



US Army Corps of Engineers  
Water Resources Support Center  
Institute for Water Resources

---

# **INTERNATIONAL GRAIN TRANSPORTATION NETWORK MODEL: SOFT WHEAT**

**SEPTEMBER 1990**

**IWR REPORT 90-R-2**

**INTERNATIONAL GRAIN TRANSPORTATION NETWORK MODEL:  
SOFT WHEAT**

**Prepared By:**

**Stephen W. Fuller  
Warren R. Grant  
Luis R. Fellin  
Swee Hor Teh  
Nicolas Gutierrez**

**Department of Agricultural Economics  
Texas Agricultural Experiment Station  
The Texas A&M University System  
College Station, TX 77843**

**Commissioned By:**

**U.S. Army Corps of Engineers  
Water Resources Support Center  
Institute for Water Resources  
Casey Building  
Ft. Belvoir, VA 22060-5586**

	Content	page
<b>Introduction.....</b>		<b>i</b>
<b>Programs</b>		
sft1.f.....		1
sft2.f.....		18
sft3.f.....		29
<b>Data files</b>		
f1.sft.....		56
f2.sft.....		63
f3.sft.....		72
f4.sft.....		84
f8.sft.....		89
f9.sft.....		91
<b>Output</b>		
file06_2.sorg.....		95
final.out.....		96
<b>Calibration.....</b>		<b>106</b>

## **INTRODUCTION**

This report documents the soft wheat model. Additional reports document corn, soybeans, hard red winter wheat, grain sorghum, hard red spring wheat and durum wheat models. A tutorial report and model and data requirements report are published separately.

In this report, the documented programs, data files and output listing are included. For the compilation and execution of the network model, the three Fortran 77 programs and six data files are presented. The programs have to be compiled and run in a sequential order (program1 followed by program2, etc.). The data must be entered into the corresponding data files.

The documented FORTRAN 77 programs and grain related data files used in the model are provided. However, the documented programs and data files cannot be used to execute the model. In each program or data file, explanations are included to provide more detail to the user.

The intermediate output listing and final output listing are included in this report. The intermediate output listing is printed in order to explain the feasibility of the data provided. The final output listings show the optimal result of the transportation network model.

The model was calibrated with Federal Grain Inspection Service, U.S. Department of Agriculture, trade data. Special adjustments used in this calibration are noted in the calibration section. The calibrated model data and source code programs are included on the attached diskette.

## **ACKNOWLEDGEMENTS**

This research was supported through USDA Cooperative Agreement Number 58-3AEK-8-00120 with resources made available by the Water Resources Support Center - Institute of Water Resources, U.S. Army Corps of Engineers, Casey Building, Ft. Belvoir, Virginia. Special appreciation is expressed to Dr. Lloyd G. Antle, Chief, Navigation Division and Ms. Shilpa Patel, Computer Systems Analyst, both with the Water Resources Support Center - Institute for Water Resources. In addition, appreciation is expressed to personnel in District Offices, U.S. Army Corps of Engineers, who offered valuable comments during our research effort. Appreciation is also expressed to Dr. Lester H. Myers, Branch Chief, USDA-CED-ERS-FMCE.

```
C ---- ****  
C ---- * SOFT1.F *  
C ---- * Documented on Sept 1990 *  
C ---- * Run on a 386 machine *  
C ---- * Compiler used NDP Fortran *  
C ---- * Grain used SOFT (SOFT1.F) *  
C ---- * Data files used F1,F2,F3,F4, and F8. *  
C ---- * Trace file U6 *  
C ---- * Input file for SOFT2.F U12 *  
C ---- ****
```

```
C ---- Declaration.  
C ---- The array size used is 27500.
```

```
COMMON /G0/ K, OPERATION  
COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G3/ LOWR(27500)  
COMMON /G4/ UPPR(27500) /G5/ KOST(27500)  
INTEGER UPPR  
INTEGER U1,U2,U3,U4,U8,U9,U6,U12
```

```
C ---- Unit number for each file.  
C ---- U1 is the unit number for data file, F1.sft.  
C ---- F1.sft contains information about the model.  
C ---- U2 is the unit number for data file, F2.sft.  
C ---- F2.sft contains information on TRUCK mileage.  
C ---- U3 is the unit number for data file, F3.sft.  
C ---- F3.sft contains information on RAIL costs.  
C ---- U4 is the unit number for data file, F4.sft.  
C ---- F4.sft contains information on BARGE costs.  
C ---- U8 is the unit number for data file, F8.sft.  
C ---- F8.sft contains information about SUPPLIES and DEMANDS.  
C ---- U9 is for data files, F9.sft.  
C ---- F9.sft contains the NAMES of all the regions.  
C ---- U6 is the unit number for a temporary trace file.  
C ---- The trace output file is used to check that the program is  
C ---- working correctly.  
C ---- U12 is the unit number for the output file.  
C ---- The output produced is used as an input file for the next  
C ---- program, SOFT2.
```

```
U1 = 13  
U2 = 14  
U3 = 15  
U4 = 16  
U8 = 8  
U9 = 9  
U6 = 17  
U12 = 12
```

```
C ---- Diagram used to show the flow of this program, SOFT1.F:
```

```

C -----
C -----          U1,U2,U3,U4,U8,U9 (input files)
C -----          |
C -----          V
C -----          SOFT1.F (program)
C -----          |
C -----          /   \
C -----          /     \
C -----          V       V
C -----          U6 (trace)   U12 (input to SOFT2.F)

```

C ---- To open files for reading and writing

```

OPEN (UNIT = U1, FILE = 'F1.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U2, FILE = 'F2.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U3, FILE = 'F3.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U4, FILE = 'F4.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U8, FILE = 'F8.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U6, FILE = 'FILE06_2.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U12, FILE = 'FILE12_2.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'UNFORMATTED')

```

C ---- Initialisation and main program.

C ---- INFN is a large number used to initialise the amount produced  
C ---- in each region.

C ---- LOWR is the amount of grain demanded in this model.

C ---- UPPR is the amount of grain supplied in this model.

C ---- KOST is the cost of transportation for each region.

C ---- DRIVER is a subroutine used to read in the data files and  
C ---- invoke other subroutines, SURPLS, DEMAND, WRITER, PELVTR,  
C ---- and RELVTR.

```

INFN      = 99999999
DO 1100 K = 1, 27500
LOWR(K)   = 0
UPPR(K)   = INFN
KOST(K)   = 0
1100 CONTINUE
K           = 0
CALL DRIVER
STOP
END

```

```

-----
C           ===== WRITER =====
C ---- Write all nodes, iarc, jarc, lowr, uppr and cost

```

```
C ---- to the data file in the unit number U12.
```

```
SUBROUTINE WRITER ( N, NODE, I, J, K, L, M )
DIMENSION I(N), J(N), K(N), L(N), M(N)
INTEGER U1,U2,U3,U4,U8,U9,U6,U12
```

```
C ---- Write N(= number of arcs), NODE(= number of nodes),
C ---- I(= IARC), J(= JARC), K(= LOWR), L(= UPPER), M(=KOST),
C ---- N(=NODES) to the next program, SOFT2.F.
```

```
      WRITE (U12,510) N, NODE, I, J, K, L, M, N
510 FORMAT ( 20I8 )
      ENDFILE U12
      RETURN
      END
```

```
C-----
C '           =====  DRIVER  ======
C ---- Read in data and invoke other subroutines.
C ---- Read in the number of surplus, deficit, river
C ---- and port locations and their code names.
C ---- Calculate the number of nodes connected.
C ---- Read in the loading and unloading costs.
```

```
SUBROUTINE DRIVER
COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /A3/ SRND, REND, DRND, PEND, FRND, DMND
COMMON /A4/ SINK, SRCE
COMMON /B1/ SRGN(65) /B2/ DRGN(65) /B3/ RIVR(45)
COMMON /B4/ PORT(20) /B5/ FRGN(25)
COMMON /F3/ TL0S, RL0S, TL0R, RL0R, BL0R, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION
COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G3/ LOWR(27500)
COMMON /G4/ UPPR(27500) /G5/ KOST(27500)
DIMENSION TITL(16)
INTEGER TITL
INTEGER SRND, REND, DRND, PEND, FRND, DMND
INTEGER SINK, SRCE
INTEGER SRGN, DRGN, RIVR, PORT, FRGN
INTEGER UPPR, ARCS
INTEGER U1,U2,U3,U4,U8,U9,U6,U12
```

```
C ---- Outline the layout of the output.
```

```
500 FORMAT ( 20A4 )
510 FORMAT ( 20I4 )
520 FORMAT ( 10F8.3 )
600 FORMAT ( 1H1, 5X, 'NETWORK GENERATOR', //,
1       6X, 'FOR GRAIN SHIPMENT PROBLEM', // )
610 FORMAT ( 6X, 'SUPPLY', I15, 6X, 'DEMAND', I15, / )
620 FORMAT ( 6X, ?????????????????????????????????????', //,
1       6X, 'INFEASIBLE NETWORK. DEMAND EXCEEDS SUPPLY', //,
```

2           6X, '???', / )  
650 FORMAT ( 317, 3110 )

C ---- Write the heading and read and write the title of grain.

```
WRITE (U6,600)
READ  (U1,500) TITL
WRITE (U6,500) TITL
```

C ---- Read in the number of surplus, deficit, river, port and  
C ---- foreign regions; the time period and the number of days  
C ---- in each time period.

C ---- Read in the code names of all the surplus regions.

C ---- Check that there is a region before reading the code name.

```
READ  (U1,510) NOSR, NODR, NORE, NOPE, NOFR
READ  (U1,510) NOTP, ( NOAY(I), I = 1, NOTP )
READ  (U1,500) ( SRGN(I), I = 1, NOSR )
IF ( NODR .GT. 0 ) READ  (U1,500) ( DRGN(I), I = 1, NODR )
IF ( NORE .GT. 0 ) READ  (U1,500) ( RIVR(I), I = 1, NORE )
IF ( NOPE .GT. 0 ) READ  (U1,500) ( PORT(I), I = 1, NOPE )
IF ( NOFR .GT. 0 ) READ  (U1,500) ( FRGN(I), I = 1, NOFR )
```

C ---- Calculate the number of shipments by time.

C ---- Find all the connecting nodes in this model.

```
NOTF      = NOTP + 1
SRND      = NOTF * NOSR
DRND      = NOTF * NODR + SRND
REND      = NOTF * NORE + DRND
PEND      = NOTF * NOPE + REND
FRND      = NOTF * NOFR + PEND
DMND      = NODR + NOFR + FRND
SINK      = DMND + 1
SRCE      = SINK + 1
NODE      = SRCE
```

C ---- Read in the loading(LO) and unloading(RI) factors.

C ---- T stands for Truck, R for Rail, S for Ship and B for Barge.

```
READ  (U1,520) TLOS, RLDS, TLOR, RLOR, BLOR, SLOP
READ  (U1,520) TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
```

C ---- Initialisation and invoke other subroutines.

C ---- IPRD is the amount of grain produced.

C ---- IEXP is the amount of grained demanded.

C ---- Find the cost of transportation (RIVER) by calling RELVTR.

