORT/23X,26(1H-)/26X,13HANNUAL
2- YEAR,16/)
- 242 FORMAT (1X,A6)
- 243 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//23X,26HCOMPARATIVE SUMMARY REPORT/23X,26(1H-)/31X,7HQUARTER
2,I2/31X,9(1H-)/17X,6(5X,I4)/18X,6(3X,6H-----))
- 244 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//23X,26HCOMPARATIVE SUMMARY REPORT/23X,26(1H-)/33X,6HANNUAL/
233X,6(1H-)/17X,6(5X,I4)/18X,6(3X,6H-----))
- 245 FORMAT (1X,17HPROD RECPTS (MPD),6F9.0//1X,17HCLSS I SLS (MPD),6F9
1.0//1X,17HCLSS II SLS (MPD),6F9.0//1X,17HCLSS III SLS(MPD),6F9.0//
21X,17HCLSS I UTIL (PCT),6F9.1//1X,17HCLSS I PRICE (\$),6F9.2//1X,1
37HCLSS II PRICE (\$),6F9.2//1X,17HCLSS III PRICE(\$),6F9.2//1X,17HGR
4S BLND PRICE(\$),6F9.2//1X,17HULU OF POOL(000\$),6F9.0//1X,17HNET FR
5M PRICE (\$),6F9.2//1X,17HRTRN OVER D.C.(\$),6F9.2//1X,17HMKTG MARGI
6N(000\$),6F9.0//1X,17HCONS EXPND (000\$),6F9.0//1X,17HRETAIL PRICE(C
7TS),6F9.1//1X,17HFM-RT P SPRD(CTS),6F9.1//1X,17HPROC CP UTIL(PCT),
86F9.1//1X,17HMFG CP UTIL (PCT),6F9.1//1X,17HSP TRN CST (000\$),6F9.
90//1X,17HSP F-P CST (000\$),6F9.0//1X,17HDS TRN CST (000\$),6F9.0//1
*X,17HDS F-P CST (000\$),6F9.0//1X,17HPKG TRN CST(000\$),6F9.0//1X,17
*HMFG TRN CST(000\$),6F9.0//1X,17HRM PROD CST(000\$),6F9.0)
- 246 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/17X,38HCOMPARATIVE REPORT - PRODUCER
2 RECEIPTS/17X,38(1H-)/31X,7HQUARTER,I2/31X,9(1H-)/1X,5HORDER,22X,1
36H(MILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(
41H-),1X,6(3X,6H-----))

- 247 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/17X,38HCOMPARATIVE REPORT - PRODUCER
2 RECEIPTS/17X,38(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,16H(M
3ILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-)
4,1X,6(3X,6H-----))
- 248 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/19X,34HCOMPARATIVE REPORT - CLASS I
2SALES/19X,34(1H-)/31X,7HQUARTER,12/31X,9(1H-)/1X,5HORDER,22X,16H(M
3ILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-)
4,1X,6(3X,6H-----))
- 249 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/19X,34HCOMPARATIVE REPORT - CLASS I
2SALES/19X,34(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,16H(MILLI
3ON POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1X,
46(3X,6H-----))
- 250 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/15X,42HCOMPARATIVE REPORT - IN AREA
2CLASS I SALES/15X,42(1H-)/31X,7HQUARTER,12/31X,9(1H-)/1X,5HORDER,2
32X,16H(MILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X
4,10(1H-),1X,6(3X,6H-----))
- 251 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/15X,42HCOMPARATIVE REPORT - IN AREA
2CLASS I SALES/15X,42(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,1
36H(MILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(
41H-),1X,6(3X,6H-----))
- 252 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/19X,35HCOMPARATIVE REPORT - CLASS II
2 SALES/19X,35(1H-)/31X,7HQUARTER,12/31X,9(1H-)/1X,5HORDER,22X,16H(
3MILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-
4),1X,6(3X,6H-----))
- 253 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/19X,35HCOMPARATIVE REPORT - CLASS II
2 SALES/19X,35(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,16H(MILL
3ION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1X
4,6(3X,6H-----))
- 254 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/18X,36HCOMPARATIVE REPORT - CLASS II
2I SALES/18X,36(1H-)/31X,7HQUARTER,12/31X,9(1H-)/1X,5HORDER,22X,16H
3(MILLION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H
4-),1X,6(3X,6H-----))
- 255 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/18X,36HCOMPARATIVE REPORT - CLASS II
2I SALES/18X,36(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,16H(MIL
3LION POUNDS)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1
4X,6(3X,6H-----))
- 256 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/16X,40HCOMPARATIVE REPORT - CLASS I
2UTILIZATION/16X,40(1H-)/31X,7HQUARTER,12/31X,9(1H-)/1X,5HORDER,25X
3,9H(PERCENT)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1
4X,6(3X,6H-----))
- 257 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/16X,40HCOMPARATIVE REPORT - CLASS I
2UTILIZATION/16X,40(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,25X,9H(
3PERCENT)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1X,6(
43X,6H-----))
- 258 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/19X,34HCOMPARATIVE REPORT - CLASS I
2PRICE/19X,34(1H-)/31X,7HQUARTER,12/31X,9(1H-)/1X,5HORDER,22X,15H(D
3OLLARS /CWT.)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),
41X,6(3X,6H-----))

```

259 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/19X,34HCOMPARATIVE REPORT - CLASS I
2PRICE/19X,34(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,15H(DOLLA
3RS /CWT.)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1X,6
4(3X,6H-----))
260 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/20X,32HCOMPARATIVE REPORT - BLEND PR
2ICE/20X,32(1H-)/31X,7HQUARTER,I2/31X,9(1H-)/1X,5HORDER,22X,15H(DOL
3LARS /CWT.)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1X
4,6(3X,6H-----))
261 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/20X,32HCOMPARATIVE REPORT - BLEND PR
2ICE/20X,32(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,15H(DOLLARS
3 /CWT.)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H-),1X,6(3
4X,6H-----))
262 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/12X,48HCOMPARATIVE REPORT - PROCESSI
2NG CAP. UTILIZATION/12X,48(1H-)/31X,7HQUARTER,I2/31X,9(1H-)/1X,5H
3RDER,25X,9H(PERCENT)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,1
40(1H-),1X,6(3X,6H-----))
263 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/12X,48HCOMPARATIVE REPORT - PROCESSI
2NG CAP. UTILIZATION/12X,48(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER
3,25X,9H(PERCENT)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1H
4-),1X,6(3X,6H-----))
264 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/15X,42HCOMPARATIVE REPORT - TOTAL PR
2OCESSING COST/15X,42(1H-)/31X,7HQUARTER,I2/31X,9(1H-)/1X,5HORDER,2
32X,15H(DOLLARS /CWT.)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,
410(1H-),1X,6(3X,6H-----))
265 FORMAT (1H1,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,22
1(1H=)//29X,A10,A4/29X,14(1H-)/15X,42HCOMPARATIVE REPORT - TOTAL PR
2OCESSING COST/15X,42(1H-)/33X,6HANNUAL/33X,6(1H-)/1X,5HORDER,22X,1
35H(DOLLARS /CWT.)/2X,3HNO.,5X,4HNAME,3X,6(5X,I4)/2X,3(1H-),2X,10(1
4H-),1X,6(3X,6H-----))
266 FORMAT (1H1//////1X,70(1H*)/1X,70(1H*)/1X,70(1H*)/3(1X,5H*****,60X
1,5H*****//),1X,5H*****,4X,52HRUN TERMINATED BEFORE REACHING TIME PE
2RIOD SPECIFIED,4X,5H*****/3(1X,5H*****,60X,5H*****//),1X,70(1H*)/1X
3,70(1H*)/1X,70(1H*))

```

```

END
SUBROUTINE SCLE (IN, JN, AA, SCALE)

```

```

*****
SCALE VARIABLES IN COMPARATIVE REPORTS
*****

```

```

DIMENSION AA(59,59)
DO 1 I=1,IN
DO 1 J=1,JN
1 AA(I,J)=SCALE*AA(I,J)
RETURN

```

```

END
SUBROUTINE RITE (I, JM, JN, JO, IR, NOR, ONAME, AA, KW, NOUT1)

```

```

*****
WRITE COMPARATIVE REPORTS
*****

```



```
DIMENSION IR(61), NOR(75), ONAME(75), AA(59,59)
DO 2 J=JM, JN
  IJK=J
  IF (JO.LE.1) IJK=IR(J)
  IJ=J
  IF (JO.GT.1) IJ=J+16
  IF (KW.EQ.1) GO TO 1
  WRITE (NOUT1,3) NOR(IJ), ONAME(IJ), AA(IJK,I), AA(IJK,I+5), AA(IJK,
1 I+10), AA(IJK,I+15), AA(IJK,I+20), AA(IJK,I+25)
  GO TO 2
1 WRITE (NOUT1,4) NOR(IJ), ONAME(IJ), AA(IJK,I), AA(IJK,I+5), AA(IJK,
1 I+10), AA(IJK,I+15), AA(IJK,I+20), AA(IJK,I+25)
2 CONTINUE
3 FORMAT (2X, I3, 2X, A10, 1X, 6F9.1)
4 FORMAT (2X, I3, 2X, A10, 1X, 6F9.2)
RETURN
```

```
END
SUBROUTINE OUTX (NT9, AA, IR, NOR, NT, NOUT2, NOPT)
```

```
*****
WRITE OUT PROJECTED DATA FOR VALIDATION PROCESS
*****
```

```
DIMENSION AA(NT9,NT9), IR(NT), NOR(NT)
DO 2 I=1,NT9
  IF (NOPT.EQ.1) GO TO 1
  WRITE (NOUT2,3) NOR(I), (AA(I,J+5), J=1,5)
  GO TO 2
1 WRITE (NOUT2,3) NOR(I), (AA(IR(I),J+5), J=1,5)
2 CONTINUE
3 FORMAT (1X, I3, 1X, 5F13.0)
RETURN
```

```
END
SUBROUTINE OUTY (NT9, AA, NOUT2, I)
```

```
*****
WRITE OUT PROJECTED DATA FOR PLOTTING
*****
```

```
DIMENSION AA(NT9,NT9)
WRITE (NOUT2,1) (I, J, AA(I, J), AA(I, J+5), AA(I, J+10), AA(I, J+15), AA(I,
1 J+20), AA(I, J+25), J=1,5)
1 FORMAT (2(1X, I2), 6F11.2)
RETURN
```

```
END
```

DRW2

DRW2, listed on the following pages, manipulates and reports results from stages two and three of DAMPS. Selected, aggregated results from stage one are also reported. The reports generated by DRW2 are as follows:

1. Supply and Utilization Report
2. Comparative Supply and Utilization Report
3. Summary Report
4. Comparative Summary Report
5. Movements Report
 - a. Stage Two
 - b. Stage Three
6. Comparative Reports
 - a. Grade B Production
 - b. Total Manufacturing Milk Production
 - c. Class II Consumption
 - d. Cheese Consumption
 - e. Butter Consumption
 - f. Nonfat Dry Milk Consumption
 - g. Miscellaneous Class III Consumption
 - h. Ending Commercial Cheese Stocks
 - i. Ending Commercial Butter Stocks
 - j. Ending Commercial Nonfat Dry Milk Stocks
 - k. Ending Government Cheese Stocks
 - l. Ending Government Butter Stocks
 - m. Ending Government Nonfat Dry Milk Stocks

Movements reports are available only for quarters; all other reports contain quarterly and annual results. For further details on the content of these reports and how they are obtained, the reader is referred to Novakovic, et al. (8).

PROGRAM DRW2 (INPUT,OUTPUT,TAPE6=OUTPUT,TAPE41,TAPE42,TAPE43)

... DRW2 IS THE FIFTH PROGRAM OF DAMPS ...
DRW2 IS THE REPORT WRITER FOR THE MANUFACTURING MILK SECTION OF
THE DAMPS SIMULATOR. DRW2 WAS WRITTEN BY A.M. NOVAKOVIC AND
D.R. MARTELLA.

DIMENSION GAP(30,10), GAC(30,9), GBP(30,9), GMP(30,9), BIMP(30,4,2
1), BS(30,6,9), BC(30,5,5), BNRP(30,5), W1(9), W2(9), W3(9), W4(9),
2 W5(4,2), W6(6,9), W7(5,5), W8(5), NY2(6), NAME1(16), NAME2(3,14),
3 NAME3(2,5), AA(30,9), BB(30,5), SU(30,18), QRB(5), ICRB(14), SUP(
430), SURODC(30), BLEND(30), CEXP(30), CL3P(30), SUPB(30), RODCB(30
5), SUPAM(30), RODCAM(30), PAM(30), CEXPB(30), CEXPAM(30), GEXP(30)
6, GTP(30), TCM(30), BPC(9), BMKP(5), GNP(30,3), GPP(30,3), CF(5)
7, CL3PN(30), BMM1(31,15), BMM2(58,10), UNREG(30,10), UNC15(30,9),
8UNGBP(30,10), UNUP(30,10), UNRODC(30,10), UNCEXP(30), W9(20), CHGP
9D(30), CHGHC(5), TSTP(30)

DATA BS/1620*0.0/, BC/750*0.0/, BIMP/240*0.0/, SU/510*0.0/, GAP/270*0.
10/, GAC/270*0.0/, GBP/270*0.0/, GMP/270*0.0/, BNRP/150*0.0/, BLEND/30*
2.0/, CL3P/30*0.0/, SUP/30*0.0/, SURODC/30*0.0/, CEXP/30*0.0/, UNUP/300*
30.0/, UNGBP/300*0.0/, UNRODC/300*0.0/, UNREG/300*0.0/, CHGPD/30*0.0/, G
4EXP/30*0.0/, GNP/90*0.0/, RODCB/30*0.0/, CEXPB/30*0.0/, SUPB/30*0.0/

DATA NAME1/10HNORTHEAST, 10HCORN BELT, 10HLAKE, 10HSOUTHEAST
1, 10HS. CENTRAL, 10HPRAIRIE, 10HMOUNTAIN, 10HSOUTHWEST, 10HNORTHW
2EST, 10HEAST, 10HWEST, 10HNORTHEAST, 10HSOUTH, 10HN.
3 CENTRAL, 10HWEST, 10HPACIFIC /

DATA NAME2/10HTOTAL MFG, 10HMILK PRODU, SHCTION, 10HCLASS II C, 10HON
1SUMPTION, 5H, 10HCHEESE CON, 10HSUMPTION, 5H, 10HBUTTER CON
2, 10HSUMPTION, 5H, 10HNFD CONSU, 10HMPTION, 5H, 10HM. C
3CLASS I, 10HII CONSUMP, SHTION, 10HENDING COM, 10HM. CHEESE, 5HSTOCK, I
4HENDING COM, 10HM. BUTTER, 5HSTOCK, 10HENDING COM, 10HM. NFD ST, 5H
5CK, 10HENDING GOV, 10HT. CHEESE, 5HSTOCK, 10HENDING GOV, 10HT. BUTTE
6R, 5HSTOCK, 10HENDING GOV, 10HT. NFD ST, 5HOCK, 10HGR. B RETU, 10HRN
7 OUR DIR, 5H COST, 10HGRADE B MI, 10HLK PRODUCT, SHION /

DATA NAME3/10H(MILLION P, 7HOUNDS), 10H(DOLLARS /, 7H CWT.), 10H(THO
1USAND, 7HPOUNDS) /

INTEGER QRB

...DEFINE MODEL PARAMETERS AND LOGICAL FILE PARAMETERS, ETC....

MS=9
MIMP=2
MC=5
NIN1=41
NIN2=42
NIN3=43
NOUT=6
SCALE1=0.001
SCALE2=0.01
SCALE3=0.0001
IPAG=0
BCRK=0.0
BCRT=0.0

MS=NUMBER OF SUPPLY REGIONS
MIMP=NUMBER OF IMPORT REGIONS

MC=NUMBER OF DEMAND REGIONS
NIN1=INPUT FILE FROM DAMPSIN
NIN2=INPUT FILE FROM DAMPSLU
NIN3=INPUT FILE FROM DAMPCLC
NDUT=OUTPUT FILE

...READ BASE DATA VARIABLES FROM DAMPSIN...

```
DO 1 J=1,4
  READ (NIN1) BLEND(J),CL3P(J),SUP(J),SURODC(J),CEXP(J)
1 CONTINUE

DO 3 J=1,4
  DO 2 K=1,MS
    READ (NIN1) UNREG(J,K),UNC1S(J,K),UNGBP(J,K),UNUP(J,K),UNROD
1    C(J,K)
2 CONTINUE
    READ (NIN1) UNCEXP(J)
3 CONTINUE

  READ (NIN1) DNUM,NY2(1),NY,NQ,ORB,ICRB,((GAP(J,K),K=1,MS),J=1,4),
1(GAC(J,K),K=1,MS),J=1,4),((GBP(J,K),K=1,MS),J=1,4),((GMP(J,K),K=1,
2MS),J=1,4),((BIMP(J,I,K),I=1,4),K=1,MIMP),J=1,4),((BS(J,I,K),I=1
3,6),K=1,MS),J=1,4),((BC(J,I,K),I=1,5),K=1,MC),J=1,4),((BNRP(J,I),
4I=1,5),J=1,4),BPC,BMKP,CF,BFPHC,BPMC,BOPC,W9,IPDEF,CHGHC
```

REWIND NIN1

...SET UP YEAR AND QUARTER MARKERS...

```
DO 4 I=2,6
  NY2(I)=NY2(1)+I-1
4 CONTINUE
  NY1=NY+1
  NOF=(NY+1)*5
```

...SET W9 INTO CHCPD...

```
DO 5 J=1,5
  CHCPD(J)=1.0
5 CONTINUE
DO 6 JI=2,NY1
  JL=JI*5
  DO 6 JJ=1,4
    JK=JJ+5*(JI-1)
    JM=JJ+4*(JI-2)
    CHCPD(JK)=W9(JM)
    CHCPD(JL)=CHCPD(JL)+CHCPD(JK)/4.0
6 CONTINUE
```

...READ CURRENT QUARTER VARIABLES FROM DAMPSLU...

```
DO 10 J=6,NOF
  IF (J.EQ.10.OR.J.EQ.15.OR.J.EQ.20.OR.J.EQ.25.OR.J.EQ.30) GO TO
1 10
  READ (NIN2) W1,W2,W3,W4,W5,W6,W7,W8,(UNREG(J,K),K=1,MS),(UNC1S(
1 J,K),K=1,MS),(UNGBP(J,K),K=1,MS),(UNUP(J,K),K=1,MS),(UNRODC(J,K
2 ),K=1,MS),UNCEXP(J),BM11,BM12,TRCS1,TRCS2
  WRITE (NIN1) BM11,BM12
  DO 7 K=1,MS
```

```
GAP(J,K)=W1(K)
GBP(J,K)=W2(K)
GAC(J,K)=W3(K)
GMP(J,K)=W4(K)+UNREG(J,K)-UNCIS(J,K)
DO 7 I=1,6
  BS(J,I,K)=W6(I,K)
7 CONTINUE
DO 8 I=1,4
  DO 8 K=1,MIMP
    BIMP(J,I,K)=W5(I,K)
8 CONTINUE
DO 9 I=1,5
  BMRP(J,I)=W8(I)
  DO 9 K=1,MC
    BC(J,I,K)=W7(I,K)
9 CONTINUE
TCMP(J)=TRCS1+TRCS2
10 CONTINUE

...READ GRADE A VARIABLES FROM DAMPLC...

DO 11 I=6,NQF
  IF (I.EQ.10.OR.I.EQ.15.OR.I.EQ.20.OR.I.EQ.25.OR.I.EQ.30) GO TO
1 11
  READ (NIN3) SUP(I),SURDDC(I),BLEND(I),CEXP(I),CL3P(I)
11 CONTINUE

...ANNUAL SUMMATION...

DO 15 JI=1,NY1
  JL=JI*5
  JM=JL-1
  DO 15 JJ=1,4
    JK=JJ+5*(JI-1)
    DO 12 K=1,MS
      GAP(JL,K)=GAP(JL,K)+GAP(JK,K)
      GAC(JL,K)=GAC(JL,K)+GAC(JK,K)
      GBP(JL,K)=GBP(JL,K)+GBP(JK,K)
      GMP(JL,K)=GMP(JL,K)+GMP(JK,K)
      UNREG(JL,K)=UNREG(JL,K)+UNREG(JK,K)
      UNCIS(JL,K)=UNCIS(JL,K)+UNCIS(JK,K)
      UNUP(JL,K)=UNUP(JL,K)+UNUP(JK,K)
    DO 12 I=1,6
      BS(JL,I,K)=BS(JM,I,K)
12 CONTINUE
    DO 13 I=1,5
      DO 13 K=1,MC
        BC(JL,I,K)=BC(JL,I,K)+BC(JK,I,K)
13 CONTINUE
    DO 14 I=1,4
      DO 14 K=1,MIMP
        BIMP(JL,I,K)=BIMP(JL,I,K)+BIMP(JK,I,K)
14 CONTINUE
    TCMP(JL)=TCMP(JL)+TCMP(JK)
    SUP(JL)=SUP(JL)+SUP(JK)
    SURDDC(JL)=SURDDC(JL)+SURDDC(JK)/4.0
    BLEND(JL)=BLEND(JL)+BLEND(JK)/4.0
    CEXP(JL)=CEXP(JL)+UNCEXP(JK)
    CEXP(JL)=CEXP(JL)+CEXP(JK)
    CL3P(JL)=CL3P(JL)+CL3P(JK)/4.0
```

15 CONTINUE

...REGIONAL SUMMATION AND CONVERT BUTTER AND NFDM TO M.E....

```
DO 19 J=1,NQF
DO 16 K=1,MS
SU(J,1)=SU(J,1)+GAP(J,K)+UNREG(J,K)
SU(J,2)=SU(J,2)+GBP(J,K)
GAP(J,10)=GAP(J,10)+GAP(J,K)
UNREG(J,10)=UNREG(J,10)+UNREG(J,K)
UNUP(J,10)=UNUP(J,10)+UNUP(J,K)
DO 16 I=1,6
II=I+6
SU(J,II)=SU(J,II)+BS(J,I,K)
16 CONTINUE
DO 17 K=1,MC
SU(J,13)=SU(J,13)+GAC(J,K)+UNC1S(J,K)
DO 17 I=1,5
II=I+13
SU(J,II)=SU(J,II)+BC(J,I,K)
17 CONTINUE
DO 18 K=1,MIMP
DO 18 I=1,4
II=I+2
SU(J,II)=SU(J,II)+BIMP(J,I,K)
18 CONTINUE
SU(J,5)=SU(J,5)*100.0/CF(4)
SU(J,6)=SU(J,6)*100.0/CF(5)
SU(J,8)=SU(J,8)*100.0/CF(4)
SU(J,9)=SU(J,9)*100.0/CF(5)
SU(J,11)=SU(J,11)*100.0/CF(4)
SU(J,12)=SU(J,12)*100.0/CF(5)
SU(J,17)=SU(J,17)*100.0/CF(4)
SU(J,18)=SU(J,18)*100.0/CF(5)
19 CONTINUE
```

...COMPUTE UNREGULATED AVERAGE PRICE AND RETURN...

```
DO 21 JI=1,NY1
JL=JI*5
DO 21 JJ=1,4
JK=JJ+5*(JI-1)
DO 20 K=1,MS
UNGBP(JK,10)=UNGBP(JK,10)+UNGBP(JK,K)*UNREG(JK,K)/UNREG(JK,10)
1 UNRODC(JK,10)=UNRODC(JK,10)+UNRODC(JK,K)*UNREG(JK,K)/UNREG(JK,10)
20 CONTINUE
UNGBP(JL,10)=UNGBP(JL,10)+UNGBP(JK,10)/4.0
UNRODC(JL,10)=UNRODC(JL,10)+UNRODC(JK,10)/4.0
21 CONTINUE
```

...CONVERT CHEESE TO PROD. WT. AND ALL PRICES TO \$/LB. OF PROD....

```
DO 25 J=1,NQF
DO 22 K=1,MS
BS(J,1,K)=BS(J,1,K)*CF(2)/100.0
BS(J,4,K)=BS(J,4,K)*CF(2)/100.0
22 CONTINUE
DO 23 K=1,MC
```

```

      BC(J,2,K)=BC(J,2,K)*CF(2)/100.0
23  CONTINUE
      DO 24 K=1,MIMP
          BIMP(J,1,K)=BIMP(J,1,K)*CF(2)/100.0
24  CONTINUE
      BNRP(J,1)=BNRP(J,1)*SCALE3
      BNRP(J,3)=BNRP(J,3)*SCALE3
      BNRP(J,2)=(BNRP(J,2)/CF(2))/100.0
      BNRP(J,4)=BNRP(J,4)/100.0
      BNRP(J,5)=BNRP(J,5)/100.0
25  CONTINUE

      ...COMPUTE WTD. AVE. ANNUAL RETAIL PRICES...

      DO 27 I=1,5
      DO 27 JI=1,NY1
          JL=JI*5
      DO 27 JJ=1,4
          JK=JJ+5*(JI-1)
          DO 26 K=1,MC
              BCRK=BCRK+BC(JK,I,K)
              BCRT=BCRT+BC(JL,I,K)
26  CONTINUE
          BNRP(JL,I)=BNRP(JL,I)+BNRP(JK,I)*BCRK/BCRT
          BCRK=0.0
          BCRT=0.0
27  CONTINUE

      *****
      SUPPLY AND UTILIZATION REPORTS
      *****

      ...WRITE SUPPLY AND UTILIZATION REPORT...

      DO 29 L=1,NY1
          LI=L-1
          IF (LI.EQ.0) GO TO 28
          IF (QRB(LI).NE.1.AND.QRB(LI).NE.2) GO TO 29
28  IPAG=IPAG+1
          WRITE (NOUT,112) DNUM,IPAG
          WRITE (NOUT,113) NY2(L)
          LM=(L-1)*5+1
          LN=L*5
          WRITE (NOUT,114) ((SU(LL,I),LL=LM,LN),I=1,18)
29  CONTINUE

      ...WRITE COMPARATIVE SUPPLY AND UTILIZATION REPORT...

      IPAG=IPAG+1
      WRITE (NOUT,112) DNUM,IPAG
      WRITE (NOUT,115) (NY2(L),L=2,6)
      WRITE (NOUT,114) ((SU(LL,I),LL=10,30,5),I=1,18)

      *****
      SUMMARY REPORTS
      *****

      ...COMPUTATIONS FOR SUMMARY REPORTS...

      CALL SCLE (1,NCF,CEXP,SCALE3)
```

```
GNP(1,1)=-2500.  
GNP(1,2)=-25300.  
GNP(1,3)=199600.  
GNP(5,1)=-400.  
GNP(5,2)=14100.  
GNP(5,3)=-35200.  
DO 42 J=1,NCF  
  BLEND(J)=(BLEND(J)*GAP(J,10)+UNGBP(J,10)*UNREG(J,10))/SU(J,1)  
  SUP(J)=SUP(J)+UNUP(J,10)  
  SUPB(J)=SUPB(J)+SU(J,2)*CL3P(J)  
  SUPAM(J)=SUP(J)+SUPB(J)  
  PAM(J)=(BLEND(J)*SU(J,1)+CL3P(J)*SU(J,2))/(SU(J,1)+SU(J,2))  
  DO 30 I=1,5  
  DO 30 L=1,MC  
    CEXPB(J)=CEXPB(J)+BNRP(J,I)*BC(J,I,L)  
30  CONTINUE  
    CEXPAM(J)=CEXP(J)+CEXPB(J)  
    DO 34 I1=1,3  
      I2=I1+3  
      IF (J.EQ.1.OR.J.EQ.5) GO TO 33  
      DO 32 K=1,MS  
        IF (J.EQ.10.OR.J.EQ.15.OR.J.EQ.20.OR.J.EQ.25.OR.J.EQ.30)  
1          GO TO 31  
          GNP(J,I1)=GNP(J,I1)+BS(J,I2,K)-BS(J-1,I2,K)  
          GO TO 32  
31          GNP(J,I1)=GNP(J,I1)+BS(J,I2,K)-BS(J-5,I2,K)  
32  CONTINUE  
33  CONTINUE  
      I3=I1+2  
      IF (I1.EQ.1) I3=2  
      GPP(J,I1)=BNRP(J,I3)/BMKP(I3)  
      CEXP(J)=CEXP(J)+GPP(J,I1)*GNP(J,I1)  
34  CONTINUE  
      IF (J.EQ.1.OR.J.EQ.5) GO TO 38  
      IF (J.EQ.10.OR.J.EQ.15.OR.J.EQ.20.OR.J.EQ.25.OR.J.EQ.30) GO TO  
1 35  
      CNPME=SU(J,7)-SU(J-1,7)  
      CNPB=SU(J,8)-SU(J-1,8)  
      CNPN=SU(J,9)-SU(J-1,9)  
      GO TO 36  
35  CNPME=SU(J,7)-SU(J-5,7)  
      CNPB=SU(J,8)-SU(J-5,8)  
      CNPN=SU(J,9)-SU(J-5,9)  
36  CCHK=CNPB*CNPN  
      IF (CCHK.LT.0.0) GO TO 37  
      IF (CNPB.LT.CNPN) CNPME=CNPME+CNPB  
      IF (CNPN.LE.CNPB) CNPME=CNPME+CNPN  
      GO TO 38  
37  CNPME=CNPME+CNPB+CNPN  
38  GNPME=GNP(J,1)*100.0/CF(2)  
      GNPB=GNP(J,2)*100.0/CF(4)  
      GNPN=GNP(J,3)*100.0/CF(5)  
      GCHK=GNPB*GNPN  
      IF (GCHK.LT.0.0) GO TO 39  
      IF (GNPB.GT.CNPN) GNPME=GNPME+GNPB  
      IF (GNPN.GE.CNPB) GNPME=GNPME+GNPN  
      GO TO 40  
39  GNPME=GNPME+GNPB+GNPN  
40  GTP(J)=GNPME/(SU(J,1)+SU(J,2))*100.0  
      TSTP(J)=(CNPME+GNPME)/(SU(J,1)+SU(J,2))*100.0
```



```

        TPME=SU(J,1)+SU(J,2)
        IF (J.EQ.1) PRINT 121
        PRINT 41, CNPME,GNPME,TPME
41  FORMAT (1X,3HPME,3F14.0)
42  CONTINUE
        TSTP(1)=5.2
        TSTP(5)=1.7
        DO 43 J=5,NOF,5
            SUPB(J)=0.0
            SUPAM(J)=0.0
            CEXPB(J)=0.0
            CEXPAM(J)=0.0
            GEXP(J)=0.0
43  CONTINUE
        DO 45 JI=1,NY1
            JL=JI*5
            DO 44 JJ=1,4
                JK=JJ+5*(JI-1)
                SUPB(JL)=SUPB(JL)+SUPB(JK)
                SUPAM(JL)=SUPAM(JL)+SUPAM(JK)
                CEXPB(JL)=CEXPB(JL)+CEXPB(JK)
                CEXPAM(JL)=CEXPAM(JL)+CEXPAM(JK)
                GEXP(JL)=GEXP(JL)+GEXP(JK)
44  CONTINUE
            BLEND(JL)=SUP(JL)/SU(JL,1)
            CL3P(JL)=SUPB(JL)/SU(JL,2)
            PAM(JL)=SUPAM(JL)/(SU(JL,1)+SU(JL,2))
45  CONTINUE
            DO 47 J=1,NOF
                Y5=FLOAT(J)/4.0+0.76
                NY5=IFIX(Y5)
                CL3PN(J)=CL3P(J)-BFPHC*CHGHC(NY5)-BPMC
                DO 46 K=1,MS
                    BPC1=BPC(K)
                    IF (IPDEF.EQ.1) BPC1=BPC(K)*CHGPD(J)
                    RODCB(J)=RODCB(J)+(CL3PN(J)-BPC1-BOPC)*GBP(J,K)/SU(J,2)
46  CONTINUE
                    BPC1=0.0
                    SURDDC(J)=(SURDDC(J)*GAP(J,10)+UNRODC(J,10)*UNREG(J,10))/SU(J,1)
                )
                RODCAM(J)=(SURDDC(J)*SU(J,1)+RODCB(J)*SU(J,2))/(SU(J,1)+SU(J,2))
                )
            )
47  CONTINUE

        CALL SCLE (1,NOF,BLEND,SCALE2)
        CALL SCLE (1,NOF,CL3P,SCALE2)
        CALL SCLE (1,NOF,PAM,SCALE2)
        CALL SCLE (1,NOF,SUP,SCALE3)
        CALL SCLE (1,NOF,SUPB,SCALE3)
        CALL SCLE (1,NOF,SUPAM,SCALE3)
        CALL SCLE (1,NOF,SURDDC,SCALE2)
        CALL SCLE (1,NOF,RODCB,SCALE2)
        CALL SCLE (1,NOF,RODCAM,SCALE2)
        CALL SCLE (1,NOF,TMP,SCALE2)
        CALL SCLE (1,NOF,TMP,SCALE1)

```

...WRITE SUMMARY REPORT...

IS 49 K=1,NY1
KI=K-1

```
IF (KI.EQ.0) GO TO 48
IF (ORB(KI).NE.1.AND.ORB(KI).NE.2) GO TO 49
48 IPAG=IPAG+1
WRITE (NOUT,112) DNUM,IPAG
WRITE (NOUT,116) NY2(K)
KM=(K-1)*5+1
KN=K*5
WRITE (NOUT,117) (BLEND(KK),KK=KM,KN), (CL3P(KK),KK=KM,KN), (PAM(
1 KK),KK=KM,KN), (SUP(KK),KK=KM,KN), (SUPB(KK),KK=KM,KN), (SUPAM(KK)
2 ,KK=KM,KN), (SURODC(KK),KK=KM,KN), (RODCB(KK),KK=KM,KN), (RODCAM(K
3 K),KK=KM,KN), (TCMP(KK),KK=KM,KN), (CEXP(KK),KK=KM,KN), (CEXPB(KK)
4 ,KK=KM,KN), (CEXPAM(KK),KK=KM,KN), (GEXP(KK),KK=KM,KN), (GPTP(KK),
5 KK=KM,KN), (TSTP(KK),KK=KM,KN)
49 CONTINUE
```

...WRITE COMPARATIVE SUMMARY REPORT...

```
IPAG=IPAG+1
WRITE (NOUT,112) DNUM,IPAG
WRITE (NOUT,118) (NY2(L),L=2,6)
WRITE (NOUT,117) (BLEND(K),K=10,30,5), (CL3P(K),K=10,30,5), (PAM(K),
1K=10,30,5), (SUP(K),K=10,30,5), (SUPB(K),K=10,30,5), (SUPAM(K),K=10,3
20,5), (SURODC(K),K=10,30,5), (RODCB(K),K=10,30,5), (RODCAM(K),K=10,30
3,5), (TCMP(K),K=10,30,5), (CEXP(K),K=10,30,5), (CEXPB(K),K=10,30,5), (
4CEXPAM(K),K=10,30,5), (GEXP(K),K=10,30,5), (GPTP(K),K=10,30,5), (TSTP
5(K),K=10,30,5)
```

COMPARATIVE REPORTS

...GRADE B MILK PRODUCTION...

```
IF (ICRB(1).EQ.0) GO TO 51
CALL SCLE (MS,NQF,GBP,SCALE1)
J1=1
IF (ICRB(1).EQ.2) J1=5
DO 50 J=J1,5
IPAG=IPAG+1
IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,14),K=1,3),J,(N
1 AME3(K,3),K=1,2), (NY2(K),K=1,6)
IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,14),K=1,3), (NAM
1 E3(K,1),K=1,2), (NY2(K),K=1,6)
CALL RITE (J,1,MS,0,NAME1,GBP,0,NOUT)
50 CONTINUE
```

...TOTAL MFG. MILK PRODUCTION...

```
51 IF (ICRB(2).EQ.0) GO TO 53
CALL SCLE (MS,NQF,GMP,SCALE1)
J1=1
IF (ICRB(2).EQ.2) J1=5
DO 52 J=J1,5
IPAG=IPAG+1
IF (J.EQ.1) WRITE (NOUT,121)
IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,1),K=1,3),J,(NA
1 ME3(K,1),K=1,2), (NY2(K),K=1,6)
IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,1),K=1,3), (NAME
1 3(K,1),K=1,2), (NY2(K),K=1,6)
CALL RITE (J,1,MS,0,NAME1,GMP,0,NOUT)
```

52 CONTINUE

...CLASS II CONSUMPTION...