C ---- Find the cost of transportation (PORT) by calling PELVTR.

```
IPRD      = 0
IEXP      = 0
K          = 0
CALL SURPLS ( IPRD )
IF ( NORE .GT. 0 ) CALL RELVTR
IF ( NOPE .GT. 0 ) CALL PELVTR
```

CALL DEMAND ( IEXP )

C ---- Invoke WRITER to write data to an output file.

```
K      = K + 1
IARC(K) = SINK
JARC(K) = SRCE
LOWR(K) = IEXP
UPPR(K) = IPRD
WRITE (U6,610) IPRD, IEXP
IF ( IEXP .GT. IPRD ) WRITE (U6,620)
ARCS = K
WRITE (U6,650) SINK, SRCE, NODE, ARCS
CALL WRITER ( ARCS, NODE, IARC, JARC, LOWR, UPPR, KOST )
RETURN
END
```

C-----  
C ===== SURPLS =====  
C ---- SURPLS is used to find out all the details relating to  
C ---- the surplus regions such as the costs of transportation, and  
C ---- storage from each surplus region to all the river regions,  
C ---- port regions, barge locations and barge (river) loading points.  
C ---- Invoke subroutine, GENARC to generate the appropriate arcs  
C ---- connecting the nodes.

```
SUBROUTINE SURPLS ( IPRD )
COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /A3/ SRND, REND, DRND, PEND, FRND, DMND
COMMON /A4/ SINK, SRCE
COMMON /B1/ SRGN(65) /B2/ DRGN(65) /B3/ RIVR(45)
COMMON /B4/ PORT(20)
COMMON /C1/ SDTR(65,65) /C2/ SDRL(65,65)
COMMON /C3/ SRTR(65,45) /C4/ SRRL(65,45)
COMMON /C5/ SPTR(65,20) /C6/ SPRL(65,20)
COMMON /D1/ RAIL, TRUCK, BARGE, SHIP, PERIOD, CHOICE
COMMON /E1/ SPLY(65)
COMMON /F1/ STOR(65) /F2/ SCST(4)
COMMON /F3/ TLOS, RLOS, TLOR, RLOR, BLOR, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION
COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G3/ LOWR(27500)
COMMON /G4/ UPPR(27500) /G5/ KOST(27500)
COMMON /H1/ IDEN(65), ALFA(65), BETA(65)
INTEGER     SRND, REND, DRND, PEND, FRND, DMND
INTEGER     SINK, SRCE
INTEGER     SRGN, DRGN, RIVR, PORT, FRGN
INTEGER     UPPR
INTEGER U1,U2,U3,U4,U8,U9,U6,U12

500 FORMAT ( 20A4 )
520 FORMAT ( 10F8.3 )
610 FORMAT (6X 'SUPPLY', F15.0, )
```

```

820 FORMAT (6X, 'UNIT = 1 (THOUSAND BUSHEL)', /)
830 FORMAT (6X, 'UNIT = 2 (THOUSAND SHORT TON )', /)
840 FORMAT (6X, 'UNIT = 3 (THOUSAND METRIC TON)', /)

C ---- Read in the amount of grain produced by and the storage
C ---- capacity of each surplus region.

      READ  (U8,520) ( SPLY(I), I = 1, NOSR )
      READ  (U1,520) ( STOR(I), I = 1, NOSR )

C ---- Read in the cost, rail, truck, barge and ship factors; and
C ---- the period when the lakes are blocked.

      READ  (U1,520) COST, RAIL, TRUCK, BARGE, SHIP, PERIOD

C ---- Read in the conversion factor and the choice of measurement
C ---- used. Display the appropriate message.

      READ  (U1,520) OPERATION, CHOICE
      IF (CHOICE .EQ. 1.000) WRITE(17,820)
      IF (CHOICE .EQ. 2.000) WRITE(17,830)
      IF (CHOICE .EQ. 3.000) WRITE(17,840)

C ---- Calculate the storage cost for each period, SCST(N).
C ---- There are NOTP periods and the cost is COST.

      IF ( NOTP .LE. 0 ) GO TO 1200
      DO 1100 N = 1, NOTP
      SCST(N) = COST * FLOAT(NDAY(N)) * 1000.0 / 365.0
1100 CONTINUE
1200 CONTINUE

C ---- Read in the mileage (by truck, TR) from each surplus region
C ---- to all the deficit regions .

      DO 1300 I = 1, NOSR
      READ  (U2,520) ( SDTR(I,J), J = 1, NODR )
1300 CONTINUE

C ---- Read in the rail costs (RL) from each surplus region to all
C ---- the deficit regions.

      DO 1400 I = 1, NOSR
      READ  (U3,520) ( SDRL(I,J), J = 1, NODR )
1400 CONTINUE

C ---- Read in the mileage (by truck, TR) from each selected (river)
C ---- barge loading location linked with all the surplus regions.

      READ  (U2,520) ( SRTR(I,1), I = 1, NOSR )

C ---- Read in the rail costs(RL) from each surplus region to all
C ---- the river regions.

      DO 1600 I = 1, NOSR

```

```

      READ  (U3,520) ( SRRL(I,J), J = 1, MORE )
1600 CONTINUE

C ---- Read in the mileage (by truck, TR) from each surplus region
C ---- to all the port regions.

      DO 1700 I = 1, NOSR
      READ  (U2,520) ( SPTR(I,J), J = 1, NOPE )
1700 CONTINUE

C ---- Read in the rail costs(RL) from each surplus region to
C ---- all the port regions.

      DO 1800 I = 1, NOSR
      READ  (U3,520) ( SPRL(I,J), J = 1, NOPE )
1800 CONTINUE

C ---- Read in the alfas and betas of the surplus regions.
C ---- Currently not used.

      READ  (U1,520) ( ALFA(I), BETA(I), I = 1, NOSR )

C ---- Read in the selected barge (river) points which linked with
C ---- the surplus regions.

      READ  (U1,500) ( IDEN(I), I = 1, NOSR )

C ---- Find the amount produced, and storage capacity of each of
C ---- the surplus region.

      DO 4000 I = 1, NOSR
      ALF      = ALFA(I)
      BET      = BETA(I)
      NF       = I
      IA       = NOTF * ( NF - 1 )
      K        = K + 1
      IARC(K)  = SRCE
      JARC(K)  = IA + 1
      UPPR(K)  = SPLY(NF) * 1000.0
      IPRD    = IPRD + UPPR(K)
      ISTR     = STOR(NF)
      IF ( NOTP .LE. 0 ), GO TO 2200

C ---- Find the storage cost(SCST) and amount produced
C ---- per quarter for each surplus region.

      DO 2100 N = 1, NOTP
      K        = K + 1
      IARC(K)  = JARC(K-1)
      JARC(K)  = IARC(K) + 1
      UPPR(K)  = ISTR
      KOST(K)  = SCST(N)
2100 CONTINUE
2200 CONTINUE

```

C ---- Find the mileage(by truck) and calculate the cost by truck.

```
DO 2400 M = 1, NODR
NT      = M
JA      = NOTF * ( NT - 1 ) + SRND
```

C ---- Check that the TRUCK mileage from the surplus region  
C ---- to the deficit region is feasible.  
C ---- If the mileage is greater than 9999 then it is ignored else  
C ---- the total cost includes the loading and unloading cost  
C ---- is calculated.

```
COST      = SDTR(NF,NT)
IF ( COST .LE. 225 ) THEN
COST = ( 0.066374 + 0.104892 * COST )
ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
COST      = ( 23.67 + 0.0 * COST )
ELSE IF ( COST .GT. 245 ) THEN
COST      = ( 0.68037 + 0.093976 * COST )
ELSE IF ( COST .GE. 9999. ) THEN
GOTO 2300
ENDIF
```

```
COST      = ( COST * TRUCK + TL0S + TRID ) * 1000.0
```

```
CALL GENARC ( IA, JA, COST )
```

```
2300 CONTINUE
```

C ---- Check that the Rail cost from the surplus region to  
C ---- the deficit region is feasible.  
C ---- If this cost is greater than 999 then it is ignored else  
C ---- the cost (including loading and unloading) is found.

```
COST      = SDRL(NF,NT)
IF ( COST .GE. 999. ) GO TO 2400
COST      = ( COST * RAIL + RL0S + RRID ) * 1000.0
CALL GENARC ( IA, JA, COST )
2400 CONTINUE
2500 CONTINUE
```

C ---- Invoke the subroutine SERIAL which  
C ---- check that all inputted data are corrected.

```
IDNT      = IDEN(1)
CALL SERIAL ( IDNT, NORE, RIVR, NT )
IF ( NT .EQ. 0 ) GO TO 2600
JA      = NOTF * ( NT - 1 ) + DRND
```

C ---- Find the mileage (by truck) from surplus region to the river  
C ---- regions. If it is greater than 9999 then it is ignored  
C ---- else the total cost by truck is found.

```
COST      = SRTR(NF,1)
IF ( COST .LE. 225 ) THEN
COST = ( 0.066374 + 0.104892 * COST )
```

```

ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
COST      = ( 23.67 + 0.0 * COST )
ELSE IF ( COST .GT. 245 ) THEN
COST      = ( 0.68037 + 0.093976 * COST )
ELSE IF ( COST .GE. 9999. ) THEN
GOTO 2600
ENEIF

COST      = ( COST * TRUCK + TLOS + TRIO ) * 1000.0

CALL GENARC ( IA, JA, COST )
2600 CONTINUE

C ---- Find the rail cost from the surplus region to
C ---- the river regions. If it is greater than 999 then
C ---- it is ignored else the cost by rail is found.

00 2800 M = 1, NOR
NT      = M
JA      = NOTF * ( NT - 1 ) + ORND
COST    = SRRRL(NF,NT)
IF ( COST .GE. 999. ) GO TO 2700
COST    = ( COST * RAIL + RLLOS + RRIR ) * 1000.0
CALL GENARC ( IA, JA, COST )
2700 CONTINUE
2800 CONTINUE

C ---- Find the mileage(by truck) from the surplus region to
C ---- the port regions. If it is greater than 9999 then
C ---- it is ignored else the total cost by truck is found.

00 3000 M = 1, NOPE
NT      = M
JA      = NOTF * ( NT - 1 ) + RENO
COST    = SPTR(NF,NT)
IF ( COST .LE. 225 ) THEN
COST = ( 0.066374 + 0.104892 * COST)
ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
COST      = ( 23.67 + 0.0 * COST )
ELSE IF ( COST .GT. 245 ) THEN
COST      = ( 0.68037 + 0.093976 * COST )
ELSE IF ( COST .GE. 9999. ) THEN
GOTO 2900
ENEIF

COST      = ( COST * TRUCK + TLOS + TRIO ) * 1000.0

CALL GENARC ( IA, JA, COST )
2900 CONTINUE

C ---- Find the rail cost from the surplus region to
C ---- the port regions. If it is greater than 999 then
C ---- it is ignored else the cost by rail is found.

COST    = SPRL(NF,NT)