53 IF (ICRB(3).EQ.0) GO TO 56
DO 54 K=1,MC
DO 54 J=1,30
BB(J,K)=BC(J,1,K)

54 CONTINUE
CALL SCLE (MC,NGF,BB,SCALE1)
J1=1

IF (ICRB(3).EQ.2) J1=5
DO 55 J=J1,5

IPAG=IPAG+1
IF (J.EQ.1) WRITE (NOUT,121)
IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,2),K=1,3),J,(NAME3(K,1),K=1,2),(NY2(K),K=1,6)
1 ME3(K,1),K=1,2),(NY2(K),K=1,6)
IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,2),K=1,3),(NAME3(K,1),K=1,2),(NY2(K),K=1,6)
1 3(K,1),K=1,2),(NY2(K),K=1,6)
CALL RITE (J,1,MC,1,NAME1,BB,0,NOUT)

55 CONTINUE

...CHEESE CONSUMPTION...

56 IF (ICRB(4).EQ.0) GO TO 59
DO 57 K=1,MC
DO 57 J=1,30
BB(J,K)=BC(J,2,K)

57 CONTINUE
CALL SCLE (MC,NGF,BB,SCALE1)
J1=1

IF (ICRB(4).EQ.2) J1=5
DO 58 J=J1,5

IPAG=IPAG+1
IF (J.EQ.1) WRITE (NOUT,121)
IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,3),K=1,3),J,(NAME3(K,1),K=1,2),(NY2(K),K=1,6)
1 ME3(K,1),K=1,2),(NY2(K),K=1,6)
IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,3),K=1,3),(NAME3(K,1),K=1,2),(NY2(K),K=1,6)
1 3(K,1),K=1,2),(NY2(K),K=1,6)
CALL RITE (J,1,MC,1,NAME1,BB,0,NOUT)

58 CONTINUE

...BUTTER CONSUMPTION...

59 IF (ICRB(5).EQ.0) GO TO 62
DO 60 K=1,MC
DO 60 J=1,30
BB(J,K)=BC(J,4,K)

60 CONTINUE
CALL SCLE (MC,NGF,BB,SCALE1)
J1=1

IF (ICRB(5).EQ.2) J1=5
DO 61 J=J1,5

IPAG=IPAG+1
IF (J.EQ.1) WRITE (NOUT,121)
IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,4),K=1,3),J,(NAME3(K,1),K=1,2),(NY2(K),K=1,6)
1 ME3(K,1),K=1,2),(NY2(K),K=1,6)
IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,4),K=1,3),(NAME3(K,1),K=1,2),(NY2(K),K=1,6)
1 3(K,1),K=1,2),(NY2(K),K=1,6)
CALL RITE (J,1,MC,1,NAME1,BB,0,NOUT)

61 CONTINUE

...NONFAT DRY MILK CONSUMPTION...

62 IF (ICRB(6).EQ.0) GO TO 65

DO 63 K=1,MC

DO 63 J=1,30

BB(J,K)=BC(J,5,K)

63 CONTINUE

CALL SCLE (MC,NQF,BB,SCALE1)

J1=1

IF (ICRB(6).EQ.2) J1=5

DO 64 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,5),K=1,3),J,(NA

1 ME3(K,1),K=1,2),(NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,5),K=1,3),(NAME

1 3(K,1),K=1,2),(NY2(K),K=1,6)

CALL RITE (J,1,MC,1,NAME1,BB,0,NOUT)

64 CONTINUE

...MISC. CLASS III CONSUMPTION...

65 IF (ICRB(7).EQ.0) GO TO 68

DO 66 K=1,MC

DO 66 J=1,30

BB(J,K)=BC(J,3,K)

66 CONTINUE

CALL SCLE (MC,NQF,BB,SCALE1)

J1=1

IF (ICRB(7).EQ.2) J1=5

DO 67 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,6),K=1,3),J,(NA

1 ME3(K,1),K=1,2),(NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,6),K=1,3),(NAME

1 3(K,1),K=1,2),(NY2(K),K=1,6)

CALL RITE (J,1,MC,1,NAME1,BB,0,NOUT)

67 CONTINUE

...ENDING COMMERCIAL CHEESE STOCK...

68 IF (ICRB(8).EQ.0) GO TO 71

DO 69 K=1,MS

DO 69 J=1,30

AA(J,K)=BS(J,1,K)

69 CONTINUE

CALL SCLE (MS,NQF,AA,SCALE1)

J1=1

IF (ICRB(8).EQ.2) J1=5

DO 70 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,7),K=1,3),J,(NA

1 ME3(K,1),K=1,2),(NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,7),K=1,3),(NAME

1 3(K,1),K=1,2),(NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,0,NOUT)

70 CONTINUE

...ENDING COMMERCIAL BUTTER STOCKS...

71 IF (ICRB(9).EQ.0) GO TO 74

DO 72 K=1,MS

DO 72 J=1,30

AA(J,K)=BS(J,2,K)

72 CONTINUE

CALL SCLE (MS,NBF,AA,SCALE1)

J1=1

IF (ICRB(9).EQ.2) J1=5

DO 73 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,8),K=1,3),J,(NA

1 ME3(K,1),K=1,2),(NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,8),K=1,3),(NAME

1 3(K,1),K=1,2),(NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,0,NOUT)

73 CONTINUE

...ENDING COMMERCIAL NONFAT DRY MILK STOCKS...

74 IF (ICRB(10).EQ.0) GO TO 77

DO 75 K=1,MS

DO 75 J=1,30

AA(J,K)=BS(J,3,K)

75 CONTINUE

CALL SCLE (MS,NBF,AA,SCALE1)

J1=1

IF (ICRB(10).EQ.2) J1=5

DO 76 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,9),K=1,3),J,(NA

1 ME3(K,1),K=1,2),(NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,9),K=1,3),(NAME

1 3(K,1),K=1,2),(NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,0,NOUT)

76 CONTINUE

...ENDING GOVERNMENT CHEESE STOCKS...

77 IF (ICRB(11).EQ.0) GO TO 80

DO 78 K=1,MS

DO 78 J=1,30

AA(J,K)=BS(J,4,K)

78 CONTINUE

CALL SCLE (MS,NBF,AA,SCALE1)

J1=1

IF (ICRB(11).EQ.2) J1=5

DO 79 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM,IPAG,(NAME2(K,10),K=1,3),J,(N

1 AME3(K,1),K=1,2),(NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM,IPAG,(NAME2(K,10),K=1,3),(NAM

1 E3(K,1),K=1,2),(NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,0,NOUT)

79 CONTINUE

...ENDING GOVERNMENT BUTTER STOCKS...

80 IF (ICRB(12).EQ.0) GO TO 83

DO 81 K=1,MS

DO 81 J=1,30

AA(J,K)=BS(J,5,K)

81 CONTINUE

CALL SCLE (MS,NQF,AA,SCALE1)

J1=1

IF (ICRB(12).EQ.2) J1=5

DO 82 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM, IPAG, (NAME2(K,11),K=1,3), J, (N

1 AME3(K,1),K=1,2), (NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM, IPAG, (NAME2(K,11),K=1,3), (NAM

1 E3(K,1),K=1,2), (NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,0,NOUT)

82 CONTINUE

...ENDING GOVERNMENT NONFAT DRY MILK STOCKS...

83 IF (ICRB(13).EQ.0) GO TO 86

DO 84 K=1,MS

DO 84 J=1,30

AA(J,K)=BS(J,6,K)

84 CONTINUE

CALL SCLE (MS,NQF,AA,SCALE1)

J1=1

IF (ICRB(13).EQ.2) J1=5

DO 85 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM, IPAG, (NAME2(K,12),K=1,3), J, (N

1 AME3(K,1),K=1,2), (NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM, IPAG, (NAME2(K,12),K=1,3), (NAM

1 E3(K,1),K=1,2), (NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,0,NOUT)

85 CONTINUE

... ..

86 IF (ICRB(14).EQ.0) GO TO 89

J1=1

IF (ICRB(14).EQ.2) J1=5

DO 87 J=J1,5

IPAG=IPAG+1

IF (J.EQ.1) WRITE (NOUT,121)

IF (J.LT.5) WRITE (NOUT,119) DNUM, IPAG, (NAME2(K,13),K=1,3), J, (N

1 AME3(K,2),K=1,2), (NY2(K),K=1,6)

IF (J.EQ.5) WRITE (NOUT,120) DNUM, IPAG, (NAME2(K,13),K=1,3), (NAM

1 E3(K,2),K=1,2), (NY2(K),K=1,6)

CALL RITE (J,1,MS,0,NAME1,AA,1,NOUT)

88 CONTINUE

ELEMENTS REPORTS

```
REWIND NINI
DO 106 I=1,ND
  READ (NINI) BMM1,BMM2
  DO 90 II=1,58
    DO 89 J=1,5
      BMM2(II,J)=BMM2(II,J)*100.0/CF(4)
89    CONTINUE
    DO 90 J=6,10
      BMM2(II,J)=BMM2(II,J)*100.0/CF(5)
90    CONTINUE
    Y3=FLOAT(I)/4.0+0.76
    NY3=IFIX(Y3)
    IQ=I-4*(NY3-1)
    IF (ORB(NY3).NE.1.AND.ORB(NY3).NE.3) GO TO 106

      .....STAGE 1.....

    IPAG=IPAG+1
    WRITE (NOUT,112) DNUM,IPAG
    WRITE (NOUT,123) IQ,NY2(NY3+1),(NAME1(JJJ),JJJ=12,16)
    WRITE (NOUT,124)
    DO 91 J=1,9
      IF (ICLK(BMM1,31,15,J,1).LE.0) GO TO 91
      WRITE (NOUT,122) NAME1(J),(BMM1(J,K),K=1,5)
91    CONTINUE
    WRITE (NOUT,125)
    DO 92 J=1,9
      IF (ICLK(BMM1,31,15,J,6).LE.0) GO TO 92
      WRITE (NOUT,122) NAME1(J),(BMM1(J,K),K=6,10)
92    CONTINUE
    WRITE (NOUT,126)
    DO 93 J=10,11
      IF (ICLK(BMM1,31,15,J,6).LE.0) GO TO 93
      WRITE (NOUT,122) NAME1(J),(BMM1(J,K),K=6,10)
93    CONTINUE
    WRITE (NOUT,127)
    DO 94 J=14,22
      JK=J-13
      IF (ICLK(BMM1,31,15,J,6).LE.0) GO TO 94
      WRITE (NOUT,122) NAME1(JK),(BMM1(J,K),K=6,10)
94    CONTINUE
    WRITE (NOUT,128)
    DO 95 J=23,31
      JK=J-22
      IF (ICLK(BMM1,31,15,J,6).LE.0) GO TO 95
      WRITE (NOUT,122) NAME1(JK),(BMM1(J,K),K=6,10)
95    CONTINUE
    WRITE (NOUT,129)
    DO 96 J=1,9
      IF (ICLK(BMM1,31,15,J,11).LE.0) GO TO 96
      WRITE (NOUT,122) NAME1(J),(BMM1(J,K),K=11,15)
96    CONTINUE
    WRITE (NOUT,126)
    DO 97 J=12,13
      IF (ICLK(BMM1,31,15,J,11).LE.0) GO TO 97
      WRITE (NOUT,122) NAME1(J-2),(BMM1(J,K),K=11,15)
97    CONTINUE
```

.....STAGE 2.....

```
IPAG=IPAG+1
WRITE (NOUT,112) DNUM,IPAG
WRITE (NOUT,123) IQ,NY2(NY3+1),(NAME1(JJJ),JJJ=12,16)
WRITE (NOUT,130)
DO 98 J=1,9
  IF (ICBK(BMM2,58,10,J,1).LE.0) GO TO 98
  WRITE (NOUT,122) NAME1(J),(BMM2(J,K),K=1,5)
98 CONTINUE
WRITE (NOUT,126)
DO 99 J=10,11
  IF (ICBK(BMM2,58,10,J,1).LE.0) GO TO 99
  WRITE (NOUT,122) NAME1(J),(BMM2(J,K),K=1,5)
99 CONTINUE
WRITE (NOUT,127)
DO 100 J=12,20
  JK=J-11
  IF (ICBK(BMM2,58,10,J,1).LE.0) GO TO 100
  WRITE (NOUT,122) NAME1(JK),(BMM2(J,K),K=1,5)
100 CONTINUE
WRITE (NOUT,126)
DO 101 J=21,29
  JK=J-20
  IF (ICBK(BMM2,58,10,J,1).LE.0) GO TO 101
  WRITE (NOUT,122) NAME1(JK),(BMM2(J,K),K=1,5)
101 CONTINUE
WRITE (NOUT,131)
DO 102 J=30,38
  JK=J-29
  IF (ICBK(BMM2,58,10,J,6).LE.0) GO TO 102
  WRITE (NOUT,122) NAME1(JK),(BMM2(J,K),K=6,10)
102 CONTINUE
WRITE (NOUT,126)
DO 103 J=39,40
  JK=J-29
  IF (ICBK(BMM2,58,10,J,6).LE.0) GO TO 103
  WRITE (NOUT,122) NAME1(JK),(BMM2(J,K),K=6,10)
103 CONTINUE
WRITE (NOUT,127)
DO 104 J=41,49
  JK=J-40
  IF (ICBK(BMM2,58,10,J,6).LE.0) GO TO 104
  WRITE (NOUT,122) NAME1(JK),(BMM2(J,K),K=6,10)
104 CONTINUE
WRITE (NOUT,128)
DO 105 J=50,58
  JK=J-49
  IF (ICBK(BMM2,58,10,J,6).LE.0) GO TO 105
  WRITE (NOUT,122) NAME1(JK),(BMM2(J,K),K=6,10)
105 CONTINUE
106 CONTINUE
PRINT 107
FORMAT (1H1)
DO 108 J=1,NO
  PRINT 108, (BNRP(J,I),I=1,5)
108 CONTINUE
109 FORMAT (1X,4HBNRP,5F8.2)
DO 110 J=1,NO
  PRINT 111, (CPP(J,IZ),IZ=1,3),(GNP(J,IY),IY=1,3)
```


110 CONTINUE
111 FORMAT (1X,3HGPP,3F8.2,5X,3HGNP,3F12.0)
STOP

...FORMATS...

- 112 FORMAT (1H1,24X,16HDECISION NAME - ,A6,16X,5HPAGE ,I3,1H./25X,22(1H=)///)
- 113 FORMAT (21X,29HSUPPLY AND UTILIZATION REPORT/19X,33(1H-)/24X,23H(THOUSAND POUNDS, M.E.)//20X,6HQTR. I,5X,7HQTR. II,3X,8HQTR. III,4X,27HQTR. IV,5X,I4/1X,15(1H-),5(3X,8(1H-))///)
- 114 FORMAT (1X,11HMARKETINGS:/2X,7HGRADE A,7X,5F11.0/2X,7HGRADE B,7X,5F11.0//1X,8HIMPORTS:/2X,6HCHEESE,8X,5F11.0/2X,12HM. CLASS III,2X,5F11.0/2X,6HBUTTER,8X,5F11.0/2X,4HNFDM,10X,5F11.0//1X,15HEND COMM S3TOCK:/2X,6HCHEESE,8X,5F11.0/2X,6HBUTTER,8X,5F11.0/2X,4HNFDM,10X,5F411.0//1X,15HEND GOUT STOCK:/2X,6HCHEESE,8X,5F11.0/2X,6HBUTTER,8X,5F11.0/2X,4HNFDM,10X,5F11.0/2X,12HCONSUMPTION:/2X,7HCLASS I,7X,5F161.0/2X,8HCLASS II,6X,5F11.0/2X,6HCHEESE,8X,5F11.0/2X,12HM. CLASS I7II,2X,5F11.0/2X,6HBUTTER,8X,5F11.0/2X,4HNFDM,10X,5F11.0////1X,5HN8OTE:/3X,32HSUPPLY AND UTILIZATION WILL NOT ,32HBALANCE BECAUSE F9IGURES HAVE NOT/3X,17HBEEN ADJUSTED TO ,35HREFLECT JOINT PRODUCTIO*N OF BUTTER./)
- 115 FORMAT (15X,41HCOMPARATIVE SUPPLY AND UTILIZATION REPORT/13X,45(1H1-)/24X,23H(THOUSAND POUNDS, M.E.)//14X,5(7X,I4)/1X,16(1H-),5(3X,8(21H-))///)
- 116 FORMAT (28X,14HSUMMARY REPORT/26X,18(1H-)//20X,6HQTR. I,5X,7HQTR. II,3X,8HQTR. III,4X,7HQTR. IV,5X,I4/1X,15(1H-),5(3X,8(1H-))///)
- 117 FORMAT (1X,15HPRICES (\$/CWT):/2X,7HGRADE A,7X,5F11.2/2X,7HGRADE B,17X,5F11.2/2X,8HALL MILK,6X,5F11.2//1X,14HGROSS VALUE OF/1X,16HFARM 2 MKTGS (T\$):/2X,7HGRADE A,7X,5F11.0/2X,7HGRADE B,7X,5F11.0/2X,8HAL3L MILK,6X,5F11.0//1X,15HRTN OUR DIRECT/1X,13HCOST (\$/CWT):/2X,7H4RADE A,7X,5F11.2/2X,7HGRADE B,7X,5F11.2/2X,8HALL MILK,6X,5F11.2//15X,13HTRANSP. COST,/1X,14HMFD PROD. (T\$),1X,5F11.0//1X,16HCONS. EXP6. (T\$):/2X,11HFLUID PROD.,3X,5F11.0/2X,10HMFD. PROD.,4X,5F11.0/2X,79HALL PROD.,5X,5F11.0//1X,15HGOUT. EXP. (T\$),5F11.0//1X,15HGOUT. P8URCH. AS/1X,13HPCNT OF SALES,2X,5F11.1//1X,15HTOTAL PURCH. AS/1X,193HPCNT OF SALES,2X,5F11.1)
- 118 FORMAT (22X,26HCOMPARATIVE SUMMARY REPORT/20X,30(1H-)//14X,5(7X,I41)/1X,16(1H-),5(3X,8(1H-))///)
- 119 FORMAT (1H2,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,221(1H=)//17X,21HCOMPARATIVE REPORT - ,2A10,A5/15X,50(1H-)/31X,7HQUAR2TER, I2/31X,9(1H-)/28X,A10,A7//9X,6HREGION,2X,6(5X,I4)/7X,10(1H-),13X,6(3X,6H-----))
- 120 FORMAT (1H2,24X,15HDECISION NAME -,1X,A6,16X,5HPAGE ,I3,1H./25X,221(1H=)//17X,21HCOMPARATIVE REPORT - ,2A10,A5/15X,50(1H-)/33X,6HANNU2AL/33X,6(1H-)/28X,A10,A7//9X,6HREGION,2X,6(5X,I4)/7X,10(1H-),1X,6(33X,6H-----))
- 121 FORMAT (1H8)
- 122 FORMAT (2X,A10,5F12.0)
- 123 FORMAT (29X,18HMOVEMENTS REPORT/29X,16(1H=)//25X,5HQUARTER -,I2,8H1 YEAR -,I5/25X,24(1H-)//12X,5(2X,A10)/12X,5(2X,10(1H-)))
- 124 FORMAT (/36X,15HCLASS II DEMAND,/1X,12HMILK SUPPLY:)
- 125 FORMAT (/37X,13HCHEESE DEMAND,/1X,12HMILK SUPPLY:)
- 126 FORMAT (1X,8HIMPORTS:)
- 127 FORMAT (1X,13HCOMM. STOCKS:)
- 128 FORMAT (1X,13HGOUT. STOCKS:)
- 129 FORMAT (/31X,22HMISC. CLASS III DEMAND,/1X,12HMILK SUPPLY:)
- 130 FORMAT (/36X,13HBUTTER DEMAND,/1X,14HBUTTER SUPPLY:)
- 131 FORMAT (/31X,22HNONFAT DRY MILK DEMAND,/1X,12HNFDM SUPPLY:)

END
FUNCTION ICHK(A,I,J,K,L)

CHECK FOR ZERO MOVEMENTS

DIMENSION A(I,J)
L1=L+4
ICLK=0
DO 1 II=L,L1
IF (A(K,II).LE.0.0) GO TO 1
ICLK=1
RETURN
1 CONTINUE
RETURN

END
SUBROUTINE SCLE (IN,JN,AA,SCALE)

SCALE VARIABLES IN COMPARATIVE REPORTS

DIMENSION AA(30,9)
DO 1 I=1,IN
DO 1 J=1,JN
1 AA(J,I)=SCALE*AA(J,I)
RETURN

END
SUBROUTINE RITE (J,IM,IN,IO,NAME,AA,KW,NOUT)

WRITE COMPARATIVE REPORTS

DIMENSION NAME(16), AA(30,9)
DO 2 I=IM,IN
II=I
IF (IO.EQ.1) II=I+11
IF (KW.EQ.1) GO TO 1
WRITE (NOUT,3) NAME(II),AA(J,I),AA(J+5,I),AA(J+10,I),AA(J+15,I)
1 ,AA(J+20,I),AA(J+25,I)
GO TO 2
1 WRITE (NOUT,4) NAME(II),AA(J,I),AA(J+5,I),AA(J+10,I),AA(J+15,I)
1 ,AA(J+20,I),AA(J+25,I)
2 CONTINUE
3 FORMAT (7X,A10,1X,SF9.1)
4 FORMAT (7X,A10,1X,SF9.2)
RETURN