```

```

        IF ( COST .GE. 999. ) GO TO 3000
        COST      = ( COST * RAIL + RLOS + RRIP ) * 1000.0
        CALL GENARC ( IA, JA, COST )
3000 CONTINUE
3900 CONTINUE
4000 CONTINUE

C ---- Display the total amount of grain produced into an intermediate
C ---- file (FILE06_2.SFT).

        WRITE(17,610) (IPRD/OPERATION)
        RETURN
        END

C-----
C      ===== RELVTR =====
C ----- RELVTR deals with all the river regions.
C ----- RELVTR is used to find the transportation costs from river
C ----- regions.
C ----- It calculates the truck & rail cost and choose the minimum
C ----- cost and invokes subroutine GENARC to generate an arc with
C ----- this minimum cost.

SUBROUTINE RELVTR
COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /A3/ SRND, REND, DRND, PEND, FRND, DMND
COMMON /B2/ DRGN(65) /B3/ RIVR(45)
COMMON /B4/ PORT(20)
COMMON /C1/ RDTR(65,65) /C2/ RDRL(65,65)
COMMON /C3/ RRBG(65,45) /C5/ RPBG(65,20)
COMMON /D1/ RAIL, TRUCK, BARGE, SHIP
COMMON /F3/ TLOS, RLOS, TLOR, RLOR, BLOR, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION
COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G3/ LOWR(27500)
COMMON /G4/ UPPR(27500) /G5/ KOST(27500)
COMMON /H1/ IDEN(65), ALFA(65), BETA(65)
DIMENSION IDN1(10), IDN2(10), IDN3(25)
INTEGER SRND, REND, DRND, PEND, FRND, DMND
INTEGER DRGN, RIVR, PORT, UPPR
INTEGER U1,U2,U3,U4,U8,U9,U6,U12
LOGICAL ICED
500 FORMAT ( 20A4 )
510 FORMAT ( 20I4 )
520 FORMAT ( 10F8.3 )

C ---- Read in the alfas and betas of the deficit regions.
C ---- Read in the selected shipping(port) points which linked with
C ---- deficit regions.

        READ (U1,520) ( ALFA(I), BETA(I), I = 1, NODR )
        READ (U1,500) ( IDEN(I), I = 1, NODR )

```

```

C ---- Read in the total number of selected barge (river) points
C ---- and barge (port) points, and their respective codes.

      READ  (U1,510) NRES, NPES
      READ  (U1,500) ( IDN1(I), I = 1, NRES ), ( IDN2(I), I = 1, NPES )

C ---- Read in the mileage (by TRuck) of selected barge unloading
C ---- locations linked with each of the deficit region.

      READ  (U2,520) ( RDTR(I,J), J = 1, NODR )

C ---- Read in the rail (RL) costs of each river region
C ---- linked with all the deficit regions.

      DO 1200 I = 1, NOR
      READ  (U3,520) ( RDRL(I,J), J = 1, NODR )
1200 CONTINUE

C ---- Read in the barge (BG) cost (per bushel) from each river
C ---- region linked with all of the selected barge (river)
C ---- shipping points.

      DO 1300 I = 1, NOR
      READ  (U4,520) ( RRBG(I,J), J = 1, NRES )
1300 CONTINUE

C ---- Read in the barge (BG) cost (per bushel) from each river
C ---- region linked with all the selected barge (port) shipping
C ---- points.

      DO 1400 I = 1, NOR
      READ  (U4,520) ( RPBG(I,J), J = 1, NPES )
1400 CONTINUE

C ---- Read in the number of river locations above the L&D 26
C ---- and their codes.

      READ  (U4,510) LAKE
      READ  (U4,500) ( IDN3(I), I = 1, LAKE )

C ---- Set all the variables to the appropriate values
C ---- and call SERIAL to check that all the required
C ---- information is correctly inputted.

      DO 3000 I = 1, NOR
      NF      = I
      IDNT    = RIVR(I)
      ICED    = .FALSE.
      CALL SERIAL ( IDNT, LAKE, IDN3, NT )
      IF ( NT .NE. 0 ) ICED = .TRUE.
      IA      = NOTF * ( NF - 1 ) + DRND

C ---- Find the truck (TR) cost linking each river region with
C ---- all the deficit regions.

```

```

DO 2200 M = 1, NODR
IDNT      = IDEN(M)
CALL SERIAL ( IDNT, MORE, RIVR, NT )
IF ( NF .NE. NT ) GO TO 2200
NT        = M
ALF       = ALFA(NT)
BET       = BETA(NT)

C ---- If the truck cost is greater than 9999 then this cost
C ---- is ignored else the total truck cost is calculated
C ---- by taking into account of the respective alfa and beta,
C ---- and the loading (TLOR) and unloading (TRID) costs.

COST      = RDTR(1,NT)
IF ( COST .LE. 225 ) THEN
COST = ( 0.066374 + 0.104892 * COST)
ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
COST = ( 23.67 + 0.0 * COST )
ELSE IF ( COST .GT. 245 ) THEN
COST = ( 0.68037 + 0.093976 * COST )
ELSE IF ( COST .GE. 9999. ) THEN
GOTO 2200
ENDIF

JA        = NOTF * ( NT - 1 ) + SRND
COST      = ( COST * TRUCK + TLOR + TRID ) * 1000.0

CALL GENARC ( IA, JA, COST )
2200 CONTINUE

C ---- Find the rail cost from each river region linked
C ---- with all the deficit regions.

DO 2300 M = 1, NODR
NT        = M
JA        = NOTF * ( NT - 1 ) + SRND

C ---- If the rail cost is more than 999 than the cost is ignored
C ---- else the cost by rail is calculated.
C ---- RAIL is the rail factor, RLOR is the loading cost and
C ---- RRID is the unloading cost.

COST      = RDRL(NF,NT)
IF ( COST .GE. 999. ) GO TO 2300
COST      = ( COST * RAIL + RLOR + RRID ) * 1000.0
CALL GENARC ( IA, JA, COST )
2300 CONTINUE

C ---- The barge cost(per bushel) is calculated for each of the chosen
C ---- barge(river) points.

DO 2500 M = 1, NRES
IDNT      = IDN1(M)
CALL SERIAL ( IDNT, MORE, RIVR, NT )

```

```

IF ( NT .EQ. NF ) GO TO 2500
IF ( NT .EQ. 0 ) GO TO 2500
JA      = NOTF * ( NT - 1 ) + DRND

C ---- If the cost by barge is greater than 999 then it is ignored.
C ---- If the river is iced then UPPR is set to 0, i.e., no passage
C ---- is possible thus no grain is transported.
C ---- BARGE is a barge ratio which can be altered (see data file F1.SFT).

COST      = RRBG(NF,M) * 100.0
IF ( COST .GE. 999. ) GO TO 2500
COST      = ( COST * BARGE + BLOR + BRIR ) * 1000.0
CALL GENARC ( IA, JA, COST )
IF ( ICED ) UPPR(K-NOTP+(PERIOD - 1)) = 0
2500 CONTINUE

C ---- Find the barge cost for each selected barge(port) point
C ---- with all the river points.

DO 2800 M = 1, NPES
IDNT      = IDN2(M)
CALL SERIAL ( IDNT, NOPE, PORT, NT )
IF ( NT .EQ. 0 ) GO TO 2800
JA      = NOTF * ( NT - 1 ) + REND

C ---- If the barge cost is larger than 999 then it is ignored
C ---- else the total cost is calculated which includes the
C ---- loading(BLOR) and unloading (BRIP) costs.

COST      = RPBG(NF,M) * 100.0
IF ( COST .GE. 999. ) GO TO 2800
COST      = ( COST * BARGE + BLOR + BRIP ) * 1000.0
CALL GENARC ( IA, JA, COST )
IF ( ICED ) UPPR(K-NOTP+(PERIOD - 1)) = 0
2800 CONTINUE
2900 CONTINUE
3000 CONTINUE
RETURN
END

C-----
C      ===== PELVTR =====
C ---- PELVTR deals with all the data relating to port regions.
C ---- PELVTR is used to find transportation cost from the port
C ---- regions. The truck and rail costs are calculated.
C ---- If the cost is not feasible, then it is ignored.
C ---- The variable, ICED is a boolean variable. It is used to
C ---- denote whether the river is iced or not. If it is iced,
C ---- then ICED is set to true which means that the river is
C ---- blocked and no passage is possible.

SUBROUTINE PELVTR
COMMON /A1/ NOSR, NODR, MORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)

```

```

COMMON /A3/ SRND, REND, DRND, PEND, FRND, DMND
COMMON /B4/ PORT(20) /B5/ FRGN(25)
COMMON /C3/ PFSP(65,45)
COMMON /D1/ RAIL, TRUCK, BARGE, SHIP
COMMON /F3/ TLOS, RLDS, TLOR, RLOR, BLOR, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION
COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G3/ LOWR(27500)
COMMON /G4/ UPPR(27500) /G5/ KOST(27500)
DIMENSION IDN3(16)
INTEGER SRND, REND, DRND, PEND, FRND, DMND
INTEGER PORT, FRGN, UPPR
INTEGER U1,U2,U3,U4,U8,U9,U6,U12
LOGICAL ICED
500 FORMAT ( 20A4 )
510 FORMAT ( 20I4 )
520 FORMAT ( 10F8.3 )

C ---- Read in the shipping costs of each port location linked
C ---- with all the foreign regions.

DO 1100 I = 1, NOPE
READ (U4,520) ( PFSP(I,J), J = 1, NOFR )
1100 CONTINUE

C ---- Read in the number of lakes used for export and their
C ---- code names.

READ (U4,510) LAKE
READ (U4,500) ( IDN3(I), I = 1, LAKE )

C ---- Set ICED to false which means that the port is passable.
C ---- Call SERIAL to check that all the necessary information
C ---- is included.

DO 2000 I = 1, NOPE
NF      = I
ICED    = .FALSE.
IDNT    = PORT(I)
CALL SERIAL ( IDNT, LAKE, IDN3, NT )
IF ( NT .NE. 0 ) ICED = .TRUE.
IA      = NOTF * ( NF - 1 ) + REND

C ---- Find the port (ship rates) cost of all the foreign regions
C ---- and generate the corresponding arcs.
C ---- Reset the value of UPPR if the port is ICED.

DO 1300 M = 1, NOFR
NT      = M
JA      = NOTF * ( NT - 1 ) + PEND

C ---- If the ship cost is greater than 999, it is ignored,
C ---- else the total cost (including shipping rates) is found.
C ---- The condition of the port is checked, if it is iced then
C ---- no grain is transported, UPPR is set to 0.

```

```
C ---- SHIP is the ship ratio which can be altered (refer to F1.SFT).
```

```
COST      = PFSP(NF,NT) * 100.0
IF ( COST .GE. 999. ) GO TO 1300
COST      = ( COST * SHIP + SLOP ) * 1000.0
CALL GENARC ( IA, JA, COST )
IF ( ICED ) UPPR(K-NOTP+(PERIOD - 1)) = 0
1300 CONTINUE
2000 CONTINUE
RETURN
END
```

```
C -----
```

```
C           ===== DEMAND =====
C ---- DEMAND deals with all the amount of grain produced
C ---- from surplus regions and the amount of grain demanded by
C ---- deficit regions.
C ---- Read in the demand required by the deficit and foreign regions.
C ---- Calculate the demand (per quarter) for the deficit and
C ---- foreign regions.
C ---- Find the total amount of grain supplied (UPPR)
C ---- and demanded (LOWR).
```

```
SUBROUTINE DEMAND ( IEXP )
COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /A3/ SRND, REND, DRNO, PEND, FRND, DMND
COMMON /A4/ SINK, SRCE
COMMON /E2/ DDND(65) /E3/ FDND(25,4)
COMMON /F3/ TLDS, RLDS, TLOR, RLOR, BLOR, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION
COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G3/ LOWR(27500)
COMMON /G4/ UPPR(27500) /G5/ KOST(27500)
INTEGER     SRND, REND, DRND, PEND, FRND, DMND
INTEGER     SINK, SRCE
INTEGER     UPPR
INTEGER     U1,U2,U3,U4,U8,U9,U6,U12
520 FORMAT ( 10F8.3 )
610 FORMAT (6X, 'DEMAND', F15.0, /)
```

```
    IF ( NODR .LE. 0 ) GO TO 1400
```

```
C ---- Read in the amount of grain demanded by the deficit regions.
```

```
READ  (U8,520) ( DDND(I), I = 1, NODR )
```

```
C ---- Find the amount demanded (in each quarter) in each of the
C ---- deficit region.
```

```
DO 1300 I = 1, NODR
IA      = NOTF * ( I - 1 ) + SRND
JA      = FRND
DO 1200 N = 1, NOTF
K      = K + 1
```

```

IARC(K) = IA + N
JARC(K) = JA + I
LOWR(K) = DDND(I) * 250.0
IEXP = IEXP + LOWR(K)

1200 CONTINUE
K = K + 1
IARC(K) = JA + I
JARC(K) = SINK

1300 CONTINUE

C ---- Read in the amount of grain demanded by the foreign regions.

1400 IF ( NOFR .LE. 0 ) RETURN
DO 1500 I = 1, NOFR
READ (U8,520) ( FDND(I,N), N = 1, NOTF )
1500 CONTINUE

C ---- Find the demand (in each quarter) by each foreign region.

DO 1700 I = 1, NOFR
IA = NOTF * ( I - 1 ) + PEND
JA = FRND + NODR
DO 1600 N = 1, NOTF
K = K + 1
IARC(K) = IA + N
JARC(K) = JA + I
LOWR(K) = FDND(I,N) * 1000.0
IEXP = IEXP + LOWR(K)

1600 CONTINUE

K = K + 1
IARC(K) = JA + I
JARC(K) = SINK

1700 CONTINUE

C ---- Display the total amount of grain demanded into an intermediate
C ---- file (FILE06_2.SFT).

      WRITE(17,610) (IEXP/OPERATION)
      RETURN
      END

C -----
C           === SERIAL ===
C ---- Check that all data are inputted correctly. An error message
C ---- will be outputted if there is insufficient data.

SUBROUTINE SERIAL ( IDNT, NOSR, SRGN, NF )
DIMENSION SRGN(NOSR)
INTEGER SRGN
600 FORMAT ( 5X, '????? ERROR IN DATA. ', A4, 'IS MISSING' )
NF = 0
DO 1100 I = 1, NOSR
IF ( IDNT .EQ. SRGN(I) ) GO TO 1200
1100 CONTINUE