END

LIST OF MAJOR VARIABLES

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
A	(59,5)	DAMPSIN, DAMPSLV: Initially used to store retail fluid milk prices. Later used to store intercept values for Class I demand functions. (59 orders, 5 quarters)
	(75)	DRW1: Work array (75 regulated production areas).
AA	(59,59)	DRW1: Work array (59 orders)
	(30,9)	DRW2: Work array (30 quarters and years, 9 unregulated production regions)
ABLEND	(59)	DAMPCLC: Simple average annual blend prices for current year by individual order areas (59 orders).
ACEXP	(2)	DAMPSIN, DAMPCLC: Aggregate consumer expenditure for current and previous years.
ACHGC1S	(45)	DAMPCLC: Annual percent change in federal order Class I sales (45 federal orders).
ACHGC2D	(45)	DAMPCLC: Annual percent change in federal order Class II consumption (45 federal orders).
ACHGC3Q	(45)	DAMPCLC: Annual percent change in federal order Class III consumption (45 federal orders).
ACHGPRR	(45)	DAMPCLC: Annual percent change in federal order total producer receipts (45 federal orders).
ACHIACL	(45)	DAMPCLC: Annual percent change in federal order in-area Class I sales (45 federal orders).
ACL1P	(59)	DAMPCLC: Simple average annual Class I prices for federal and state orders (59 orders).
ACL1UT	(59)	DAMPCLC: Simple average annual Class I utilization in federal and state orders (59 orders).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
ACL2D	(59,2)	DAMPSIN, DAMPCLC: Annual Class II consumption in federal and state orders for current and previous years (59 orders, 2 years).
ACL3Q	(59,2)	DAMPSIN, DAMPCLC: Annual Class III consumption for federal and state orders for current and previous years. (59 orders, 2 years).
ACST		DAMPSIN, DAMPCLC: Raw milk assembly cost in federal orders.
ADFPST		DAMPSIN, DAMPCLC: Aggregate annual transportation and assembly charges for direct-ship producers in federal orders.
ADTCST		DAMPSIN, DAMPCLC: Aggregate annual handling charges from direct-ship producers in federal orders to other federal orders.
AFRPS	(59)	DAMPCLC: Simple average annual farm-retail fluid milk price spread in federal and state orders (59 orders).
AGRC1	(45)	DAMPCLC: Annual Class I sales in federal order processing centers (45 federal orders).
AIAC1S	(2)	DAMPSIN, DAMPCLC: Average aggregate blend price in federal orders for current and previous years (2 years).
AMBLND	(2)	DAMPSIN, DAMPCLC: Average aggregate blend price in federal orders for current and previous years (2 years).
AMC1P	(2)	DAMPSIN, DAMPCLC: Average aggregate Class I price in federal orders for current and previous years (2 years).
AMC1U	(2)	DAMPSIN, DAMPCLC: Average aggregate Class I utilization in federal orders for current and previous years (2 years).
AMC20		DAMPCLC: Aggregate annual Class II products obtained from non-order sources in federal orders.
AMC2P	(2)	DAMPSIN, DAMPCLC: Average aggregate Class II price for current and previous years (2 years).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
AMC3P	(2)	DAMPSIN, DAMPCLC: Average aggregate Class III price for current and previous years (2 years).
AMMFG	(61,28)	DAMPSIN, DAMPCLC: Annual individual movements from federal order production to manufacturing centers. (61 production areas, 28 manufacturing areas).
AMOCC		DAMPCLC: Aggregate annual movements from federal order processing centers to other than own consumption centers.
AMOPC		DAMPCLC: Aggregate annual movements from federal order production centers to other than own processing center.
AMPPC		DAMPCLC: Aggregate annual movements from federal order production centers to own processing centers.
AMRTL	(2)	DAMPSIN, DAMPCLC: Aggregate annual average retail price of fluid products in federal orders for current and previous years (2 years).
AMUCP	(2)	DAMPSIN, DAMPCLC: Aggregate average utilization of federal order processing capacity for current and previous years (2 years).
ANFP	(45)	DAMPCLC: Average annual net farm price in federal orders (45 orders).
AOACIT	(45)	DAMPCLC: Average annual packaged milk unit transportation cost in federal orders (45 orders).
AP		DAMPSIN: Intercept value for packaged milk transportation cost function.
APCL1	(59,2)	DAMPSIN, DAMPCLC: Class I receipts in federal and state orders for current and previous years (59 orders, 2 years).
APMM	(45,45)	DAMPSIN, DAMPCLC: Annual individual movements from federal order processing centers to consumption centers. Temporarily used to store other information in MAIN of DAMPSIN (45 orders).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
APPREC	(59,2)	DAMPSIN, DAMPCLC: Total producer receipts in federal and state orders for current and previous years (59 orders, 2 years).
AR		DAMPSIN, DAMPCLC: Intercept value for raw milk transportation cost function.
ARMD	(45,45)	DAMPSIN, DAMPCLC: Annual individual movements from federal order direct-ship production centers to processing centers (45 orders).
ARMMS	(16,45)	DAMPSIN, DAMPCLC: Annual individual movements from federal order supply plants to processing centers. Temporarily used in MAIN of DAMPSIN to store labels for base data (16 plants, 45 orders).
ARODC	(45)	DAMPCLC: Annual average return over direct cost in federal orders (45 orders).
ARPR	(59)	DAMPCLC: Annual average retail price of fluid products in federal and state orders (59 orders).
ASCL2D	(2)	DAMPSIN, DAMPCLC: Aggregate federal order Class II sales net of sales from nonorder sources for current and previous years (2 years).
ASCL3Q	(2)	DAMPSIN, DAMPCLC: Aggregate federal order Class III sales for the current and previous years (2 years).
ASFPCST		DAMPSIN, DAMPCLC: Aggregate annual transportation and assembly charges for supply plant producers.
ASFRPS	(2)	DAMPSIN, DAMPCLC: Aggregate annual summation of the farm-retail fluid milk price spread multiplied by area Class I consumption in each federal order market for current and previous years (2 years).
ASGRCL		DAMPCLC: Aggregate annual total producer receipts.
ASIACL	(2)	DAMPSIN, DAMPCLC: Aggregate federal order Class I consumption for current and previous years (2 years).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
ASMDS		DAMPCLC: Aggregate annual movements from the direct-ship federal order production centers to manufacturing centers.
ASMDSD		DAMPCLC: Aggregate annual movements from direct-ship federal order production centers to the manufacturing center dummies.
ASMFGU	(2)	DAMPSIN, DAMPCLC: Aggregate annual utilization of federal order manufacturing capacity for current and previous years (2 years).
ASMPG		DAMPCLC: Aggregate annual sales by federal order processors in their own consumption area.
ASMSP		DAMPCLC: Aggregate annual movements from supply plants to manufacturing centers.
ASMSPD		DAMPCLC: Aggregate annual movements from supply plants to manufacturing center dummies.
ASNFP	(2)	DAMPSIN, DAMPCLC: Aggregate average net farm price in federal orders for current and previous years (2 years).
ASOAC1D		DAMPCLC: Annual aggregate average unit packaged milk transportation cost in federal orders.
ASOAC1T	(2)	DAMPSIN, DAMPCLC: Aggregate total packaged milk transportation cost in federal orders for current and previous years (2 years).
ASOPC		DAMPCLC: Aggregate annual value of supply plant shipments to own area processors.
ASPCL1	(2)	DAMPSIN, DAMPCLC: Aggregate producer receipts used in Class I in federal orders for current and previous years (2 years).
ASPRRC	(2)	DAMPSIN, DAMPCLC: Aggregate total producer receipts in federal orders for current and previous years (2 years).
ASRODC	(2)	DAMPSIN, DAMPCLC: Aggregate average return over direct cost in federal orders for current and previous years (2 years).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
ASTCCWT		DAMPCLC: Aggregate average total per unit cost of Class I sales by processing center in federal orders for current and previous years (2 years).
ASTRMC		DAMPCLC: Aggregate annual average raw milk acquisition cost per unit of Class I sales in federal orders.
ASTRNSD		DAMPCLC: Aggregate annual average raw milk transportation cost per unit of Class I sales in federal orders.
ASTRNSR	(2)	DAMPSIN, DAMPCLC: Aggregate annual handling charges for supply plant milk shipped to other orders for the current year and the previous years (2 years).
ATCCWT	(59)	DAMPCLC: Average annual total per unit cost of Class I sales by individual federal and state order processors (59 orders).
ATMCST		DAMPSIN, DAMPCLC: Aggregate annual handling charges for milk shipped to federal order manufacturing plants outside of own orders.
ATMFGM		DAMPCLC: Aggregate annual movements from federal order production centers to manufacturing centers.
ATMFGU		DAMPCLC: Aggregate annual utilization of manufacturing capacity in federal orders.
ATMMC	(53)	DAMPCLC: Total milk received annually at each federal order manufacturing center (53 centers)
ATMOP		DAMPCLC: Total annual movements from supply plants to other area processing centers.
ATRANSR	(45)	DAMPCLC: Average annual raw milk transportation cost per unit of Class I sales in federal orders (45 orders).
ATRMC	(45)	DAMPCLC: Average annual raw milk acquisition cost per unit of Class I sales in federal orders (45 orders).
ATRMPC		DAMPSIN, DAMPCLC: Aggregate annual raw milk production costs in federal orders.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
AUMCP	(53)	DAMPCLC: Annual average utilization of individual manufacturing center capacity in federal orders (53 centers).
AUNC1S	(6,9)	DRW1: Annual sales of Class I products in unregulated Grade A regions (6 years, 9 regions).
AUNGBP	(6,9)	DRW1: Annual gross farm price of milk in unregulated Grade A regions (6 years, 9 regions).
AUNNBP	(6,9)	DRW1: Annual net farm price of milk in unregulated Grade A regions (6 years, 9 regions).
AUNREG	(6,9)	DRW1: Annual raw milk production in unregulated Grade A regions (6 years, 9 regions).
AUNRODC	(6,9)	DRW1: Annual returns over direct cost to farmers in unregulated Grade A regions (6 years, 9 regions).
AUTCP	(45)	DAMPCLC: Annual average utilization of Class I processing capacity in federal orders (45 orders).
AVP	(2)	DAMPSIN, DAMPCLC: Aggregate annual pool values in federal orders for current and previous years (2 years).
AWFRPS	(2)	DAMPSIN, DAMPCLC: Average aggregate annual farm-retail fluid milk price spread in federal orders for current and previous years (2 years).
B	(9,4)	DAMPSIN, DAMPSLV: Base Grade B milk production (9 regions, 4 quarters).
	(59)	DRW1: Work array (59 orders).
BB	(53)	DRW1: Work array (53 manufacturing areas).
	(30,5)	DRW2: Work array (30 quarters and years, 5 consumption regions)
BBC3P	(5)	DAMPSIN, DAMPSLV: Equals BC3P.
BC	(30,5,5)	DRW2: Manufactured milk product consumption (30 quarters, and years, 5 products, 5 regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
BCEXP	(5)	DAMPSIN, DAMPCLC: Aggregate consumer expenditure on fluid products in federal orders by quarter for current year.
BCHGI	(4)	DAMPSIN, DAMPSLV: Exogenous change in base import levels (4 products).
BC2P	(5)	DAMPSIN, DAMPSLV: Used to read in quarterly base Class II prices which are converted to retail level prices.
BC3P	(5)	DAMPSIN: Used to read in quarterly base Class III prices.
BCRK		DRW2: Total quarterly consumption of individual manufactured product; used to calculate weighted average retail prices.
BCRT		DRW2: Total annual consumption of individual manufactured product; used to calculate weighted average retail prices.
BDE	(5)	DAMPSIN, DAMPSLV: Regional demand elasticities for manufactured milk products (5 consumption regions).
BFPHC		DAMPSIN, DRW2: Farm to plant hauling costs for Grade B milk producers.
BIACL	(59,4)	DAMPSIN, DAMPSLV: Quarterly federal and state order sales bases. Producer receipts used in Class I or in-area Class I sales (59 orders, 4 quarters).
BIMP	(4,2,4)	DAMPSIN, DAMPSLV: Base imports (4 products, 2 regions, 4 quarters).
	(30,4,2)	DRW2: Imports (30 quarters and years, 4 products, 2 regions).
BLAMR	(20)	DAMPSLV, DAMPCLC: 1-10 contains average annual blend prices for consolidated orders; 11-20 average annual Class I utilization for consolidated orders.
BLDIF	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Blend price differentials for individual orders in consolidated orders (59 orders).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
BLEND	(59,6)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly blend prices for base, current, and previous quarters (59 orders, 6 quarters).
	(30)	DRW2: Average blend prices for all orders (30 quarters and years).
BLNDRM	(20)	DAMPSIN, DAMPCLC: 1-10 contains quarterly blend prices for consolidated orders; 11-20 contains quarterly Class I utilization for consolidated orders.
BLWT		DAMPSIN, DRW1: Location differential used to generate blend price differentials from distances.
BMBLEND	(5)	DAMPSIN, DAMPCLC: Quarterly average federal order blend price for current year (5 quarters).
BMCLLP	(5)	DAMPSIN, DAMPCLC: Quarterly average order Class I price for current year (5 quarters).
BMCLLU	(5)	DAMPSIN, DAMPCLC: Aggregate average order Class I utilization by quarter for current year.
BMKP	(5)	DAMPSIN, DAMPSLV, DRW2: Retail markup used in retail price formula for manufactured products (5 products).
BMD1	(31,15)	DAMPSLV: Transportation costs, stage 2.
	(58,10)	DAMPSLV: Transportation costs, stage 3.
BMM1	(31,15)	DRW2: Transportation cost work array, stage 2; same as BMD1.
BMM2	(58,10)	DRW2: Transportation cost work array, stage 3; same as BMD2.
BMRTL	(5)	DAMPSIN, DAMPCLC: Quarterly average order retail prices of fluid milk for current year (5 quarters).
BNCL2P	(5)	DAMPSIN, DAMPCLC: Quarterly Class II prices for current year (5 quarters).
BNCL3P	(5)	DAMPSIN, DAMPCLC: Quarterly Class III prices for current year (5 quarters).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
BNF	(2)	DAMPSIN, DAMPSLV: Contains retail price function weights on the farm price of milk used to jointly produce butter and nonfat dry milk (2 products).
BNINV	(2,9)	DAMPSIN, DAMPSLV: Ending commercial stocks of butter and nonfat dry milk by regions. (2 products, 9 regions).
BNRP	(30,5)	DRW2: Retail prices for manufactured products. (30 quarters and years, 5 products).
BOPC		DAMPSIN, DRW2: Miscellaneous charges to Grade B milk producers.
BP	(20)	DAMPSIN: Up to 20 quarters of Class I base prices for federal orders.
BPC	(9)	DAMPSIN, DAMPSLV, DRW2: Direct cost of producing Grade B milk (9 regions).
BPC1		DRW2: BPC adjusted for exogenous changes in costs; BPC only equals base costs in DRW2.
BPMC		DAMPSIN, DRW2: Grade B milk producer marketing charges.
BPOP	(31)	DAMPSLV: Current quarter movements from supply nodes in stage 2 (31 supply nodes).
BPRR	(15)	DAMPSLV: Current quarter movements to demand nodes in stage 2 (15 demand nodes).
BPS	(20)	DAMPSIN: Contains up to 20 quarters of Class I base prices for state orders (20 quarters).
BP1		DAMPSIN: Slope coefficient for packaged milk transportation cost function.
BR		DAMPSIN: Slope coefficient for raw milk transportation cost function.
BRCST	(5)	DAMPSIN, DAMPSLV: Wholesale margin used in retail price functions for manufactured products (5 products).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
BS	(6,9,4)	DAMPSIN, DAMPSLV: Ending commercial and government stocks (6 stocks, (3 products-2 types), 9 regions, 4 quarters).
	(30,6,9)	DRW2: Ending commercial and government stocks. (30 quarters and years, 6 stocks, 9 regions).
BSE	(9)	DAMPSIN, DAMPSLV: Grade B milk price elasticities of supply (9 regions).
BSMFGU	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate utilization of order manufacturing capacity for current year (5 quarters).
BSMUCP	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate utilization of order Class I processing capacity for current year (5 quarters).
BSNFP	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate average net farm price in federal orders for current year (5 quarters).
BSOACT	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate total packaged milk transportation costs in orders for current year (5 quarters).
BSTRNSR	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate total raw milk transportation costs in orders for current year (5 quarters).
BSVFRPS	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate value of the summation of the product of farm-retail fluid milk price spread and Class I consumption in orders for current years (5 quarters).
BSVP	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate order pool values for current year (5 quarters).
BSVRODC	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate average return over direct cost in orders for current year (5 quarters).
BTCST	(2,5)	DAMPSIN, DAMPSLV: Manufactured milk product transportation cost parameters (intercept and slope, 5 products).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
BTPSQ	(59,4)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly total producer receipts in federal and state orders for base year (59 orders, 4 quarters).
BTRANS	(11,5)	DAMPSIN, DAMPSLV: Mileages between supply and demand centers in stages two and three (11 supply centers, 5 demand centers).
BWFRPS	(5)	DAMPSIN, DAMPCLC: Quarterly aggregate average farm-retail fluid milk price spread in orders for current year (5 quarters).
C	(5,5,4)	DAMPSIN, DAMPSLV: Manufactured milk product consumption in base year, (5 products, 5 regions, 4 quarters).
	(59)	DRW1: Work array (59 orders).
CC	(53)	DRW1: Work array (53 manufacturing centers).
CCEXP		DAMPCLC: Percent change in consumer expenditure on fluid products in federal orders from same quarter a year ago.
CCHK		DRW2: Used to determine which procedure is to be used to add commercial stocks together.
CCL2P		DAMPCLC: Percent change in Class II price from same quarter a year ago.
CCL3P		DAMPCLC: Percent change in Class III price from same quarter a year ago.
CEXP		DAMPCLC: Current quarter aggregate consumer expenditure on fluid products in federal orders.
	(30)	DRW2: Consumer expenditure for fluid products (30 quarters and years).
CEXPAM	(30)	DRW2: Consumer expenditure on all milk products (30 quarters and years).
CEXPB	(30)	DRW2: Consumer expenditure on manufactured milk products (30 quarters and years).
CEX1		DAMPSLV: Current quarter aggregate consumer expenditure on fluid products in federal and state orders.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CF	(5)	DAMPSIN, DAMPSLV, DRW2: Conversion factors for manufactured milk products in pounds of product per 100 pounds of milk (5 products).
CHABLND		DAMPCLC: Percent change in annual average federal order blend price from a year ago.
CHAEXP		DAMPCLC: Annual percent change in aggregate federal order consumer expenditure on fluid products from a year ago.
CHAFRPS		DAMPCLC: Annual percent change in ASFRPS from a year ago.
CHAMC1P		DAMPCLC: Annual percent change in AMC1P from a year ago.
CHAMC2P		DAMPCLC: Annual percent change in AMC2P from a year ago.
CHAMC3P		DAMPCLC: Annual percent change in AMC3P from a year ago.
CHAMRTL		DAMPCLC: Annual percent change in AMRTL from a year ago.
CHASC1		DAMPCLC: Annual percent change in ASIAC1 from a year ago.
CHASC1U		DAMPCLC: Annual percent change in AMC1U from a year ago.
CHASC2		DAMPCLC: Annual percent change in ASCL2D from a year ago.
CHASC3		DAMPCLC: Annual percent change in ASCL3Q from a year ago.
CHASFGU		DAMPCLC: Annual percent change in ASMFGU from a year ago.
CHASNFP		DAMPCLC: Annual percent change in ASNFP from a year ago.
CHASOA		DAMPCLC: Annual percent change in ASOACT from a year ago.
CHASPC		DAMPCLC: Annual percent change in ASPCL1 from a year ago.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CHASPR		DAMPCLC: Annual percent change in ASPRRC from a year ago.
CHASRDC		DAMPCLC: Annual percent change in ASRODC from a year ago.
CHASTRN		DAMPCLC: Annual percent change in ASTRNSR from a year ago.
CHAUCP		DAMPCLC: Annual percent change in AMUCP from a year ago.
CHAVP		DAMPCLC: Annual percent change in AVP from a year ago.
CHAWRPS		DAMPCLC: Annual percent change in AWRFRPS from a year ago.
CHEESE	(9)	DAMPSLV: Current quarter cheese sales by each production region (9 regions).
CHGBP		DAMPSIN: Quarterly change in federal order Class I base price.
CHGBPS		DAMPSIN: Quarterly change in state order Class I base price.
CHGBT	(5)	DAMPSIN, DAMPSLV: Annual index for exogenous change in raw milk transportation costs.
CHGCL1P	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly change in federal and state order Class I prices. (59 orders).
CHGCL1S	(45)	DAMPCLC: Percent change in federal order producer receipts used in Class I from same quarter a year ago (45 orders).
CHGCL2D	(45)	DAMPCLC: Percent change in federal order producer receipts used in Class II same quarter a year ago. (45 orders).
CHGCL2P		DAMPSIN, DAMPSLV, DAMPCLC: Quarterly change in Class II price, as read from user input form.
CHGCL3P		DAMPSIN, DAMPSLV, DAMPCLC: Quarterly change in Class III price, as read from user input form.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CHGCL3Q	(45)	DAMPCLC: Percent change in federal order producer receipts used in Class III from same quarter a year ago (45 orders).
CHGHC	(5)	DAMPSIN, DAMPSLV, DAMPCLC, DRW2: Annual index for exogenous change in raw milk hauling cost (5 years).
CHGIAC1	(45)	DAMPCLC: Percent change in federal order Class I consumption from same quarter a year ago (45 orders).
CHGMC	(5,5)	DAMPSIN, DAMPSLV: Exogenous percent change in consumption of manufactured products by region (5 products, 5 regions).
CHGNC1Q	(45)	DAMPSIN, DAMPSLV: Quarterly exogenous percentage change in federal order Class I consumption (45 orders).
CHGCN2Q	(45)	DAMPSIN, DAMPSLV: Quarterly exogenous percentage change in federal order Class II consumption (45 orders).
CHGPD	(20)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly exogenous change index for raw milk production costs (20 quarters).
	(30)	DRW2: 1-5 equals 1, 6-9 equals 1-4 in CHGPD of DAMPSIN, 10 equals average of 6-9, 11-14 equals 5-8 in CHGPD of DAMPSIN, 15 equals average of 11-14, and so on up to 26-29 equals 17-20 in CHGPD of DAMPSIN, 30 equals average of 26-29 (30 years and quarters).
CHGPRC	(5)	DAMPSIN, DAMPSLV, DAMPCLC: Annual exogenous change index for Class I processing costs (5 years).
CHGPRR	(45)	DAMPCLC: Percent change in total federal order producer receipts from same quarter a year ago (45 orders).
CHGPT	(5)	DAMPSIN: Annual index for exogenous change in packaged milk unit transportation cost (5 years).
CHI		DAMPSLV: Amount of butter produced as whey cream by-product from production of one pound of cheese.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CIAC1		DAMPCLC: Aggregate percent change in federal order Class I consumption from same quarter a year ago.
CINSP	(59)	DAMPSIN, DAMPSLV: Inspection costs for federal and state orders (59 orders).
CINV	(9,3)	DAMPSLV: Inventory demand for commercial cheese stocks based on movements from raw milk production, beginning commercial stocks, and beginning government stocks nodes (9 regions, 3 node types).
CL1CP	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Federal and state order Class I processing capacity (59 orders).
CL1P	(59,5)	DAMPSIN, DAMPSLV, DAMPCLC: Initially used to store base year Class I prices in DAMPSIN. Later used to store proposed Class I prices from primary input form (59 orders, 5 quarters).
CL1Q	(59,4)	DAMPSIN, DAMPCLC: Quarterly producer receipts used in Class I for current year (59 orders, 4 quarters).
CL1RR	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Percentage federal and state order Class I required reserve (59 orders).
CL1SLS	(59)	DAMPSLV: Actual, optimal network flow through federal and state order processing centers in current year (59 orders).
CL1SP	(16)	DAMPSIN, DAMPSLV, DAMPCLC: Class I price for supply plants by market (16 plants).
CL1UT	(59)	DAMPCLC: Current quarter federal and state order percent Class I utilization (59 orders).
CL2		DAMPSIN: Quarterly percentage change in Class II price.
CL2Q	(59,4)	DAMPSIN, DAMPSLV: Quarterly base federal and state order producer receipts used in Class II (59 orders).
CL3		DAMPSIN: Quarterly percentage change in Class III price.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CL3DS	(59)	DAMPSLV: Current quarter shipments from federal and state order direct-ship production centers to manufacturing centers (59 orders).
CL3P	(30)	DRW2: Quarterly Class III price (30 quarters and years).
CL3PN	(30)	DRW2: Quarterly net Grade B milk price (30 quarters and years).
CL3SP	(16)	DAMPSLV: Current quarter milk shipments from supply plants to manufacturing centers (16 plants).
CMBLEND		DAMPCLC: Percent change in MBLEND from same quarter a year ago.
CMCL1P		DAMPCLC: Percent change in BMCL1P from same quarter a year ago.
CMCL1U		DAMPCLC: Percent change in BMCL1U from same quarter a year ago.
CMCL2		DAMPCLC: Percent change in aggregate federal order Class II consumption from same quarter a year ago.
CMCL3		DAMPCLC: Percent change in aggregate federal order Class III consumption from same quarter a year ago.
CMRLTP		DAMPCLC: Percent change in aggregate federal order retail price of fluid products from same quarter a year ago.
CMP	(2,2)	DRW1: Labels for federal and state order Comparative Reports.
CMUCP		DAMPCLC: Percent change in SMUCP from same quarter a year ago.
CNPB		DRW2: Quarterly net purchases of butter for commercial stocks.
CNPME		DRW2: Quarterly net purchases of all products for commercial stocks, measured in milk equivalent.
CNPN		DRW2: Quarterly net purchases of nonfat dry milk for commercial stocks.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CPRCHG	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Change in federal and state order direct-ship producer receipts as a percentage of base direct-ship receipts (59 orders).
CQB	(9)	DAMPSLV: Current quarter production of Grade B milk by region (9 regions).
CQBNS	(2,9)	DAMPSLV: Current quarter production of butter and nonfat dry milk by regions (2 products, 9 regions).
CQBS	(2,3,3)	DAMPSLV: Current quarter beginning commercial and government stocks of manufactured products by region (commercial or government, 3 products, 9 regions).
CQC	(5,5)	DAMPSLV: Current quarter consumption of manufactured products by region (5 products, 5 regions).
CQC1D	(59)	DAMPSLV, DAMPCLC: Current quarter federal and state order Class I consumption (59 orders).
CQC2D	(59)	DAMPSLV: Current quarter federal and state order Class II consumption (59 orders).
CQES	(6,9)	DAMPSLV: Current quarter ending commercial and government stocks of manufactured products by region (6 stocks, 9 regions).
CQIMP	(4,2)	DAMPSLV: Current quarter imports of manufactured products by region (4 products, 2 regions).
CQMFG	(9)	DAMPSLV: Milk available for manufacturing in the current quarter by region (9 regions).
CQTSP	(59)	DAMPSLV, DAMPCLC: Current quarter federal and state order raw milk production (59 orders).
CQUCEXP		DAMPSLV: Current quarter consumer expenditure on fluid products in unregulated Grade A milk regions.
CQUCLP	(9)	DAMPSLV: Current quarter "Class I" prices in unregulated Grade A milk regions (9 regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CQUC1S	(9)	DAMPSLV: Current quarter sales of fluid products in unregulated Grade A milk regions (9 regions).
CQUGBP	(9)	DAMPSLV: Current quarter gross farm prices of unregulated Grade A milk (9 regions).
CQUNBP	(9)	DAMPSLV: Current quarter net farm prices of unregulated Grade A milk (9 regions).
CQUN P	(9)	DAMPSLV: Current quarter retail prices of fluid products in unregulated Grade A milk regions (9 regions).
CQUREG	(9)	DAMPSLV: Current quarter milk production in unregulated Grade A milk regions (9 regions).
CQURODC	(9)	DAMPSLV: Current quarter returns over direct cost of producing milk in unregulated Grade A milk regions (9 regions).
CQUVP	(9)	DAMPSLV: Current quarter gross producer returns in unregulated Grade A milk regions (9 regions).
CR		DAMPSIN, DRW1: Used to determine which input forms are to be read.
CSMFGU		DAMPCLC: Percent change in SMFGU from same quarter a year ago.
CSNFP		DAMPCLC: Percent change in SNFP from same quarter a year ago.
CSOACL1T		DAMPCLC: Percent change in SOACL1T from same quarter a year ago.
CSPCL1		DAMPCLC: Percent change in SPCL1 from same quarter a year ago.
CSPREC		DAMPCLC: Percent change in SPRREC from same quarter a year ago.
CSTRNSR		DAMPCLC: Percent change in STRNSR from same quarter a year ago.
CSVFRPS		DAMPCLC: Percent change in SVFRPS from same quarter a year ago.
CSV P		DAMPCLC: Percent change in SVP from same quarter a year ago.
CSVRODC		DAMPCLC: Percent change in SVRODC from same quarter a year ago.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
CWFRPS		DAMPCLC: Percent change in WFRPS from same quarter a year ago.
C2MGN		DAMPSIN, DAMPSLV, DAMPCLC: Class II retail margin.
D	(59)	DRW1: Work array (59 orders).
DCHG1		DAMPSIN: Extra charge on direct-ship raw milk moving over DIST miles.