```

```
      RETURN
1200 NF      = I
      RETURN
      END

C -----
C           ===== GENARC =====
C ---- Generate the arcs for each node.
C ---- Arcs are generated for each quarter.
C ---- The total cost of transportation is stored in KOST.

      SUBROUTINE GENARC ( IA, JA, COST )
      COMMON /A2/ NOTP, NOTF, NDAY(4)
      COMMON /GO/ K, OPERATION
      COMMON /G1/ IARC(27500) /G2/ JARC(27500) /G5/ KOST(27500)
      DO 1100 N = 1, NOTF
      K      = K + 1
      IARC(K) = IA + N
      JARC(K) = JA + N
      KOST(K) = COST
1100 CONTINUE
      RETURN
      END
```

```
C ---- ****  
C ---- * SOFT2.F *  
C ---- * DOCUMENTED ON : SEPT 1990 *  
C ---- * RUN ON : A 386 machine *  
C ---- * COMPILER USED : NDP Fortran *  
C ---- * GRAIN USED : SOFT WHEAT *  
C ---- * DATA FILES USED : U12 *  
C ---- * TRACE FILE : U6 *  
C ---- * INPUT FILE FOR SOFT3.F : UF1 *  
C ---- ****
```

```
C ---- Declaration of all variables used.
```

```
COMMON /AA/ NR, NN, FSBL, NTIM, TOTL, MAXA  
COMMON /B1/ IWK(1500) /B2/ LABL(1500) /B3/ NODE(1500)  
COMMON /B4/ MIDL(1500) /B5/ NSAVE(1500)  
COMMON /C1/ ILO(34000) /C2/ ISAVE(34000) /C3/ JSAVE(34000)  
COMMON /C4/ JWV(34000)  
COMMON /D1/ KOS(68000) /D2/ MIR(68000) /D3/ NA(68000)  
COMMON /D4/ NC(68000) /D5/ NF(68000)  
LOGICAL FSBL  
INTEGER U6,U12,UF1  
610 FORMAT (' ***** OPTIMAL SOLUTION ***** ')  
620 FORMAT (' ?????? INFEASIBLE SOLUTION ?????? ')  
630 FORMAT (' TOTAL COST ', F15.0 )
```

```
C ---- Unit number for each file.  
C ---- U6 is the unit number used for the temporary file  
C ---- which is used to trace/check that the output from  
C ---- this program.  
C ---- U12 is the unit number for the output file created by  
C ---- the first(previous) program, SOFT1.  
C ---- This data file contains the relevant information which  
C ---- is necessary for SOFT2 to run.  
C ---- UF1 is the unit number for the output file of this program  
C ---- This output file will be used in the third program, SOFT3.
```

```
U6 = 17  
U12 = 12  
UF1 = 18
```

```
C ---- To open files FILE06_2, FLOW1_2, FILE12_2  
C ---- for reading and writing.
```

```
OPEN (UNIT = UF1, FILE = 'FLOW1_2.SFT', STATUS = 'UNKNOWN',  
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')  
OPEN (UNIT = U6, FILE = 'FILE06_2.SFT', STATUS = 'UNKNOWN',  
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')  
OPEN (UNIT = U12, FILE = 'FILE12_2.SFT', STATUS = 'UNKNOWN',
```

```

1      ACCESS = 'SEQUENTIAL', FORM = 'UNFORMATTED')

NTIM      = 0
FSBL      = .TRUE.
MAXA      = 34000

C ---- Read in the data from the output file(generated by) SOFT1.F.
C ---- The data read in will be used in this current program.

      READ  (U12,100) NR,NN,( NF(K), K = 1, NR ),( NA(K), K = 1, NR ),
1          ( ILO(K), K = 1, NR ), ( JSAVE(K), K = 1, NR ),
2          ( ISAVE(K), K = 1, NR )
100   FORMAT(2018)

      DO 1100 K = 1, NR
      NC(K)    = 0
      KOS(K)   = 0
1100 CONTINUE
      CALL SUPERK
      IF ( .NOT. FSBL ) WRITE (U6,620)
      IF ( FSBL ) WRITE (U6,610)
      CSUM     = 0
      DO 1500 K = 1, NR
      COST     = FLOAT ( ISAVE(K) )
      GRAN    = FLOAT ( NC(K) )
      CSUM     = CSUM + COST * GRAN
1500 CONTINUE
      CSUM     = CSUM / 10.0

C ---- Write the optimal solution showing the total cost.
C ---- This information is only for checking purpose and
C ---- will not be used any further.

      WRITE (U6,630) CSUM

C ---- Write the solution to an output file, FILE12_2_grn
C ---- and this is used by the third and last program PROG03.
C ---- Note: Format used might not be right. Changes will have
C ---- to be made.

      WRITE (UF1,2000) NR, ( NC(K), K = 1, NR ), ( KOS(K), K = 1, NR )
2000 FORMAT (2018)
      ENDFILE UF1
      STOP
      END

```

```

C-----  

C          ===== SUPERK =====  

SUBROUTINE SUPERK
COMMON /AA/ NR, NN, FSBL, NTIM, TOTL, MAXA
COMMON /B1/ IWF(1500) /B2/ LABL(1500) /B3/ NODE(1500)
COMMON /B4/ MIDL(1500) /B5/ NSAVE(1500)
COMMON /C1/ ILO(34000) /C2/ ISAVE(34000) /C3/ JSIZE(34000)
COMMON /C4/ JWF(34000)
COMMON /D1/ KOS(68000) /D2/ MIR(68000) /D3/ NA(68000)
COMMON /D4/ NC(68000) /D5/ NF(68000)
LOGICAL      FSBL
1234 CONTINUE
INFIN=100000000
IFLOW=0
KLAB=0
KPOT=0
KBRK=0
IP=0
NUMS=0
NONS=0
IPL=0
NR2=NR*2
NN1=NN+1
IF ( NTIM .GE. 1 ) GO TO 12
DO 5 I=1,NN1
NODE(I)=0
5   LABL(I)=0
DO 10 M=1,MR
I=NF(M)
J=NA(M)
IFLOW=NC(M)
KOST=ISAVE(M)
NODE(I)=NODE(I)+1
NODE(J)=NODE(J)+1
N=M+NR
NF(N)=J
NA(N)=I
KOS(M)=KOST
KOS(N)=-KOST
NC(M)=JSIZE(M)-IFLOW
NC(N)=IFLOW-ILO(M)
10  CONTINUE
DO 11 I=1,NN1
11  NSAVE(I)=NODE(I)
GO TO 1401
12  DO 13 I=1,NN1
NODE(I)=NSAVE(I)
13  LABL(I)=0

```

```

DO 14 M=1,NR
N=M+NR
I = NF(M)
J = NA(M)
NF(N) = J
NA(N) = I
IFLOW=NC(M)
KOST=ISAVE(M)+KOS(M)
KOS(M)=KOST
KOS(N)=-KOST
NC(M)=JSAVE(M)-IFLOW
NC(N)=IFLOW-ILO(M)
14  CONTINUE
1401 CONTINUE
C ****
C
C      SETUP SECTION
C
C ****
KL=1
DO 15 K=1,NN1
JK=NODE(K)
NODE(K)=KL
JWV(K)=KL
KL=JK+KL
15  MIDL(K)=KL-1
DO 20 L=1, NR
LL=L+NR
J=NA(L)
I=NA(LL)
KOST=KOS(L)
K=NC(L)
LO=-NC(LL)
C      RIGHT=2    LEFT=1
MAIN=2
MIRROR=2
IF(KOST) 29,29,30
29  IF(K)32,32,31
30  IF(LO)35,36,31
31  MAIN=1
32  IF(KOST) 33,34,34
33  IF(K) 35,36,36
34  IF(LO) 35,36,36
35  MIRROR=1
36  GO TO(43,44),MAIN
43  II=JWV(I)
MIR(II)=L
JWV(I)=II+1
GO TO 45
44  II=MIDL(I)
MIR(II)=L
MIDL(I)=II-1
45  GO TO(46,47),MIRROR
46  II=JWV(J)
MIR(II)=LL

```

```

        JMV(J)=II+1
        GO TO 20
47    II=MIDL(J)
        MIR(II)=LL
        MIDL(J)=II-1
20    CONTINUE
C ****
C
C     GO - SUPERKILTER
C
        ND=INFIN
C
C     MAIN LOOP (100)
C
        NR2=NR*2
        DO 1000 MAIN=1, NR
        MAINM=MAIN+NR
        DO 1000 MODE=1, 2
        GO TO(52, 53), MODE
52    II=MAIN
        JZ=MAINM
        GO TO 54
53    II=MAINM
        JZ=MAIN
54    IF(NC(II)) 65, 55, 56
55    IF(NC(JZ)) 63, 990, 990
56    IF(KOS(II)) 63, 55, 55
C           IS, IT = START, END NODE NOS, JS, JT = ARC, MIRROR ARC NOS
C           FOR ARC NEEDING FLOW INCREASE
C           WANT TO INCREASE FLOW, START LABELING AT JJ
63    IS=NA(JZ)
        JS=II
        IT=NA(II)
        JT=JZ
        GO TO 70
C           WANT TO DECREASE FLOW, START LABELING AT II
65    IT=NA(JZ)
        IS=NA(II)
        JS=JZ
        JT=II
C
C     LABELING PROCEDURE
C
C ****
70    IPL=1
        IPLL=1
        IPS=0
        NUMS=0
        LABL(IT)=JS
        IWF(IPL)=IT
84    KLAB=KLAB+1
        GO TO 86
85    IF(IPS-IPL)86, 200, 86
86    IPS=IPS+1
        IA=IWF(IPS)

```

```

IB=NODE(IA)
IE=MIDL(IA)
IF(IB-IE) 87,87,85
87 DO 90JJ=IB,IE
J=MIR(JJ)
NUNODE=NA(J)
IF(LABL(NUNODE)) 90,88,90
88 LABL(NUNODE)=J
IPL=IPL+1
JWV(IPL)=NUNODE
IF(NUNODE-IS) 90,96,90
90 CONTINUE
GO TO 85

C
C      BREAKTHROUGH      BREAKTHROUGH      BREAKTHROUGH
C
96 KBRK=KBRK+1
97 IALPHA=INFIN

C
C      FIRST RETRACE
C
C          IJ = PREDECESSOR ARC INDEX
C          JI = MIRROR ARC INDEX
C          K = JWV POINTER
C          NEXT = PREDECESSOR NODE
C
K=0
NOW=IS
100 IJ=LABL(NOW)
JI=IJ-NR
IF(JI) 101,101,102
101 JI=JI+NR2
102 NEXT=NA(JI)
K=K+1
IF(KOS(IJ)) 105,105,104
104 NET=-NC(JI)
JWV(K)=NET
GO TO 110
105 NET=NC(IJ)
JWV(K)=NET
110 IALPHA=MIN0(IALPHA,NET)
IF(NEXT-IS) 111,120,111
111 NOW=NEXT
GO TO 100
C      SECOND RETRACE
C
120 K=0
NOW=IS
125 IJ=LABL(NOW)
JI=IJ-NR
IF(JI) 126,126,127
126 JI=JI+NR2
127 NEXT=NA(JI)
K=K+1
NC(IJ)=NC(IJ)-IALPHA

```

```

NET=NC(JI)
NETNU=NET+IALPHA
NC(JI)=NETNU
IF(KOS(JI)) 128,1271,128
1271 IF(NET) 1272,1272,128
1272 IF(NETNU) 128,128,1273
1273 CALL LEFT(NOW,JI)
128 IF(JWV(K)-IALPHA) 129,1281,129
1281 CALL RIGHT(NEXT,IJ)
129 IF(NEXT-IS) 130,150,130
130 NOW=NEXT
GO TO 125

C
C      ERASE LABELS AND GO FOR O-K CHECK
C
150 DO 155 I=1,IPL
J=JWV(I)
155 LABL(J)=0
GO TO 54

C
C      POTENTIAL CHANGE
C
200 KPOT=KPOT+1
201 KSET=NUMS
NEWLAB=0
NUMS=0
IMTHR=0
MIN=INFIN
NEW=NONS
NONS=MAXA+1
IF(KSET) 204,204,202
202 IF(NEW-MAXA) 295,295,312
C      NON-S (L,L-) SET RECYCLING FILTER
295 MAXNEW=MAXA+NEW
DO 310 L=NEW,MAXA
K=MAXNEW-L
KK=JWV(K)
KKK=NA(KK)
IF(LABL(KKK)) 310,300,310
300 NONS=NONS-1
JWV(NONS)=KK
310 CONTINUE
C      S-SET RECYCLING FILTER
312 DO 203 K=1,KSET
KK=JWV(K)
KKK=NA(KK)
IF(LABL(KKK)) 203,2021,203
2021 IF(KOS(KK)) 2023,2023,2022
2022 NUMS=NUMS+1
JWV(NUMS)=KK
MIN=MIND(MIN,KOS(KK))
GO TO 203
2023 NONS=NONS-1
JWV(NONS)=KK
203 CONTINUE

```

```

204  CONTINUE
    IF(IPLL-IPL) 2039,2039,2111
C      FIND MIN(C-BAR) OVER SET S
2039 DO 211 LL=IPLL,IPL
    L=IWV(LL)
    JMID=MIDL(L)+1
    JRT=NODE(L+1)-1
    IF(JMID-JRT) 2045,2045,211
2045 DO 210 KK=JMID,JRT
    K=MIR(KK)
    I=NA(K)
    IF(LABL(I)) 210,2040,210
2040 IF(NC(K)) 206,2041,2041
2041 IF(KOS(K)) 206,206,205
205  NUMS=NUMS+1
    JWV(Nums)=K
    MIN=MIN0(MIN,KOS(K))
    GO TO 210
206  NONS=NONS-1
    JWV(NONS)=K
210  CONTINUE
211  CONTINUE
2111 IPLL=IPL+1
    IF(NUMS) 212,212,215
212  FSBL      = .FALSE.
    CALL DUMPO(NR,II)
    PRINT 2125,IS,IT,II
    IF (.NOT. FSBL) RETURN
    PRINT 2121,(I,LABL(I),I=1,NN)
    PRINT 2122,(I,IWV(I),I=1,IPL)
    PRINT 2123,(JWV(I),I=NEW,MAXA)
2121 FORMAT(' LABELS, BY NODE'/(5(19,'=',I10)) )
2122 FORMAT(' LABELED NODES (IWV)'/(10I10))
2123 FORMAT(' THE SET (L,L-), NON-S'/(10I10))
2125 FORMAT(' OIS=',15,' IT=',15,10X,'INFEASIBLE ARC =',15)
    RETURN
C      UPDATE RELATIVE COSTS
C
C      UPDATE COST FOR SET S
215  DO 230 I=1,NUMS
    IJ=JWV(I)
    JI=IJ-NR
    IF(JI) 216,216,217
216  JI=IJ+NR
217  KOST=KOS(IJ)-MIN
    KOS(IJ)=KOST
    KOS(JI)=-KOST
    IF(KOST) 230,218,230
218  IF(NC(IJ)) 230,230,220
220  NODEB=NA(IJ)
    CALL LEFT(NA(JI),IJ)
    IF(LABL(NODEB)) 230,223,230
223  LABL(NODEB) =IJ
    IPL=IPL+1
    IWV(IPL)=NODEB

```

```

IF(NODEB-IS) 230,225,230
225 IMTHR=1
230 CONTINUE
C      UPDATE COST FOR NON-S
IF(NONS-MAXA) 240,240,345
240 DO 270 I=NONS,MAXA
IJ=JWV(I)
JI=IJ-NR
IF(JI) 242,242,244
242 JI=JI+NR
244 KOSTA=KOS(IJ)
KOSTB=KOSTA-MIN
KOS(IJ)=KOSTB
KOS(JI)=-KOSTB
C      CHECK FOR MIRROR LEAVING MU STATE
C      CHECK LATER FOR COMBINING IF-CHECKS HERE
260 IF(KOSTA) 270,262,262
262 IF(KOSTB) 264,270,270
264 IF(NC(IJ)) 270,269,269
269 IF(NC(JI)) 270,270,2691
2691 CALL RIGHT(NA(IJ),JI)
270 CONTINUE
C      OUT-OF-KILTER CHECK
345 IF(NC(II)) 360,350,351
350 IF(NC(JZ)) 360,980,980
351 IF(KOS(II)) 360,350,350
C      BREAKTHROUGH CHECK
360 IF(IMTHR) 361,361,96
361 IF(IPS-IPL) 84,200,84
980 DO 981 I=1,IPL
J=IWV(I)
981 LABL(J)=0
990 CONTINUE
1000 CONTINUE
TOTL = 0.0
DO 1010 I=1,NR
KOS(I)=KOS(I)-ISAVE(I)
NC(I)=JSAVE(I)-NC(I)
TOTL = TOTL + NC(I) * ISAVE(I)
1010 CONTINUE
RETURN
END

```

```

C -----
C           =====  RIGHT   =====
C
SUBROUTINE RIGHT(I,INDEX)
COMMON /AA/ NR, NN, FSBL, NTIM, TOTL, MAXA
COMMON /B1/ IWV(1500) /B2/ LABL(1500) /B3/ NODE(1500)
COMMON /B4/ MIDL(1500) /B5/ NSAVE(1500)
COMMON /C1/ ILO(34000) /C2/ ISAVE(34000) /C3/ JSIZE(34000)
COMMON /C4/ JWV(34000)
COMMON /D1/ KOS(68000) /D2/ MIR(68000) /D3/ NA(68000)
COMMON /D4/ NC(68000) /D5/ NF(68000)

```

```

LOGICAL      FSBL
1234 CONTINUE
    MID=MIDL(I)
    IA=NODE(I)
    DO 1 II=IA,MID
    IF(MIR(II)-INDEX) 1,3,1
1   CONTINUE
    KWAY=1
2   PRINT 900, I,INDEX,KWAY
    IFROM=NODE(I)
    ITO=NODE(I+1)-1
    PRINT 910,IFROM,MIDL(I),ITo,(K,MIR(K),K=IFROM,ITo)
910  FORMAT(3I6/(2016))
    RETURN
3   ITEMP=MIR(MID)
    MIR(MID)=INDEX
    MIR(II)=ITEMP
    MIDL(I)=MID-1
    RETURN
    ENTRY LEFT(I,INDEX)
    MID=MIDL(I)+1
    IB=NODE(I+1)-1
    DO 10 II=MID,IB
    IF(MIR(II)-INDEX) 10,12,10
10  CONTINUE
    KWAY=2
    GO TO 2
12  ITEMP=MIR(MID)
    MIR(MID)=INDEX
    MIR(II)=ITEMP
    MIDL(I)=MID
    RETURN
900  FORMAT(5H NODE,15,5H ARC, 15, 16H LOST ON SHIFT ,14,4H LOC ,14
1  )
    ENTRY DUMPO
    NLines = 1
    ID = INDEX
    PRINT 1120, ID
    DO 1070 M=1,NLines
    N=M+NR
    I=NA(N)
    J=NA(M)
    L=ILO(M)
    K=JSAVE(M)
    KOST=ISAVE(M)
    KBAR=KOS(M)
    IFLOW=K-NC(M)
    IF(IFLOW.LT.L .OR. IFLOW.GT.K) PRINT 1121
    IF(KBAR) 1065,1070,1067
1065 IF(IFLOW.LT.K) PRINT 1122
    GO TO 1070
1067 IF(IFLOW.GT.L) PRINT 1122
1070 PRINT 1125,M,I,J,L,K,IFLOW,KOST,KBAR
1125 FORMAT(3I5,3I10,5X,2I10)
1120 FORMAT('1 ARC     I      J      L      K      IFLOW

```

```
*OST      KBAR' ,115      /)
1121 FORMAT(' THE FOLLOWING ARC IS PRIMAL INFEASIBLE')
1122 FORMAT(' THE FOLLOWING ARC IS DUAL INFEASIBLE')
      RETURN
      END
```