DE	(59)	DAMPSIN, DAMPSLV: Fluid milk product demand elasticities in all orders (59 orders).
DFPCST		DAMPSIN, DAMPCLC: Aggregate quarterly transportation and assembly charges for direct-ship producers.
DGSC	(20)	DAMPSIN, DAMPSLV, DRW1: Desired government stocks of cheese (20 quarters).
DIFF		DAMPSIN: Coefficient used to generate federal order Class I prices from a base price.
DIFFS		DAMPSIN: Coefficient used to generate state order Class I prices from a base price.
DIST		DAMPSIN: Used to pass distances to GCL1P for generating federal order Class I prices from a base price.
DISTB	(75)	DAMPSIN: Maximum shipping distances for raw milk (75 direct-ship and supply plant areas).
DISTD		DAMPSIN: Maximum distance direct-ship milk can move without incurring extra charge DCHG1.
DISTP	(59)	DAMPSIN: Maximum shipping distances for packaged milk by order areas (59 orders).
DISTS		DAMPSIN: Used to pass distances to GCL1P for generating state order Class I prices from a base price.
DMMFG	(27,10)	DAMPSIN, DAMPSLV: Initially contains distances from production centers to multiple manufacturing centers, in DAMPSIN. Then used to store unit transportation costs (27 manufacturing areas, up to 10 production areas).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
DNUM		DRW1, DRW2: Decision or experiment name, same as TITLE.
DS	(4,9)	DAMPSLV: Desired quarterly government stocks of cheese; see DGSC (4 quarters, 9 regions).
DTCST		DAMPSIN, DAMPCLC: Aggregate quarterly handling charges for direct-ship producers to other orders.
E	(59)	DRW1: Work array (59 orders).
ECHBT	(5)	DAMPSIN, DAMPSLV: Annual index for exogenous change in raw milk transportation costs (see CHGBT (5 years)).
ECHG		DAMPSIN, DAMPSLV: Extra per unit charge on milk moving to single manufacturing dummy.
EFPR	(59)	DAMPSLV: Effective reserve for federal and state order processing plants for the current quarter (45 orders).
EFRR1		DAMPSLV: Effective reserve for federal and state order processing plants.
ERCST	(75)	DAMPSIN, DAMPSLV, DAMPCLC: Initially contains assembly plus end-of-route cost. Later used to store end-of-route cost only. (75 production areas).
F	(59)	DRW1: Work array (59 orders).
FL1		DAMPSLV: Amount of butter produced from cream as by-product from production of one pound of fluid milk products.
FRPS	(59)	DAMPCLC: Current quarter federal and state order farm-retail fluid milk price spread (59 orders).
G	(59)	DRW1: Work array (59 orders).
GAC	(30,9)	DRW2: Fluid product consumption in all sectors (30 quarters and years, 9 regions).
GAP	(30,9)	DRW2: Production of Grade A milk in all sectors (30 quarters and years, 9 regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
GBP	(30,9)	DRW2: Production of Grade B milk (30 quarters and years, 9 regions).
GCHK		DRW2: Used to determine which procedure is to be used to add government stocks together.
GEXP	(30)	DRW2: Net government expenditures on dairy products (30 quarters and years).
GMP	(30,9)	DRW2: Milk available for manufacturing from all sectors (30 quarters and years, 9 regions).
GNP	(30,3)	DRW2: Government net purchases of cheese, butter and nonfat dry milk (30 quarters and years, 3 products).
GNPB		DRW2: Quarterly net purchases of butter for government stocks.
GNPME		DRW2: Quarterly net purchases of all products for government stocks, measured in milk equivalent.
GNPN		DRW2: Quarterly net purchases of nonfat dry milk for government stocks.
GPP	(30,3)	DRW2: Government purchase prices for manufactured product, equals retail price less retail markup, i.e., a wholesale price (30 quarters and years, 3 products).
GPTP	(30)	DRW2: Net government purchases as a percent of total production (30 quarters and years).
GRBCNV	(59)	DAMPSIN, DAMPSLV: Annual quantities of Grade B milk which are added to orders using a quarterly index, if exogenous factors vary (59 orders).
GRBIND	(4)	DAMPSIN, DAMPSLV: Quarterly Grade B conversion indices (4 quarters).
GRC1	(59)	DAMPSLV, DAMPCLC: Current quarter federal and state order packaged milk sales (59 orders).
H	(59)	DRW1: Work array (59 orders).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
HCHG1	(16)	DAMPSIN, DAMPSLV: Intra-market supply plant handling charge (16 plants).
HCHG2	(16)	DAMPSIN, DAMPSLV: Inter-market supply plant handling charge (16 plants).
IACL1	(59,4)	DAMPSIN, DAMPCLC: Quarterly federal and state order Class I consumption for current year (59 orders, 4 quarters).
IACL1S	(59)	DAMPSLV, DAMPCLC: Current quarter federal and state order Class I consumption for current year (59 orders, 4 quarters).
IBTEF		DAMPSIN: Annual index for exogenous changes in unit raw milk transportation costs.
ICMEF		DAMPSLV: Equal IC2EF.
ICR	(10)	DAMPSIN, DAMPCLC, DRW1: Used to specify which federal order comparative reports are to be printed (10 reports).
ICRB	(14)	DAMPSIN, DRW1, DRW2: Used to specify which manufacturing market comparative reports are to be printed (14 reports).
ICRS	(10)	DAMPSIN, DRW1: Used to specify which state order comparative reports are to be printed (10 reports).
ICST	(5,10)	DAMPSIN, DAMPSLV: Array containing node numbers for areas able to ship at zero inspection cost to areas having inspection costs.
IC1EF		DAMPSIN, DAMPSLV: If equal to 1, then Class I exogenous factors vary.
IC2EF		DAMPSIN, DAMPSLV: If equal to 1, then Class II exogenous factors vary.
IDICST	(5)	DAMPSIN: Node numbers of areas with inspection costs (5 areas).
IDMG1	(26)	DAMPSIN, DAMPSLV, DAMPCLC: Node numbers of federal order production centers linked to single manufacturing centers (26 centers).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
IDMG2	(19)	DAMPSIN, DAMPSLV, DAMPCLC: Node numbers of federal order production centers linked to multiple manufacturing centers (19 production centers).
IDMS		DAMPSLV: Accumulates total demand in networks two and three.
IDXB	(9,12)	DAMPSIN, DAMPSLV: Index numbers to aggregate regulated Grade A milk areas by unregulated regions (9 regions, up to 12 areas per region).
IGBEF		DAMPSIN, DAMPSLV: If IGBEF equals 1, Grade B milk converts to Grade A and is added to Grade A milk production on a quarterly basis.
IHCEF		DAMPSIN, DAMPCLC: If IHCEF equals 1, ERCST varies exogenously by CHGHC each quarter.
II	(59)	DRW1: Work array (59 orders).
III	(59)	DRW1: Work array (59 orders).
ILI	(45)	DAMPSIN, DAMPSLV: Used as a row incrementer for ICST (45 federal orders).
IMFGB		DAMPSIN, DAMPSLV, DRW1: If not equal to 0, unregulated regions are included in the model.
IMPHLD		DAMPSIN, DAMPSLV: If equal to 1, imports vary exogenously each year according to PCHGI, if equal to 0 imports vary only initially by PCHGI.
IMR		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: The value for the number of proposed mergers.
IMRB		DAMPSIN: The value for the option used for defining blend price differentials.
IMRG	(59)	DAMPSIN, DAMPSLV, DRW1: Federal and state order numbers for orders in proposed mergers (59 orders).
IMRN	(11)	DAMPSIN, DAMPSLV, DRW1: Numbers of orders in each of the consolidations (11 consolidations).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
IMRT		DAMPSIN, DRW1: The value for the total number of orders being consolidated.
IPAG		DRW1, DRW2: Page number.
IPCEF		DAMPSIN, DAMPSLV, DAMPCLC: If IPCEF equals 1, market processing costs vary exogenously by CHGPRC per quarter.
IPDEF		DAMPSIN, DAMPSLV, DAMPCLC, DRW2: If IPDEF equals 1, direct milk production costs vary exogenously by CHGPD per quarter.
IPNARC		DAMPSIN, DAMPSLV: If IPNARC equals 1, number of arcs in each sector of each network is printed in quarter 1.
IPPM	(45)	DAMPSIN, DAMPSLV: Initial percentage of sales federal order processor must make in own area. Used if in-area Class I sales are used as a sales base (45 federal orders).
IPRI	(20)	DAMPSIN, DAMPSLV: Quarterly print options for displaying arc input, if equal to 1, arc input is printed (20 quarters).
IPRO	(20)	DAMPSIN, DAMPSLV: Quarterly print options for displaying arc output, if equal to 1, arc output is printed (20 quarters).
IPRTB		DAMPSIN: If IPRTB equals 1, model base data is printed.
IPTEF		DAMPSIN: If IPTEF equals 1, unit packaged milk transportation costs vary exogenously by CHGPT per quarter.
IP1		DAMPSIN: Tape 8, first (federal order) base data file. DAMPSLV: Tape 10, input file from DAMPSIN (OP5). DAMPCLC: Tape 9, input file from DAMPSIN (OP2).
IP2		DAMPSIN: Tape 5, input file from federal order input forms (on cards).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
IP2		DAMPSLV: Tape 5, input file (on cards). DAMPCLC: Tape 13, input file from DAMPSLV (OP6).
IP3		DAMPSIN: Tape 10, second (state and Grade B) base data file.
IP4		DAMPSIN: Tape 5, input file from state order and Grade B input forms (on cards).
IR	(61)	DAMPSIN, DAMPCLC, DRW1: Node (index) numbers for each federal order production area (61 production areas).
IRR	(59)	DAMPCLC: Node (index) numbers for each direct-ship production area (59 federal and state orders).
ISCL1D		DAMPSLV: Aggregate Class I consumption plus reserve requirement for current quarter.
ISCL1CP		DAMPSIN, DAMPSLV: Aggregate fluid milk processing capacity for current quarter.
ISS		DAMPSIN, DAMPSLV, DRW1: Government stocks release option.
ISPS		DAMPSLV: Accumulates total supply in networks two and three.
ISTATE		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: State order option (see input form).
ISTPS		DAMPSLV: Aggregate raw milk supply for current quarter.
ISUP1-ISUP9		DAMPSIN, DAMPCLC, DRW1, DRW2: Contains user input form options 10a-f and 12a-c.
IWORK1-IWORK13		DAMPSIN, DAMPSLV: The number of elements in common blocks and WORK arrays, see internal documentation at beginning of programs for details.
IXT	(3)	DRW1: User input form options 13a-c.
I12A-I12D		DAMPSIN: Input form options 12a-d.
I13A-I13C		DAMPSIN: Input form options 13a-c.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
JPLT	(10)	DRW1: CDC plot routine option.
KGNET		DAMPSLV: If equal to 1, it implies stage 2; if equal to 2, it implies stage 3.
LBCHK		DAMPSIN, DAMPSLV: If LBCHK equals 1, total raw milk shipping requirements to own area are limited to own area Class I processing capacity.
LMMFG	(27,10)	DAMPSIN, DAMPSLV: Node numbers of federal order production areas linked to multiple manufacturing areas (27 multiple manufacturing areas, up to 10 production areas).
MBLEND		DAMPCLC: Current quarter aggregate average federal order blend price.
MC		DAMPSIN, DAMPSLV, DRW2: Number of manufactured product consumption regions.
MCL1P		DAMPCLC: Current quarter aggregate average federal order Class I price.
MCL1Q	(5)	DAMPSIN, DAMPCLC: Current quarter aggregate Class I receipts (5 quarters).
MCL1U		DAMPCLC: Current quarter aggregate average utilization of Class I capacity in federal orders.
MCL2Q	(5)	DAMPSIN, DAMPCLC: Current quarter aggregate Class II receipts (5 quarters).
MCL3Q	(5)	DAMPSIN, DAMPCLC: Current quarter aggregate Class III receipts (5 quarters).
MCR		DAMPSIN, DAMPSLV: If MCR equals 1, individual manufacturing center capacity is unlimited.
MFG	(33)	DRW1: Same as IDMG2.
MFID	(70)	DRW1: Manufacturing center numbers (70 centers).
MFN	(14)	DRW1: State order single manufacturing center names (14 names).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
MFNAME	(70)	DRW1: Manufacturing center names (70 centers).
MIACL1	(4)	DAMPSIN, DAMPCLC: Current quarter aggregate Class I consumption (4 quarters).
MIMP		DAMPSIN, DAMPSLV, DRW2: Number of import regions.
MMDM	(19)	DAMPSIN, DAMPSLV: Initially contains mileages from production areas to multiple manufacturing dummies in DAMPSIN. Later used to store unit transportation costs (19 production areas).
MMFC	(27)	DAMPSIN, DAMPSLV, DAMPCLC: Capacities for multiple manufacturing centers (27 centers).
MMFG	(75,42)	DAMPSLV, DAMPCLC: Current quarter individual movements from production centers to manufacturing centers (75 production centers, 42 manufacturing centers).
MMNC	(27,10)	DAMPSIN: Used to indicate which shipments from production centers to manufacturing centers occur at zero transportation cost (27 manufacturing centers, 10 production centers).
MOCC		DAMPCLC: Current quarter aggregate movements from processing centers to other than own consumption centers.
MODL		DAMPSIN, DRW1: Model number, e.g. DAMPS, Model A.
MOPC		DAMPCLC: Current quarter aggregate movements from production centers to other than own processing centers.
MPDC		DAMPSIN, DAMPSLV: If equal to 1, the amount of milk free to move in a quarter from production or processing centers to other than own areas is limited according to percentage specified in sections 28 or 24 of base data.
MPDPM	(59)	DAMPSIN, DAMPSLV: Quarterly, percentage decline which can occur in processor own area sales. Applies if PMR does not equal 1. (59 orders).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
MPDRM	(59)	DAMPSIN, DAMPSLV: Quarterly percentage decline which can occur in direct-ship production shipments to own area. Applies if RMR does not equal 1.(59 orders).
MPPC		DAMPCLC: Quarterly aggregate movements from production centers to own processing centers.
MR	(5)	DAMPSIN, DAMPCLC, DRW1: Print options for federal order reports (5 reports).
MRTL P		DAMPCLC: Aggregate average retail price for fluid products for current quarter.
MS		DAMPSIN, DAMPSLV, DRW1, DRW2: Number of Grade B milk production regions.
MSI		DAMPSIN: MS+MIMP.
MTPSQ	(5)	DAMPSIN, DAMPCLC: Aggregate total producer receipts for current quarter.
MXM		DAMPSIN: Equal to number of columns in arrays LMMFG, DMMFG, MMNC.
MXN		DAMPSIN, DAMPSLV: Equal to number of columns in array ICST.
MXP		DAMPSIN: Equal to number of rows in array ICST.
M1		DAMPSLV: First cheese import center node number.
M2		DAMPSLV: Last cheese import center node number.
M3		DAMPSLV: First miscellaneous Class III products import center node number.
M4		DAMPSLV: Last miscellaneous Class III products import center node number.
M5		DAMPSLV: First beginning commercial cheese stock center node number.
M6		DAMPSLV: Last beginning commercial cheese stock center node number.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
M7		DAMPSLV: First beginning government cheese stock center node number.
M8		DAMPSLV: Last beginning government cheese stock center node number.
M9		DAMPSLV: First Class II products consumption center node number.
M10		DAMPSLV: Last Class II products consumption center node number.
M11		DAMPSLV: First cheese consumption center node number.
M12		DAMPSLV: Last cheese consumption center node number.
M13		DAMPSLV: First miscellaneous Class III products consumption center node number.
M14		DAMPSLV: Last miscellaneous Class III products consumption center node number.
M15		DAMPSLV: Residual Sink node number for stage two.
M16		DAMPSLV: Super Source node number for stage two.
M17		DAMPSLV: Super Sink node number for stage two.
M18		DAMPSLV: First butter import center node number.
M19		DAMPSLV: Last butter import center node number.
M20		DAMPSLV: First commercial butter stock center node number.
M21		DAMPSLV: Last commercial butter stock center node number.
M22		DAMPSLV: First government butter stock center node number.
M23		DAMPSLV: Last government butter stock center node number.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
M24		DAMPSLV: First nonfat dry milk production center node number.
M25		DAMPSLV: Last nonfat dry milk production center node number.
M26		DAMPSLV: First nonfat dry milk import center node number.
M27		DAMPSLV: Last nonfat dry milk import center node number.
M28		DAMPSLV: First commercial nonfat dry milk stock center node number.
M29		DAMPSLV: Last commercial nonfat dry milk stock center node number.
M30		DAMPSLV: First government nonfat dry milk sector center node number.
M31		DAMPSLV: Last government nonfat dry milk sector center node number.
M32		DAMPSLV: First butter consumption center node number.
M33		DAMPSLV: Last butter consumption center node number.
M34		DAMPSLV: First nonfat dry milk consumption center node number.
M35		DAMPSLV: Last nonfat dry milk consumption center node number.
M36		DAMPSLV: Residual Sink node number for stage three.
M37		DAMPSLV: Super Source node number for stage three.
M38		DAMPSLV: Super Sink node number for stage three.
NAD		DAMPSIN, DAMPSLV: Parameter for GNETA. Specifies maximum number of arcs excluding connections to super source and super sink.
NAME	(9)	DRW1: Supply region names (9 regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
NAME1	(16)	DRW2: Supply, import, and manufactured milk demand regions names (16 regions).
NAME2	(3,14)	DRW2: Comparative Reports titles (14 reports).
NAME3	(3,5)	DRW2: Quantity and price units for tables.
NARC	(15)	DAMPSLV: Number of arcs in each arc category.
NARCS		DAMPSLV: Total number of arcs generated in a network.
NCL1P	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Current quarter federal and state order Class I price (59 orders).
NCL2P		DAMPSIN, DAMPSLV, DAMPCLC: Current quarter Class II price.
NCL2Q	(59,4)	DAMPSIN, DAMPCLC: Quarterly federal and state order Class II receipts for current year (59 orders, 4 quarters).
NCL3P		DAMPSIN, DAMPSLV, DAMPCLC: Current quarter Class III price.
NDM		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Number of federal order production centers connected to multiple manufacturing centers.
NDP		DAMPSIN, DAMPSLV: Number of last federal order dummy processing node.
NDP1		DAMPSIN, DAMPSLV: Number of last state order processing node.
NFP	(45)	DAMPCLC: Current quarter net farm price in federal orders (45 orders).
NIC		DAMPSIN, DAMPCLC, DRW1: Number of federal and state order comparative reports to be printed.
NIN1		DRW1: Tape 8, input file from DAMPSIN(IP1). DRW2: Tape 41, input file from DAMPSIN(OP8).
NIN2		DRW1: Tape 17, input file from DAMPCLC(OP1). DRW2: Tape 42, input file from DAMPSLV(OP7).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
NIN3		DRW1: Tape 16, input file from DAMPSLV(OP2). DRW2: Tape 43, input file from DAMPCLC(OP8).
NIXB		DAMPSIN, DAMPSLV: Number of regulated areas in unregulated regions.
NMD		DAMPSIN, DAMPSLV: Node number for multiple manufacturing area dummy.
NMFGT		DAMPSIN, DAMPCLC, DRW1: Equal to 1 greater than the larger of NMMF or NSMF.
NMM		DAMPSIN: Node number of last multiple manufacturing area.
NMMF		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Number of multiple manufacturing areas.
NMN		DAMPSIN, DAMPSLV: Node number of single manufacturing area dummy.
NMS		DAMPSIN, DAMPSLV: Node number of manufacturing sink.
NNN		DAMPSIN: Node number of last federal order single manufacturing area.
NNND		DAMPSIN, DAMPSLV: Equals NAD.
NNNE		DAMPSIN, DAMPSLV: Equals NNODES.
NNOD		DAMPSIN, DAMPSLV: Equal to array size for node length arrays in GNETA.
NNODES		DAMPSIN, DAMPSLV: Equal to actual number of nodes in largest network.
NOD		DAMPSLV: Equal to NNOD.
NOR	(75)	DAMPSIN, DAMPSLV, DRW1: Contains order numbers (75 orders).
NOUT		DRW2: Tape 6, printed output file.
NOUT1		DRW1: Tape 6, printed output file.
NOUT2		DRW1: Tape 4, output file for plotting routine.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
NPN		DAMPSIN, DAMPSLV: Node number of last state order processing center.
NPN1		DAMPSIN, DAMPSLV: Node number of last federal order single manufacturing center.
NQ		DAMPSIN, DAMPSLV, DAMPCLC, DRW1, DRW2: Total number of quarters to be run, equals 4·NY.
NQF		DRW2: (NY+1)5, number of quarters and years simulated plus 4 base quarters and 1 base year.
WRPB	(5)	DAMPSLV: Current quarter retail price of manufactured products (5 products).
NRPR	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Current quarter federal and state order retail price of fluid products (59 orders).
NSA1		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Number of federal orders.
NSALS		DAMPSIN, DAMPSLV, DRW1: Number of state orders.
NSC		DAMPSIN, DAMPSLV: Super Source node number, stage 1.
NSK		DAMPSIN, DAMPSLV: Super Sink node number, stage 1.
NSMF		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Number of federal order single manufacturing centers.
NSMFS		DAMPSIN, DAMPSLV: Number of state order single manufacturing centers.
NSMF1		DRW1: Equals NSMF+1.
NSMF2		DRW1: Equals NSMF+2.
NSN		DAMPSIN, DAMPSLV: Last direct-ship dummy production area with supply plant milk.
NSP		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Number of federal order production areas with supply plant milk.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
NSPM		DAMPSIN, DAMPCLC, DRW1: Number of single plus multiple manufacturing areas.
NSPMA		DRW1: Equals NSPM3 + NSALS.
NSPM2		DRW1: Equals NSPM + 2.
NSPM3		DAMPSIN, DRW1: Equals NSPM + 3.
NSPM4		DRW1: Equals NSPM + 4.
NSPX		DAMPSIN: Equals NSP.
NSX1		DAMPSIN: Equals NSAL.
NT		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Node number of last supply plant and total number of federal order direct-ship and supply plant areas.
NTN		DAMPSIN, DAMPSLV: Node number of last federal order consumption center.
NTN1		DAMPSIN: Node number of last state order consumption center.
NT1		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Node number of last state order direct-ship production center and total number of direct-ship and supply plant centers.
NT2		DAMPSIN, DAMPSLV, DRW1: Node numbers of last federal order direct-ship dummy.
NT3		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Total number of federal and state order processing centers.
NT4		DAMPSIN, DAMPSLV, DRW1: Node number of last federal order processing center.
NT5		DAMPSIN, DAMPCLC, DRW1: Node number of first federal order supply plant center.
NT6		DAMPSIN, DAMPCLC, DRW1: Node number of first state order direct-ship production center.
NT7		DAMPSIN, DRW1: NSMF + 1.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
NT8		DAMPSIN, DRW1: Total number of federal and state order single manufacturing centers.
NT9		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Total number of federal and state order areas.
NT10		DAMPSIN, DAMPCLC, DRW1: Equal to 1 plus the number of state order single manufacturing centers and the larger of NMMF or NSMF, i.e., NMF _{GT} + NSM _{FS} .
NT11		DAMPSIN, DAMPSLV, DRW1: Equals maximum number of links between a state order and other orders.
NW		DAMPSIN, DAMPSLV, DAMPCLC: Node number of last direct-ship production area without supply plant milk.
NY		DAMPSIN, DRW1, DRW2: Length of model run in years.
NYBASE		DAMPSIN, DAMPCLC, DRW1: Model base year.
NYCNT		DAMPSIN, DAMPSLV: Year counter, varies from 1 to 5.
NYID	(5)	DAMPSIN: Contains NYBASE + 1 to NYBASE + 5, i.e., the years that can be simulated.
NYRBC		DAMPSIN, DAMPSLV, DAMPCLC: Varies from NYBASE + 1 to NYBASE + 5, i.e., a simulation year.
NY2	(6)	DRW1, DRW2: Year label (base year plus up to 5 simulation years).
OACL1S	(59)	DAMPSLV: Current quarter out-of-area sales by federal and state processing centers (59 centers).
OACL1T	(59)	DAMPSLV, DAMPCLC: Current quarter cost of transportation corresponding to OACL1S by federal and state order processing center (59 orders).
OCST		DAMPSIN, DAMPSLV, DAMPCLC: Other costs associated with processing Class I milk.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
OCST2		DAMPSIN, DAMPSLV, DAMPCLC: Other costs associated with the production of Class II products in federal and state orders.
OMC1P	(75)	DAMPSIN: Distances used to generate a set of Class I prices from a Class I base price (75 production centers).
ONAME	(75)	DAMPSIN, DAMPSLV, DRW1: Contains federal and state production center names (75 centers).
OPCLL	(75)	DAMPSIN, DAMPSLV: If OPCLL equals 1, in-area sales are used as a Class I sales base.
OPEXF		DAMPSIN: If OPEXF equals 1, all exogenous factors vary.
OP1		DAMPSIN: Tape 8, output file to DRW1. DAMPCLC: Tape 17, output file to DRW1.
OP2		DAMPSIN: Tape 9, output file to DAMPCLC. DAMPSLV: Tape 16, output file to DRW1. DAMPCLC: Tape 18, scratch file.
OP3		DAMPSIN: Tape 6, printed output file. DAMPSLV: Tape 6, printed output file from all subprograms except GNETA.
OP4		DAMPSLV: Tape 6, printed output file from GNETA.
OP5		DAMPSIN: Tape 10, output file to DAMPSLV. DAMPSLV: Tape 10, scratch file.
OP6		DAMPSLV: Tape 13, output file to DAMPCLC. DAMPCLC: Tape 19, scratch file.
OP7		DAMPSIN: Tape 12, output file to DAMPCLC. DAMPSLV: Tape 42, output file to DRW2. DAMPCLC: Tape 12, input file from DAMPSIN.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
OP8		DAMPSIN: Tape 41, output file to DRW2. DAMPCLC: Tape 43, output file to DRW2.
PAM	(30)	DRW2: Gross farm price of all milk (30 quarters and years).
PCHGI	(4)	DAMPSIN, DAMPSLV: Annual, percentage change in imports (4 products).
PCL1	(59)	DAMPSLV, DAMPCLC: Current quarter producer receipts used in Class I (59 orders).
PCL2	(59)	DAMPCLC: Current quarter producer receipts used in Class II by market (59 orders).
PCR		DAMPSIN, DAMPSLV: If PCR equals 1, processing capacity is unlimited for all processing centers.
PDMI	(45,45)	DAMPSIN, DAMPSLV: Initially contains mile-ages from federal order processing to consumption centers in DAMPSIN. Then used to store unit transportation costs (45 orders).
PMCHG		DAMPSIN, DAMPSLV: Single producer marketing charge applicable to all areas.
PMCST	(59)	DAMPSLV: Used to store data for use in calculation of current quarter federal and state order fluid product retail prices (59 orders).
PMM	(59,59)	DAMPSLV, DAMPCLC: Current quarter individual package milk movements from processing to consumption centers (59 orders).
PMR		DAMPSIN, DAMPSLV: If PMR equals 1, processors are not constrained to make any sales in their own area.
POPB	(58)	DAMPSLV: Current quarter total product flows from each butter and nonfat dry milk production, import, and stocks center (58 centers).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
PRCST	(59)	DAMPSIN, DAMPSLV, DAMPCLC: Current quarter federal and state order processing cost (59 orders).
PRECAM	(10)	DAMPSLV, DAMPCLC: Total producer receipts for each consolidated order for the current year (10 consolidations).
PRRB	(10)	DAMPSLV: Current quarter total product flows to each butter and nonfat dry milk consumption center (10 consumption centers).
PRRD	(59)	DAMPSLV, DAMPCLC: Current quarter movements from direct-ship production areas to own area processors (59 orders).
PRREC	(59)	DAMPSLV, DAMPCLC: Current quarter total producer receipts (59 orders).
PRRECM	(20)	DAMPSLV, DAMPCLC: 1-10 contains the total producer receipts for each consolidated order for the current quarter; 11-20 contains the total Class I sales for each consolidated order for the current year.
PRROD	(59)	DAMPSLV: Current quarter processing center receipts of direct-ship milk from other than own area (59 order processing centers).
PRRS	(16)	DAMPSLV, DAMPCLC: Current quarter processing center receipts from own area supply plants (16 plants).
PSE2		DAMPSIN, DAMPSLV: Federal and state order Class II demand function price elasticity.
QCL1P	(59,20)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly Class I prices if read in by user or generated from a base price (59 orders, 20 quarters).
QCL2P	(20)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly Class II prices, if read in by user (20 quarters).
QCL3P	(20)	DAMPSIN, DAMPSLV, DAMPCLC: Quarterly Class III prices, if read in by user (20 quarters).
QCNT		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Quarter counter, varies from 1 to NY times 4.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
QCNT1		DAMPSIN, DAMPSLV, DAMPCLC: Equals CCNT.
QR	(20)	DAMPSIN, DAMPCLC, DRW1: Quarterly options for federal order output (20 quarters).
QRB	(5)	DAMPSIN, DRW1, DRW2: Grade B report writer options from input form for each simulation year (5 years).
QSPLL	(59)	DAMPSLV, DAMPCLC: Current quarter federal and state order production; see CQTSP (59 orders).
QSPR	(16)	DAMPSLV: Equals the percentage that the supply plant shipping requirement is of the sum of the federal order direct-ship to own-area shipping requirement plus the supply plant shipping requirement (16 plants).
QT		DAMPSIN, DAMPSLV, DAMPCLC, DRW1: Counter, varies from 6 to 9.
RDA	(5)	DAMPSLV: Used to pass arc and node information to GNETA ("from" node number, "to" node number, unit cost, upper bound, lower bound).
RESID	(18,2)	DAMPSLV: Movements from production nodes to the residual sink in stages two or three (up to 18 regions, raw milk or butter and nonfat dry milk production nodes).
RESIDC	(18,2)	DAMPSLV: Movements from commercial stocks nodes to the residual sink in stages two or three (up to 18 regions, cheese or butter and nonfat dry milk stocks).
RESIDG	(18,2)	DAMPSLV: Movements from government stocks nodes to the residual sink (up to 18 regions, cheese or butter and nonfat dry milk stocks).
RMGN		DAMPSIN, DAMPSLV: Retail margin for packaged (fluid) milk.
RMMA	(59,59)	DAMPCLC: Accumulated movements from direct-ship production centers to processing centers (59 production and 59 processing areas)