C-----

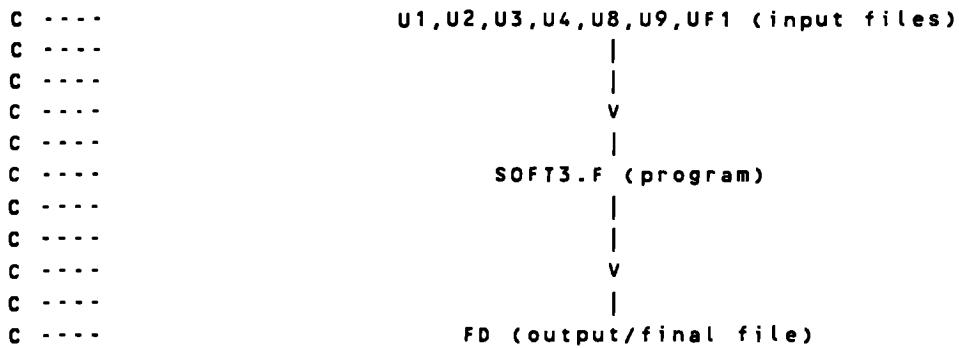
```
C ---- ****  
C ---- * SOFT3.F  
C ---- * Documented on SEPT 1990  
C ---- * Run on a 386 machine  
C ---- * Compiler used NDP Fortran  
C ---- * Grain used SOFT WHEAT  
C ---- * Data files used F1,F2,F3,F4,F8,F9  
C ---- * Intermediate input file UF1  
C ---- * Final output file FD19  
C ---- *  
C ---- ****
```

```
C ---- Declaration of 2 common blocks.  
C ---- FLOW is the array that stores the output from the previous  
C ---- program SOFT2.F, it is used as an input for this program.  
C ---- K is counter.  
C ---- U1,U2,U3,U4,U8,U9,UF1,FD store unit numbers for the files.
```

```
COMMON /GO/ K, OPERATION /G1/ FLOW(50000)  
INTEGER U1,U2,U3,U4,U8,U9,UF1,FD  
  
C ---- Unit number for each file.  
C ---- U1 is the unit number for data file, F1.sft.  
C ---- F1.sft contains information about the model.  
C ---- U2 is the unit number for data file, F2.sft.  
C ---- F2.sft contains information on TRUCK mileage.  
C ---- U3 is the unit number for data file, F3.sft.  
C ---- F3.sft contains information on RAIL costs.  
C ---- U4 is the unit number for data file, F4.sft.  
C ---- F4.sft contains information on BARGE costs.  
C ---- U8 is the unit number for data file, F8.sft.  
C ---- F8.sft contains information about SUPPLIES and DEMANDS.  
C ---- U9 is for the unit number for data files, F9.sft.  
C ---- F9.sft contains the NAMES of all the regions.  
C ---- UF1 is the unit number for the input file produced by SOFT2.F.  
C ---- It is the output file from SOFT2.F and  
C ---- is used as an input file for this program.  
C ---- FD is the unit number for the final (output) data file.  
C ---- This is the final output file which will show all the solutions  
C ---- for this model.
```

```
U1 = 13  
U2 = 14  
U3 = 15  
U4 = 16  
U8 = 8  
U9 = 9  
UF1 = 18  
FD = 19
```

C ---- The diagram below shows the files and program involved.



C ---- To open files for reading and writing

```
OPEN (UNIT = U1, FILE = 'F1.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U2, FILE = 'F2.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U3, FILE = 'F3.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U4, FILE = 'F4.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U8, FILE = 'F8.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = U9, FILE = 'F9.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
OPEN (UNIT = UF1, FILE = 'FLOW1_2.SFT', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'UNFORMATTED')
OPEN (UNIT = FD, FILE = 'final.out', STATUS = 'UNKNOWN',
1      ACCESS = 'SEQUENTIAL', FORM = 'FORMATTED')
```

C ---- The output data (from 2nd program) is required to run this  
C ---- 3rd and final program.  
C ---- Read in the processed data in FLOW1\_2.SFT with unit number  
C ---- UF1. The processed data is the output of SOFT2.F.  
C ---- N is the total number of data processed in SOFT2.F.  
C ---- FLOW is an array which contains relevant data required to get  
C ---- the final output of this model.  
C ---- Invoke subroutine DRIVER which is responsible for the overall  
C ---- supervision of this program.

```
READ  (UF1,100) N, ( FLOW(I), I = 1, N )
100  FORMAT(20I8)
      K      = 0
      CALL DRIVER
      ENDFILE UF1
      STOP
      END
```

```

C -----
C           ===== DRIVER =====
C
C ----- DRIVER acts as the "driver" of this program i.e it calls
C ----- the other subroutines, SURPLS, RELVTR, PELVTR, and DEMAND
C ----- and read in all the data from the respective data files.
C ----- Calculate the total cost of storage, cost of transportation
C ----- by truck, rail, barge, and ship and total handling cost.
C ----- Write all the results to the output file, FD.

C -----
C           ===== SUBROUTINE DRIVER =====

C ----- Declaration of all variables.

COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /B1/ SRGN(65) /B2/ DRGN(65) /B3/ RIVR(45)
COMMON /B4/ PORT(20) /B5/ FRGN(25)
COMMON /F3/ TL0S, RL0S, TL0R, RL0R, BL0R, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION /G1/ FLOW(50000)
COMMON /GA/ SUMG, SUMT, SUMR, SUMB, SUMS, SUMH
COMMON /GB/ TOPR, TODD, TOFD
COMMON /GC/ TGS0(65), TGRD(65), TGRF(65)
COMMON /GD/ TGRR(65,4), TGRP(65,4)
COMMON /R1/ NAM1(65,3), NAM2(65,3), NAM3(45,3)
COMMON /R2/ NAM4(20,3), NAM5(25,3)
DIMENSION TITL(16)
INTEGER TITL,
INTEGER FLOW
INTEGER SRGN, DRGN, RIVR, PORT, FRGN
INTEGER U1,U2,U3,U4,U8,U9,UF1,FD

C ----- Layout of the output file.
C ----- The format used to display the solution of this model.

500 FORMAT ( 20A4 )
510 FORMAT ( 20I4 )
520 FORMAT ( 10F8.3 )
530 FORMAT ( 3A4 )
600 FORMAT ( 1H1, 5X, 'Network Generator', /,
1       6X, 'For Grain Shipment Problem', // )
610 FORMAT ( 6X, 'SUPPLY', I15, 6X, 'DEMAND', I15, / )
620 FORMAT ( 6X, '?????????????????????????????????????', /,
1       6X, 'INFEASIBLE NETWORK. DEMAND EXCEEDS SUPPLY', //,
2       6X, '?????????????????????????????????????????', / )
650 FORMAT ( 3I7, 3I10 )

C ----- Read in and write the title(TITL) of this model onto the final
C ----- report.
C ----- Read in the number of surplus(NOSR), deficit(NODR), river(NORE),

```

```
C ---- port(NOPE) and foreign(NOFR) regions.  
C ---- Read in the number of time period(NOTP) and the number of days  
C ---- (NDAY(I)) in each period.  
C ---- Read in all the code numbers(SRGN(I)) of the surplus regions.
```

```
1  
      READ  (U1,500) TITL  
      WRITE (19,600)  
      WRITE (19,500) TITL  
      READ  (U1,510) NOSR, NODR, NORE, NOPE, NOFR  
      READ  (U1,510) NOTP, ( NDAY(I), I = 1, NOTP )  
      READ  (U1,500) ( SRGN(I), I = 1, NOSR )
```

```
C ---- Check that the number of regions is greater than 0.  
C ---- If it is then read in all the code numbers for the regions  
C ---- (deficit(DRGN), river(RIVR), port(PORT) and foreign(FRGN)).  
C ---- The number of time factor (NOTF) is found to be the  
C ---- number of time period(NOTP) plus 1. (i.e 3+1)
```

```
      IF ( NODR .GT. 0 ) READ  (U1,500) ( DRGN(I), I = 1, NODR )  
      IF ( NORE .GT. 0 ) READ  (U1,500) ( RIVR(I), I = 1, NORE )  
      IF ( NOPE .GT. 0 ) READ  (U1,500) ( PORT(I), I = 1, NOPE )  
      IF ( NOFR .GT. 0 ) READ  (U1,500) ( FRGN(I), I = 1, NOFR )  
      NOTF    = NOTP + 1
```

```
C ---- Initialise the following arrays:  
C ---- TGSO stores Total amount of Grain shipped to a Surplus region.  
C ---- TGRD stores Total amount of Grain shipped to a Deficit region.  
C ---- TGRF stores Total amount of Grain shipped to a Foreign region.
```

```
      DO 1100 I = 1, 53  
      TGSO(I)    = 0.0  
      TGRD(I)    = 0.0  
      TGRF(I)    = 0.0
```

```
C ---- Initialaise the arrays, TGRR and TGRP for each time factor(NOTF).  
C ---- TGRR stores Total amount of Grain shipped to a River region.  
C ---- TGRP stores Total amount of Grain shipped to a Port region.
```

```
      DO 1100 N = 1, NOTF  
      TGRR(I,N) = 0.0  
      TGRP(I,N) = 0.0  
1100 CONTINUE
```

```
C ---- Initialisation of all variables used.  
C ---- TOPR Total grain PROduced from surplus regions.  
C ---- TODD Total grain Demanded by Deficit regions.  
C ---- TOFD Total grain Demanded by Foreign regions.  
C ---- SUMG SUM of storage cost.  
C ---- SUMT SUM of Truck cost.
```

C ---- SUMR SUM of Rail cost.  
C ---- SUMB SUM of Barge cost.  
C ---- SUMS SUM of Shipping cost.  
C ---- SUMH SUM of Handling cost.

TOPR	= 0.0
TODD	= 0.0
TOFD	= 0.0
SUMG	= 0.0
SUMT	= 0.0
SUMR	= 0.0
SUMB	= 0.0
SUMS	= 0.0
SUMH	= 0.0

C ---- Read in the loading and unloading costs.  
C ---- TLOS Truck Loading (country elevator) cost.  
C ---- RLOS Railcar Loading (country elevator) cost.  
C ---- TLOR Truck LOading (River location) cost.  
C ---- RLOR Railcar LOading (River location) cost.  
C ---- BLOR Barge LOading (River location) cost.  
C ---- SLOP Ship LOading (Port loading ocean vessel) cost.  
C ---- TRID TRuck unloading (Deficit) cost.  
C ---- RRID Rail unloading (Deficit) cost.  
C ---- TRIR River location unloading TRuck cost.  
C ---- RRIR River location unloading Rail cost.  
C ---- BRIR River location unloading Barge cost.  
C ---- TRIP Port unloading Truck cost.  
C ---- RRIP Port unloading Rail cost.  
C ---- BRIP Port unloading Barge cost.

READ (U1,520) TLOS, RLOS, TLOR, RLOR, BLOR, SLOP  
READ (U1,520) TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP

C ---- Read in the actual (real) names of all the regions involved.  
C ---- NAM1 stores the names of the surplus regions.  
C ---- NAM2 stores the names of the deficit regions.  
C ---- NAM3 stores the names of the river regions.  
C ---- NAM4 stores the names of the port regions.  
C ---- NAM5 stores the names of the foreign regions.

```
DO 2100 I = 1, NOSR
READ (U9,530) ( NAM1(I,J), J = 1, 3 )
2100 CONTINUE
DO 2200 I = 1, NODR
READ (U9,530) ( NAM2(I,J), J = 1, 3 )
2200 CONTINUE
DO 2300 I = 1, NORE
READ (U9,530) ( NAM3(I,J), J = 1, 3 )
2300 CONTINUE
DO 2400 I = 1, NOPE
```

```

      READ  (U9,530) ( NAM4(I,J), J = 1, 3 )
2400 CONTINUE
      DO 2500 I = 1, NOFR
      READ  (U9,530) ( NAM5(I,J), J = 1, 3 )
2500 CONTINUE

C ----- IPRD stores the amount of grain PRoDuced and is set to 0.
C ----- IEXP stores the amount of grain EXPended is set to 0.
C ----- K is a counter.
C ----- Invoke subroutine SURPLS to deal with all the information
C ----- relating to the surplus regions.
C ----- Call subroutine RELVTR and PELVTR if there is more than
C ----- one river region and one port region involved.
C ----- RELVTR and PELVTR deal with the River rEgions and Port rEgions
C ----- respectively and their Truck and Rail costs.
C ----- Subroutine DEMAND is called to calculate all demand required
C ----- by each region.

      IPRD      = 0
      IEXP      = 0
      K         = 0
      CALL SURPLS ( IPRD )
      IF ( NORE .GT. 0 ) CALL RELVTR
      IF ( NOPE .GT. 0 ) CALL PELVTR
      CALL DEMAND ( IEXP )

C ----- Write the total cost of Storage, Truck, Rail, Barge
C ----- Ship and Handling to the output file, FD.
C ----- FD is the final output data file.
C ----- The format used is 740.

      WRITE (FD,740) SUMG, SUMT, SUMR, SUMB, SUMS, SUMH

C ----- The format used in displaying the above information
C ----- in the output file, FD.

      740 FORMAT ( 1H1, //, 5X, 'STORAGE COST', F12.0, /,
      1 5X, 'TRUCK COST ', F12.0, /, 5X, 'RAIL COST ', F12.0, /,
      2 5X, 'BARGE COST ', F12.0, /, 5X, 'SHIP COST ', F12.0, /,
      3 5X, 'HANDLING CST', F12.0 )
      TOPR = (TOPR / OPERATION)
      TODD = (TODD / OPERATION)
      TOFD = (TOFD / OPERATION)
      WRITE (FD,750) TOPR, TODD, TOFD
      750 FORMAT ( ///, 5X, 'GRAIN SHIPPED FROM SURPLUS REGIONS', F15.0,
      1           //, 5X, 'GRAIN SHIPPED TO DEFICIT REGIONS ', F15.0,
      2           //, 5X, 'GRAIN SHIPPED TO FOREIGN REGIONS ', F15.0,
      3           // )
      RETURN
      END

C -----
C          ===== SURPLS =====
C ----- SURPLS is used to display the heading of the output file.
C ----- It reads the data concerning the storage, the amount of