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
RMMD	(59,59)	DAMPSLV, DAMPCLC: Current quarter individual movements from direct-ship production centers to processing centers (59 production centers, 59 processing centers).
RMMS	(16,59)	DAMPSLV, DAMPCLC: Current quarter individual movements from supply plants to processing centers (16 plants, 59 orders).
RMPC	(75)	DAMPSIN, DAMPSLV, DAMPCLC: Direct raw milk production cost by federal and state production centers (75 production centers).
RMR		DAMPSIN, DAMPSLV: If RMR equals 1, production centers are not constrained to ship milk to their own processing center.
RODC	(45)	DAMPCLC: Current quarter returns over direct cost to milk producers in federal order areas (45 orders).
RODCAM	(30)	DRW2: Returns over direct cost to all producers of milk (30 quarters and years).
RODCB	(30)	DRW2: Returns over direct cost to all producers of Grade B milk (30 quarters and years).
RPB	(5,5)	DAMPSIN, DAMPSLV: Base year retail prices of manufactured milk products (5 products, 5 quarters).
RSS	(9)	DAMPSIN, DAMPSLV: Percentage of total government cheese stock held in each region (9 regions).
SAD	(14,4)	DAMPSIN: Data from input form S-A (14 states, 4 categories of data).
SCALE1		DRW1, DRW2: Scales data by 0.001.
SCALE2		DRW1, DRW2: Scales data by 0.01.
SCALE3		DRW1, DRW2: Scales data by 0.0001.
SCLLCP		DAMPSIN, DAMPSLV, DAMPCLC: Aggregate federal order Class I processing capacity inflated by reserve requirements.
SCLLD		DAMPCLC: Current quarter aggregate federal order Class I consumption.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
SCL2D		DAMPCLC: Current quarter aggregate federal order Class II consumption.
SCL3Q		DAMPCLC: Current quarter aggregate federal order Class III consumption.
SE	(59)	DAMPSIN, DAMPSLV: Federal and state order milk production center price elasticities of supply (59 orders).
SERCST	(5)	DAMPSIN: Used in calculations for report writer from base year data.
SE2		DAMPSIN: Class II price elasticity of demand.
SFPCST		DAMPSIN, DAMPCLC: Aggregate quarterly transportation and assembly charges for supply plant producers.
SGRC1		DAMPCLC: Current quarter aggregate federal order Class I sales.
SIDX	(3,4)	DAMPSIN, DAMPSLV: Indices of seasonal inventory demands for commercial stocks as a percentage of regional sales (3 products, 4 quarters).
SMC20		DAMPSLV, DAMPCLC: Aggregate federal order Class II sales made from nonorder sources for current quarter.
SMDS		DAMPCLC: Current quarter aggregate shipments from federal order direct-ship production areas to manufacturing centers.
SMDSD		DAMPCLC: Current quarter aggregate shipments from federal order direct-ship production areas to manufacturing center dummies.
SMFC	(40)	DAMPSIN, DAMPSLV, DAMPCLC: Single manufacturing center capacities (26 centers).
SMFGU		DAMPCLC: Current quarter percent utilization of aggregate federal order manufacturing capacity.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
SMFC		DAMPSIN, DAMPSLV: Aggregate multiple manufacturing center capacity.
SMPC		DAMPSLV: Current quarter aggregate sales by federal order processors in their own area.
SMSP		DAMPSLV: Current quarter aggregate shipments from supply plants to manufacturing centers.
SMSPD		DAMPCLC: Current quarter aggregate movements from federal order supply plants to manufacturing center dummies.
SMUCP		DAMPCLC: Current quarter aggregate federal order percentage utilization of Class I processing capacity.
SMXC		DAMPSIN, DAMPSLV: Used to pass MXC to GNETA.
SNFP		DAMPSIN, DAMPSLV: Current quarter aggregate federal order average net farm price.
SOACLD		DAMPCLC: Current quarter aggregate federal order average per unit transportation costs on out-of-area packaged milk sales.
SOACLT		DAMPCLC: Current quarter aggregate federal order transportation costs on out-of-area packaged milk sales.
SPCL1		DAMPCLC: Current quarter aggregate federal order producer receipts used in Class I.
SPLB	(16,4)	DAMPSIN, DAMPSLV: Federal order supply plant shipping requirements by quarter (16 plants, 4 quarters).
SPMI	(61,45)	DAMPSIN, DAMPSLV: Initially contains mileages between federal order production centers and processing centers in DAMPSIN. Then used to store unit transportation costs (61 production centers, 45 processing centers).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
SPOPD	(59)	DAMPSLV; DAMPCLC: Current quarter movements from direct-ship production centers to processing centers (59 production centers).
SPOPS		DAMPSLV, DAMPCLC: Current quarter movements from supply plants to other area processing centers (16 plants).
SPRC	(16)	DAMPSIN, DAMPSLV, DAMPCLC: Current quarter percentage that supply plant supply is of total supply in areas with supply plant milk (16 plants).
SPRREC		DAMPCLC: Current quarter aggregate federal order producer receipts.
SRMPC	(5)	DAMPSIN: Used in calculations for report writer from base year data.
SSMFC		DAMPSIN, DAMPSLV: Aggregate federal order single manufacturing center capacity.
SSPMI	(14,8,8)	DAMPSIN, DAMPSLV: Initially contains indices of possible links between state and federal order areas in DAMPSIN; distances are later replaced by unit transportation costs. Types of links permitted are as follows: state order production to federal order processing, federal and state order production to state order processing, state order processing to federal order consumption, federal and state order processing to state consumption (14 states up to 8 links, indices and distances for each of 4 types of linkages).
SSOPC		DAMPCLC: Current quarter aggregate shipments from supply plants to own processing centers.
STCCWT		DAMPCLC: Current quarter aggregate federal order average unit cost on Class I sales.
STRMC		DAMPCLC: Current quarter aggregate federal order average unit raw milk acquisition cost.
STRNSD		DAMPCLC: Current quarter aggregate federal order average unit raw milk transportation cost to processing centers.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
STRNSR		DAMPCLC: The aggregate quarterly handling charges for supply plant milk shipped to other orders.
SU	(30,18)	DRW2: Results for Summary Reports in DRW2. (30 quarters and years, 18 variables).
SVCLI		DAMPCLC, DAMPSLV: Current quarter aggregate value of movements from direct-ship production centers to processing centers.
SVFRPS		DAMPCLC: Current quarter aggregate federal order value of farm-retail fluid milk price spread multiplied by Class I consumption.
SVP		DAMPCLC: Current quarter aggregate federal order pool value.
	(30)	DRW2: Gross returns to Grade A milk producers in regulated areas, see SVPl (30 quarters and years).
SVPAM	(30)	DRW2: Gross returns to all milk producers (30 quarters and years).
SVPB	(30)	DRW2: Gross returns to Grade B milk producers (30 quarters and years).
SVPl		DAMPCLC: Current quarter aggregate federal and state order pool value.
SVRODC		DAMPCLC: Current quarter aggregate average return over direct cost to producers in federal orders.
	(30)	DRW2: Aggregate average return over direct cost to all producers in federal and state orders, see SVRl (30 quarters and years).
SVRl		DAMPCLC: Current quarter aggregate average return over direct cost to producers in federal and state orders.
TCCWT	(59)	DAMPCLC: Quarterly total unit Class I cost by processing center (59 orders).
TCL3	(59)	DAMPCLC, DAMPSLV: Current quarter Class III receipts (59 orders).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
TCMP	(30)	DRW2: Aggregate, quarterly transportation costs for all manufactured products (30 quarters and years).
TITLE		DAMPSIN, DAMPSLV, DAMPCLC: Same as DNUM.
TMCST		DAMPSIN, DAMPSLV, DAMPCLC: Aggregate quarterly handling charges for milk shipped to federal order manufacturing plants in other orders.
TMFGM		DAMPCLC: Current quarter aggregate movements from federal order production centers to manufacturing centers.
TMFGU		DAMPCLC: Current quarter aggregate federal order percentage utilization of manufacturing center capacity.
TMMC	(53)	DAMPCLC: Current quarter milk received at each federal order manufacturing center (53 centers).
TMOP		DAMPSLV, DAMPCLC: Current quarter aggregate movements from supply plants to other area processors.
TPME		DRW2: Total milk production.
TPSQ	(59,4)	DAMPSIN, DAMPCLC: Quarterly producer receipts for current year (59 orders, 4 quarters).
TRANSRD	(59)	DAMPSLV: Quarterly acquisition cost to processing centers for milk from supply plants (59 centers).
TRANSRP	(59)	DAMPSLV, DAMPCLC: Quarterly transportation cost to processing centers for milk received from supply plants (59 centers).
TRCS1		DAMPSLV, DRW2: Current quarter aggregate transportation cost in stage 2.
TRCS2		DAMPSLV, DRW2: Current quarter aggregate transportation cost in stage 3.
TRMC	(59)	DAMPSLV, DAMPCLC: Current quarter acquisition cost for raw milk by processing center (59 centers).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
TRMPC		DAMPSIN, DAMPCLC: Aggregate federal order quarterly raw milk production costs.
TSMFC		DAMPSIN, DAMPCLC: Aggregate federal order manufacturing center capacity.
TSTP	(30)	DRW2: Total commercial and government stocks as a percent of total milk production (30 quarters and years).
UMCP	(53)	DAMPCLC: Current quarter percent utilization of federal order manufacturing center capacity (53 centers).
UNCEXP		DAMPSIN: Current quarter total consumer expenditure on fluid products in unregulated Grade A milk regions in the base year.
	(30)	DRW2: Aggregate quarterly consumer expenditures on fluid products in unregulated Grade A regions (30 quarters and years).
UNCLP		DAMPSIN: Average price paid to producers in unregulated Grade A milk regions for milk used for fluid purposes.
UNCLS		DAMPSIN: Quarterly base year total sales of fluid products in unregulated Grade A milk regions.
	(9)	DRW1: Current quarter sales of fluid products in unregulated Grade A milk regions (9 regions)
	(30,9)	DRW2: Consumption of fluid products in unregulated Grade A milk regions (30 quarters and years, 9 regions).
UNFPHC		DAMPSIN, DAMPSLV: Farm to plant hauling cost for unregulated Grade A milk producers.
UNGBP	(9,4)	DAMPSIN, DAMPSLV: Base year quarterly gross farm prices of Grade A milk in unregulated regions (9 regions, 4 quarters).
	(9)	DRW1: Current quarter gross farm prices of Grade A milk in unregulated regions (9 regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
UNGBP	(30,10)	DRW2: Gross farm prices of Grade A milk in unregulated regions (30 quarters and years, 9 regions plus average for all regions).
UNNBP		DAMPSIN: Quarterly base year average net farm price of Grade A milk in unregulated regions, i.e., gross price less direct production costs.
	(9)	DRW1: Current quarter net farm price of Grade A milk in unregulated regions (9 regions).
UNNRP		DAMPSIN: Quarterly base year average retail price of fluid products in unregulated Grade A milk regions.
UNPC		DAMPSIN, DAMPSLV: Direct cost of producing Grade A milk in unregulated regions.
UNPMC		DAMPSIN, DAMPSLV: Unregulated Grade A milk producer marketing charge.
UNREG		DAMPSIN: Quarterly base year production of Grade A milk in unregulated regions.
	(9)	DRW1: Current quarter production of Grade A milk in unregulated regions.
	(30,10)	DRW2: Production of Grade A milk in unregulated regions (30 quarters and years, 9 regions plus total of all regions).
UNRODC		DAMPSIN; Quarterly base year average returns over direct costs to producers of Grade A milk in unregulated regions.
	(9)	DRW1: Current quarter returns over direct costs to producers of Grade A milk in unregulated regions (9 regions).
	(30,10)	DRW2: Returns over direct costs to producers of Grade A milk in unregulated regions (30 quarters and years, 9 regions plus average of all regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
UNRS	(9)	DAMPSIN, DAMPSLV: Percentage of total production of unregulated Grade A milk in each unregulated region (9 regions).
UNS	(4)	DAMPSIN, DAMPSLV: Aggregate, base year production of Grade A milk in all unregulated regions (4 quarters).
UNUT		DAMPSIN, DAMPSLV, DRW1: Percentage of Grade A milk produced in unregulated regions that is consumed as fluid milk products.
UNVP		DAMPSIN: Quarterly average base year gross returns to producers of Grade A milk in unregulated regions.
	(30,10)	DRW2: Gross returns to producers of Grade A milk in unregulated regions (30 quarters and years, 9 regions plus sum of all regions).
UTCP	(59)	DAMPSLV: Current quarter percent utilization of Class I processing capacity by processing center (59 centers).
VP	(59)	DAMPSLV, DAMPCLC: Current quarter order pool values (59 orders).
VPAMRG	(10)	DAMPSLV, DAMPCLC: Total Class I sales for each consolidated order for the current year (10 consolidations).
VPMRG	(20)	DAMPSLV, DAMPCLC: 1-10 contains the value of the pool for each of the consolidated orders for the current quarter; 11-20 contains the total transportation charges for each of the consolidated orders for the current quarter.
WCL1D		DAMPCLC: Current quarter average federal order Class I demand.
WCL1P		DAMPCLC: Current quarter weighted average federal order Class I price.
WFRPS		DAMPCLC: Current quarter aggregate federal order average farm-retail fluid milk price spread.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
WK1	(9)	DAMPSLV: Work array (9 regions).
WK2	(9)	DAMPSLV: Work array (9 regions).
WORK1	(14992)	DAMPSIN, DAMPCLC: Work array equal to the size of common in GNETA.
WORK2	(8025)	DAMPSIN: Work array equal to the size of common in BLK1.
	(11619)	DAMPSLV, DAMPCLC: Work array equal to the size of common in BLK1.
WORK3	(24)	DAMPCLC: Work array equal to the size of common in TCOST, BLK2 (same as WORK4 in DAMPSIN and DAMPSLV).
WORK4	(24)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK2 (same as WORK3).
	(890)	DAMPCLC: Work array equal to size of common in BLK5 (same as WORK5 in DAMPSIN and DAMPSLV).
WORK5	(890)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK5 (same as WORK4 in DAMPCLC).
	(45)	DAMPCLC: Work array equal to size of common in BLK7 (same as WORK6 in DAMPSIN and DAMPSLV).
WORK6	(45)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK7 up to and including QCNT1 (same as WORK5 in DAMPCLC).
	(11)	DAMPCLC: Work array equal to size of common in BLK9 (same as WORK8 in DAMPSIN and DAMPSLV).
WORK7	(407)	DAMPCLC: Work array (same as WORK9 in DAMPSIN and DAMPSLV).
WORK8	(11)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK9 (same as WORK6 in DAMPCLC).
	(852)	DAMPCLC: Work array (same as WORK10 in DAMPSIN AND DAMPSLV).
WORK9	(407)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK10 (same as WORK7).
	(18)	DAMPCLC: Work array (same as WORK11).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
WORK10	(852)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK11 up to and including EFRR (same as WORK8 in DAMPCLC).
	(10)	DAMPCLC: Work array (same as WORK13 in DAMPSIN and DAMPSLV).
WORK11	(18)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK12 (same as WORK9 in DAMPCLC).
WORK12	(767)	DAMPSIN, DAMPSLV: Work array equal to size of common in BLK14.
WORK13	(10)	DAMPSIN: Equal to size of common in BLK11 after EFRR, see WORK10 (same as WORK10 in DAMPCLC).
W1	(11619)	DAMPSLV: Work array equal to size of common in BLK1, see WORK2.
	(9)	DRW2: Quarterly Grade A milk production in unregulated areas (9 supply regions).
W2	(890)	DAMPSLV: Work array equal to size of common in BLK5, see WORK5.
	(9)	DRW2: Quarterly Grade B milk production (9 supply regions).
W3	(852)	DAMPSLV: Work array equal to size of common in BLK11, see WORK10.
	(9)	DRW2: Quarterly fluid products consumption (9 regions).
W4	(2104)	DAMPSLV: Work array equal to size of common in BLK13.
	(9)	DRW2: Quarterly Grade A milk from regulated areas used in manufacturing (9 supply regions).
W5	(4,2)	DRW2: Quarterly imports (4 products, 2 import regions).
W6	(6,9)	DRW2: Quarterly ending stocks (3 products and 2 stock types, 9 regions).
W7	(5,5)	DRW2: Quarterly manufactured products consumption (5 products, 5 consumption regions).