```

C ---- grain produced by, truck(mileage) and rail cost  
C ---- of all the surplus regions.

```
SUBROUTINE SURPLS ( IPRD )
DIMENSION T(4), R(4), STCT(65), S(65,4)
COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /B1/ SRGN(65) /B2/ DRGN(65) /B3/ RIVR(45)
COMMON /B4/ PORT(20)
COMMON /C1/ SDTR(65,65) /C2/ SDRL(65,65)
COMMON /C3/ SRTR(65,45) /C4/ SRRL(65,45)
COMMON /C5/ SPTR(65,16) /C6/ SPRL(65,16)
COMMON /D1/ RAIL, TRUCK, BARGE, SHIP, CHOICE
COMMON /E1/ SPLY(65)
COMMON /F1/ STOR(65) /F2/ SCST(4)
COMMON /F3/ TL0S, RL0S, TL0R, RL0R, BL0R, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION /G1/ FLOW(50000)
COMMON /GA/ SUMG, SUMT, SUMR, SUMB, SUMS, SUMH
COMMON /GB/ TOPR, TODD, TOFD
COMMON /GC/ TGSO(65), TGRD(65), TGRF(65)
COMMON /GD/ TGRR(65,4), TGRP(65,4)
COMMON /R1/ NAM1(65,3), NAM2(65,3), NAM3(45,3)
COMMON /R2/ NAM4(20,3), NAM5(25,3)
COMMON /H1/ IDEN(65), ALFA(65), BETA(65)
INTEGER FLOW
INTEGER SRGN, DRGN, RIVR, PORT, FRGN
INTEGER U1,U2,U3,U4,U8,U9,UF1,FD
```

C ---- The format used to display the results on the output file.

```
500 FORMAT ( 20A4 )
520 FORMAT ( 10F8.3 )
700 FORMAT ( 1H1, //, 9X, 18HORIGIN/DESTN MODE, 6X, 6HSUPPLY, 14X,
    1      16HSHIPMENT BY TIME, 16X, 5HTOTAL, 6X, 4HUNIT,
    2      7X, 5HTOTAL, 5X, 8HHANDLING, /, 39X, 4I10, 4X,
    3      8HSHIPMENT, 5X, 28HCOST HAULING COST COSTS, // )
710 FORMAT ( /, 5X, 1HS, A4, 1X, 3A4, 4X, F12.0, / )
720 FORMAT ( 5X, 1HD, A4, 1X, 3A4, 2X, 2HT, 12X, 4F10.0,
    1      F12.0, F10.5, 2F12.0 )
730 FORMAT ( 5X, 1HD, A4, 1X, 3A4, 2X, 2HR, 12X, 4F10.0,
    1      F12.0, F10.5, 2F12.0 )
740 FORMAT ( 5X, 1HR, A4, 1X, 3A4, 2X, 2HT, 12X, 4F10.0,
    1      F12.0, F10.5, 2F12.0 )
750 FORMAT ( 5X, 1HR, A4, 1X, 3A4, 2X, 2HR, 12X, 4F10.0,
    1      F12.0, F10.5, 2F12.0 )
760 FORMAT ( 5X, 1HP, A4, 1X, 3A4, 2X, 2HT, 12X, 4F10.0,
    1      F12.0, F10.5, 2F12.0 )
770 FORMAT ( 5X, 1HP, A4, 1X, 3A4, 2X, 2HR, 12X, 4F10.0,
    1      F12.0, F10.5, 2F12.0 )
800 FORMAT ( 1H1, //, 5X, 14HSURPLUS REGION, 10X, 6HSUPPLY,
    1      14X, 7HSTORAGE, 12X, 12HSTORAGE COST, / )
810 FORMAT ( 3X, 1HS, A4, 1X, 3A4, 2X, F12.0, 3F10.0, F12.0 )
```

C ---- Read in the amount of grain produced (SPLY) and storage

```
C ---- available(STOR) in each surplus region; and  
C ---- the cost(COST) and rail(RAIL) factors, the conversion  
C ---- factor and the unit of measurement.
```

```
READ  (U8,520) ( SPLY(I), I = 1, NOSR )  
READ  (U1,520) ( STOR(I), I = 1, NOSR )  
READ  (U1,520) COST, RAIL, TRUCK, BARGE, SHIP  
READ  (U1,520) OPERATION, CHOICE
```

```
C ---- Check that the number of periods(NOTP) is greater than 0.  
C ---- If it is then find the storage cost for the leftover  
C ---- (surplus) grain in storage for each time period.  
C ---- SCST(N) is the cost of storage for each period.  
C ---- NDAY(N) stores the number of days in each time period.  
C ---- eg, NDAY(1) is 122 days.  
C ---- eg, SCST(2) will be the cost of storing extra grain in  
C ---- the second period.
```

```
IF ( NOTP .LE. 0 ) GO TO 1200  
DO 1100 N = 1, NOTP  
SCST(N) = COST * FLOAT(NDAY(N)) / 36.50  
1100 CONTINUE  
1200 CONTINUE
```

```
C ---- SURPLUS regions ----> by TRUCK ----> DEFICIT regions.  
C ---- Read in the Truck mileage from each surplus region  
C ---- to all the deficit regions.
```

```
DO 1300 I = 1, NOSR  
READ  (U2,520) ( SDTR(I,J), J = 1, NODR )  
1300 CONTINUE
```

```
C ---- SURPLUS regions ----> by Rail ----> DEFICIT regions.  
C ---- Read in the Rail cost from each surplus region to all  
C ---- the deficit regions.
```

```
DO 1400 I = 1, NOSR  
READ  (U3,520) ( SDRL(I,J), J = 1, NODR )  
1400 CONTINUE
```

```
C ---- SURPLUS regions ----> by TRUCK ----> Selected River points.  
C ---- Read in the TRUCK mileage from each surplus region to  
C ---- the selected (river) points.
```

```
READ  (U2,520) ( SRTR(I,1), I = 1, NOSR )
```

```
C ---- SURPLUS regions ----> by Rail ----> RIVER regions.  
C ---- Read in the Rail cost from each surplus region to all  
C ---- the river regions.
```

```
DO 1600 I = 1, NOSR  
READ  (U3,520) ( SRRL(I,J), J = 1, MORE )  
1600 CONTINUE
```

```

C ---- SURPLUS regions ----> by TRUCK ----> PORT regions.
C ---- Read in the TRUCK mileage from each surplus region to
C ---- all the port regions.

DO 1700 I = 1, NOSR
READ (U2,520) ( SPTR(I,J), J = 1, NOPE )
1700 CONTINUE

C ---- SURPLUS regions ----> by Rail ----> PORT regions.
C ---- Read in the Rail cost from each surplus region to
C ---- all the port regions.

DO 1800 I = 1, NOSR
READ (U3,520) ( SPRL(I,J), J = 1, NOPE )
1800 CONTINUE

C ---- Read in alphas and betas of all the surplus regions.

READ (U1,520) ( ALFA(I), BETA(I), I = 1, NOSR )

C ---- Read in selected (receiving) points (IDEN(I)) that are
C ---- linked with the surplus regions.

READ (U1,500) ( IDEN(I), I = 1, NOSR )

C ---- Print out the heading of the output file i,e,
C ORIGIN/DESTN MODE SUPPLY SHIPMENT TOTAL UNIT TOTAL HANDLING
C                               SHIPMENT COST HAULING COST COSTS

WRITE (FD,700) ( I, I = 1, 4 )

C ---- Find the storage cost of each of the surplus region.

DO 4000 I = 1, NOSR
ALF      = ALFA(I)
BET      = BETA(I)
NF       = I
K        = K + 1
ISTR     = STOR(NF)
IF ( NOTP .LE. 0 ) GO TO 2200

C ---- For each surplus region, find
C ---- S(I,N) is the amount of grain in storage per period(FLOW),
C ---- STCT(I) is the total storage cost.
C ---- SCST(N) is the cost of storage for each period.
C ---- eg, SCST(2) will be the cost of storing extra grain in
C ---- the second period.

STCT(I) = 0.0
DO 2100 N = 1, NOTP
K        = K + 1
S(I,N)   = FLOW(K)
STCT(I) = STCT(I) + FLOW(K) * SCST(N)
2100 CONTINUE
2200 CONTINUE

```

```

C ---- Calculate the actual supply(SPLY) and display the code
C ---- number (SRGN(I)) and the name (NAM1) of
C ---- and the supply (SPLY(I)) from this surplus region.

      SPLY(I) = SPLY(I) * 1000.0
      WRITE (FD,710) SRGN(I), ( NAM1(I,L), L = 1, 3 ),
      1           SPLY(I)/OPERATION

C ---- SURPLUS region ----> by TRUCK ----> DEFICIT regions.
C ---- COST (=SDTR) is the truck mileage from each surplus to
C ---- all the deficit regions.
C ---- If the mileage is greater than 9999 then it is ignored
C ---- else the cost of handling and by truck is calculated.
C ---- Call the subroutine GENFLO to calculate the
C ---- handling cost (HCST) and truck cost(TCST) of the deficit
C ---- region.
C ---- Update the handling cost (SUMH) and the truck (hauling)
C ---- cost(SUMT).
C ---- TLOS and TRID is the loading and unloading costs.
C ---- T is the amount of grain per shipment (by time).
C ---- TT is the total shipment from this surplus region
C ---- to the selected deficit region.

      DO 2500 M = 1, NODR
      NT      = M
      COST    = SDTR(NF,NT)
      IF ( COST .LE. 225 ) THEN
      COST = ( 0.066374 + 0.104892 * COST)
      ELSE IF ( COST .GT. 225 .and. CDST .LE. 245 ) THEN
      COST = ( 23.67 + 0.0 * COST )
      ELSE IF ( COST .GT. 245 ) THEN
      COST = ( 0.68037 + 0.093976 * CDST )
      ELSE IF ( COST .GE. 9999. ) THEN
      GOTO 2300
      ENDIF

      COST     = ( COST * TRUCK ) / 100.0

      CALL GENFLO ( COST, TLOS, TRID, T, TT, TCST, HCST )
      SUMH    = SUMH + HCST
      SUMT    = SUMT + TCST

C ---- If the total shipment (TT) to the selected deficit
C ---- region is less than 0 then it is ignored i.e no
C ---- grain is transported to this deficit region else
C ---- the name of the deficit region (DRGN), amount
C ---- of grain (T(N)) per shipment by time, total
C ---- shipment(TT), unit cost(COST), total (truck)
C ---- hauling cost (TCST) and total handling cost for
C ---- this deficit region are displayed on the output file, FD.

      IF ( TT .LE. 0.0 ) GO TO 2300
      WRITE (FD,720) DRGN(M), ( NAM2(M,L), L = 1, 3 ),