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
W8	(5)	DRW2: Quarterly retail prices of manufactured products (5 products).
W9	(20)	DRW2: Equals CHGPD in DAMPSIN (20 quarters).
XBP		DAMPSIN, DAMPSLV, DAMPCLC: Starting quarter Class II price.
XBP3		DAMPSIN, DAMPSLV, DAMPCLC: Starting quarter Class III price.
XCEXP		DAMPCLC: Change in CEXP from same quarter a year ago.
XCFRP		DAMPCLC: Change in SVFRPS from same quarter a year ago.
XCHABLN		DAMPCLC: Annual change in AMBLND from a year ago.
XCHACEX		DAMPCLC: Annual change in ACEXP from a year ago.
XCHACP		DAMPCLC: Annual change in AMUCP from a year ago.
XCHAC1		DAMPCLC: Annual change in AMCLP from a year ago.
XCHAC2		DAMPCLC: Annual change in ASCL2D from a year ago.
XCHAC3		DAMPCLC: Annual change in ASCL3Q from a year ago.
XCHAFRP		DAMPCLC: Annual change in ASFRPS from a year ago.
XCHAGU		DAMPCLC: Annual change in ASMFGU from a year ago.
XCHAMRT		DAMPCLC: Annual change in AMRTL from a year ago.
XCHANFP		DAMPCLC: Annual change in ASNFP from a year ago.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
XCHAPC		DAMPCLC: Annual change in ASIAC1 from a year ago.
XCHAPR		DAMPCLC: Annual change in ASPRRC from a year ago.
XCHARDC		DAMPCLC: Annual change in ASRODC from a year ago.
XCHASO		DAMPCLC: Annual change in ASOAC1T from a year ago.
XCHATR		DAMPCLC: Annual change in ASTRNSR from same quarter a year ago.
XCHAU		DAMPCLC: Annual change in AMCIU from a year ago.
XCHAVP		DAMPCLC: Annual change in AVP from a year ago.
XCHAWF		DAMPCLC: Annual change in AWRFRPS from a year ago.
XCHA2		DAMPCLC: Annual change in AMC2P from a year ago.
XCHA3		DAMPCLC: Annual change in AMC3P from a year ago.
XCHC2P		DAMPCLC: Quarterly change in NCL2P from same quarter a year ago.
XCHC3P		DAMPCLC: Change in NCL3P from same quarter a year ago.
XCHMB		DAMPCLC: Change in MBLND from same quarter a year ago.
XCL2P		DAMPSIN, DAMPSLV, DAMPCLC: Current quarter Class II price, used in demand function CL2D.
XCMC1P		DAMPCLC: Change in MCL1P from same quarter a year ago.
XCMC1U		DAMPCLC: Change in MCL1U from same quarter a year ago.

<u>Variable</u>	<u>Dimension</u>	<u>Location and Description</u>
XCMC2		DAMPCLC: Change in SCL2D from same quarter a year ago.
XCMC3		DAMPCLC: Change in SCL3Q from same quarter a year ago.
XCMFU		DAMPCLC: Change in SMFGU from same quarter a year ago.
XCMRTP		DAMPCLC: Change in MRTLP from same quarter a year ago.
XCNFP		DAMPCLC: Change in SNFP from same quarter a year ago.
XCRDC		DAMPCLC: Change in SVRODC from same quarter a year ago.
XCSOA		DAMPCLC: Change in SOAC1T from same quarter a year ago.
XCSPC1		DAMPCLC: Change in SPCL1 from same quarter a year ago.
XCSPRC		DAMPCLC: Change in SPRREC from same quarter a year ago.
XCSVP		DAMPCLC: Change in SVP from same quarter a year ago.
XCTRNS		DAMPCLC: Change in STRNSR from same quarter a year ago.
XCUCP		DAMPCLC: Change in SMUCP from same quarter a year ago.
XCWFR		DAMPCLC: Change in WFRPS from same quarter a year ago.
YEND		DRW1: Year counter, varies from 1 to maximum number of year simulated.

BIBLIOGRAPHY

1. Babb, E.M., D.E. Banker, O. Goldman, D.R. Martella, and J.E. Pratt, "Economic Model of Federal Milk Marketing Order Policy Simulator--Model A," Station Bulletin No. 158, Agricultural Experiment Station, Purdue University, April, 1977.
2. Babb, E.M., D.E. Banker, O. Goldman, D.R. Martella, and J.E. Pratt, "User's Manual for Federal Milk Marketing Order Policy Simulator--Model A," Station Bulletin No. 157, Agricultural Experiment Station, Purdue University, April, 1977.
3. Babb, E.M. and J.E. Pratt, "Projections of Federal Milk Marketing Order Performance Under Alternative Pricing and Pooling Provisions," Station Bulletin No. 171, Agricultural Experiment Station, Purdue University, October, 1977.
4. Banker, David E., "A Model for the Analysis of Alternative Pricing Policies in Federal Milk Marketing Orders," unpublished Ph.D. thesis, Purdue University, August, 1977.
5. Banker, D.E., E.M. Babb, O. Goldman, D.R. Martella, and J.E. Pratt, "Computer Program Documentation for Federal Milk Marketing Order Policy Simulator--Model A," Station Bulletin No. 164, Agricultural Experiment Station, Purdue University, June, 1977.
6. Martella, D.R., D.E. Banker, E.M. Babb, and J.E. Pratt, "Computer Program Documentation for Federal Milk Marketing Order Policy Simulator--Model B," Station Bulletin No. 172, Agricultural Experiment Station, Purdue University, October, 1977.
7. Novakovic, Andrew Milovan, "An Economic Analysis of the U.S. Dairy Price Support Program and Alternate Policies," unpublished Ph.D. thesis, Purdue University, May, 1979.
8. Novakovic, A.M., E.M. Babb, D.R. Martella, and J.E. Pratt, "Input Forms and Output from the Dairy Market Policy Simulator (Model A), Staff Paper 79-5, Department of Agricultural Economics, Cornell University, May, 1979.