```

```
1      ( T(N)/OPERATION, N = 1, NOTF ),
2      TT/OPERATION, COST*OPERATION, TCST, HCST
```

```
C ---- Update the following totals:  
C ---- TODD is total amount grain demanded by the deficit regions.  
C ---- TGSO is the total shipment by time for each surplus region.  
C ---- TOPR is the amount of grain produced by the surplus region.  
C ---- TGRD is the total shipment by time for each deficit region.
```

```
TODD      = TODD + TT  
TGSO(I)   = TGSO(I) + TT  
TOPR      = TOPR + TT  
TGRD(M)   = TGRD(M)+ TT
```

```
2300 CONTINUE
```

```
C ---- SURPLUS region ----> by RAIL ----> DEFICIT regions.  
C ---- The Rail cost (COST) from surplus region to this deficit  
C ---- region is tested. If it is greater than 999 then it is  
C ---- ignored else the rail cost (COST) is calculated.  
C ---- Subroutine GENFLO is called to find the total rail  
C ---- (hauling) cost (RCST) and the total handling (HCSR) cost  
C ---- by rail.  
C ---- RLOS and RRID is the railcar loading and unloading costs.  
C ---- R is the amount of grain per shipment (by time).  
C ---- RT is the total amount shipment for this deficit region.  
C ---- SUMR is the total rail cost.  
C ---- SUMH is the total handling cost.
```

```
COST      = SDRL(NF,NT)
IF ( COST .GE. 999. ) GO TO 2400
COST      = COST * RAIL / 100.0
CALL GENFLO ( COST, RLOS, RRID, R, RT, RCST, HCSR )
SUMR      = SUMR + RCST
SUMH      = SUMH + HCSR
```

```
C ---- If the total shipment for this deficit region(RT) is  
C ---- less than 0 i.e no grain is transported to this  
C ---- deficit region then this region is ignored and no  
C ---- output regarding this region is printed.
```

```
IF ( RT .LE. 0.0 ) GO TO 2500
```

```
C ---- Print out the information about this deficit region  
C ---- which receives grain.  
C ---- DRGN is the name of the deficit region.  
C ---- R(N) is the amount of shipment by time(per period).  
C ---- RT is the total shipment received by this deficit  
C ---- region.  
C ---- COST is the unit cost.  
C ---- RCST is the rail(hauling) cost.  
C ---- HCSR is the handling cost.
```

```
WRITE (FD,730) DRGN(M), ( NAM2(M,L), L = 1, 3 ),
1           ( R(N)/OPERATION, N = 1, NOTF ),
```

## 2 RT/OPERATION, COST\*OPERATION, RCST, HCSR

C ---- Update the following totals:  
 C ---- TODD is the amount of grain received by the deficit regions.  
 C ---- TGSO is total shipment by time for each surplus region.  
 C ---- TOPR is the amount of grain produced by surplus regions.  
 C ---- TGRD is total shipment by time for each deficit region.

```
TODD      = TODD + RT
TGSO(I)   = TGSO(I) + RT
TOPR      = TOPR + RT
TGRD(M)   = TGRD(M) + RT
```

2400 CONTINUE

2500 CONTINUE

C ---- SURPLUS region ---->TRUCK-----> RIVER region.  
 C ---- Subroutine SERIAL is used to check that all input data  
 C ---- correctly entered.  
 C ---- Check that the cost by truck (SRTR) is not greater than -  
 C ---- 999. If it is not, then the cost by truck from a surplus  
 C ---- region to this river region is found.  
 C ---- Subroutine GENFLO is called to calculate the cost of  
 C ---- hauling by truck (TCST) and the handling cost (HCST).  
 C ---- SUMH is the total handling cost.  
 C ---- SUMT is the total truck cost.

```
IDNT      = IDEN(I)
CALL SERIAL ( IDNT, NORE, RIVR, NT )
IF ( NT .EQ. 0 ) GO TO 2600
COST      = SRTR(NF,1)

IF ( COST .LE. 225 ) THEN
COST = ( 0.066374 + 0.104892 * COST)
ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
COST = ( 23.67 + 0.0 * COST )
ELSE IF ( COST .GT. 245 ) THEN
COST = ( 0.68037 + 0.093976 * COST )
ELSE IF ( COST .GE. 9999. ) THEN
GOTO 2600
ENDIF
```

```
COST      = ( COST * TRUCK ) / 100.0
```

```
CALL GENFLO ( COST, TLDS, TRIR, T, TT, TCST, HCST )
SUMH      = SUMH + HCST
SUMT      = SUMT + TCST
```

C ---- If the amount of grain supplied (TT) is less than 0  
 C ---- no information on that region is outputted, else  
 C ---- the name (RIVR(NT)) of the river region, the  
 C ---- amount of grain on each shipment by time(T(N)),  
 C ---- unit cost(COST), total hauling (truck)cost and  
 C ---- handling cost (HCST) is printed on the output file, FD.

```

    IF ( TT .LE. 0.0 ) GO TO 2600
    WRITE (FD,740) RIVR(NT), ( NAM3(NT,L), L = 1, 3 ),
1                  ( T(N)/OPERATION, N = 1, NOTF ),
2                  TT/OPERATION, COST*OPERATION, TCST, HCST

C ---- Update TGSO, the total shipment by time
C ---- and TOPR, the amount of grain produced by the surplus
C ---- regions.

    TGSO(I)      = TGSO(I) + TT
    TOPR        = TOPR + TT

C ---- Update TGRR, the total amount of grain shipped to a
C ---- river region.

    DO 2550 N = 1, NOTF
    TGRR(NT,N)= TGRR(NT,N) + T(N)
2550 CONTINUE
2600 CONTINUE

C ---- SURPLUS region -----> by RAIL -----> RIVER regions
C ---- Check that the cost by rail (COST) is less than 999.
C ---- If it is then find the cost by rail(COST) and invoke
C ---- subroutine GENFLO is calculate the hauling (rail)
C ---- cost(RCST) and the handling cost (HCSR).
C ---- R is the amount of grain in each shipment by time.
C ---- RT is the amount of grain in total shipment.

    DO 2800 M = 1, MORE
    NT          = M
    COST        = SRRL(NF,NT)
    IF ( COST .GE. 999. ) GO TO 2700
    COST        = COST * RAIL / 100.0
    CALL GENFLO ( COST, RLOS, RRIR, R, RT, RCST, HCSR )

C ---- If the amount of grain in total shipment in this river
C ---- region is less than 0 i.e no grain is received then
C ---- it is ignored, else update the total hauling(rail) cost
C ---- (SUMR) and the handling cost(SUMH) and output
C ---- the code number(RIVR(NT)) and the name (NAM3(NT,L)) of
C ---- the river region, the amount of grain in each shipment
C ---- by time (R(N)), the total shipment(RT), unit cost(COST),
C ---- total hauling cost(RCST) and handling cost(HCSR).

    IF ( RT .LE. 0.0 ) GO TO 2700
    SUMH        = SUMH + HCSR
    SUMR        = SUMR + RCST
    WRITE (FD,750) RIVR(NT), ( NAM3(NT,L), L = 1, 3 ),
1                  ( R(N)/OPERATION, N = 1, NOTF ),
2                  RT/OPERATION, COST*OPERATION, RCST, HCSR

C ---- Update the following totals: TGSO(surplus),TOPR(port).

    TGSO(I)      = TGSO(I) + RT
    TOPR        = TOPR + RT

```

```

C ---- Update the total shipment (TGRR) to a river region.

      DO 2650 N = 1, NOTF
      TGRR(NT,N)= TGRR(NT,N) + R(N)
2650 CONTINUE
2700 CONTINUE
2800 CONTINUE

C ---- SURPLUS REGION -----> by TRUCK -----> PORT regions
C ---- Check that the cost by truck (COST) is less than 9999.
C ---- If it is then find the cost by rail(COST) and invoke
C ---- subroutine GENFLO is calculate the hauling (truck)
C ---- cost(TCST) and the handling cost (HCST).
C ---- T is the amount of grain in each shipment by time.
C ---- TT is the amount of grain in total shipment.
C ---- Update the total hauling(truck) cost
C ---- (SUMT.) and the handling cost(SUMH).

      DO 3000 M = 1, NOPE
      NT      = M
      COST    = SPTR(NF,NT)

      IF ( COST .LE. 225 ) THEN
      COST = ( 0.066374 + 0.104892 * COST)
      ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
      COST    = ( 23.67 + 0.0 * COST )
      ELSE IF ( COST .GT. 245 ) THEN
      COST    = ( 0.68037 + 0.093976 * COST )
      ELSE IF ( COST .GE. 9999. ) THEN
      GOTO 2900
      ENDIF

      COST    = ( COST * TRUCK ) / 100.0

      CALL GENFLO ( COST, TL0S, TRIP, T, TT, TCST, HCST )
      SUMT   = SUMT + TCST
      SUMH   = SUMH + HCST

C ---- If the amount of grain in total shipment in this port
C ---- region is less than 0 i.e no grain is received then
C ---- it is ignored, else
C ---- the name (PORT(M)) of the port region, the
C ---- amount of grain on each shipment by time(T(N)),
C ---- unit cost(COST), total hauling (truck)cost (TCST) and
C ---- handling cost (HCST) is printed.

      IF ( TT .LE. 0.0 ) GO TO 2900
      WRITE (FD,760) PORT(M), ( NAM4(M,L), L = 1, 3 ),
      1           ( T(N)/OPERATION, N = 1, NOTF ),
      2           TT/OPERATION, COST*OPERATION, TCST, HCST

C ---- Update the following totals: TGSO, and TOPR.

      TGSO(I)    = TGSO(I) + TT

```

```

TOPR      = TOPR + TT

C ---- Update total shipment (TGRP) to a port region.

DO 2850 N = 1, NOTF
TGRP(M,N) = TGRP(M,N) + T(N)
2850 CONTINUE
2900 CONTINUE

C ---- SURPLUS region -----> by RAIL -----> PORT regions
C ---- SPRL stands for the Rail cost from a Surplus region
C ---- to a Port region.
C ---- Check that the cost by rail (COST) is less than 999.
C ---- If it is then find the cost by rail(COST) and invoke
C ---- subroutine GENFLO is calculate the hauling (rail)
C ---- cost(RCST) and the handling cost (HCSR).
C ---- R is the amount of grain in each shipment by time.
C ---- RT is the amount of grain in total shipment.
C ---- Update the total hauling(rail) cost
C ---- (SUMR) and the handling cost(SUMH).

COST      = SPRL(NF,NT)
IF ( COST .GE. 999. ) GO TO 3000
COST      = COST * RAIL / 100.0
CALL GENFLO ( COST, RLOS, RRIP, R, RT, RCST, HCSR )
SUMR      = SUMR + RCST
SUMH      = SUMH + HCSR

C ---- If the amount of grain in total shipment (RT)
C ---- in this port region is less than 0 i.e no grain is
C ---- received then it is ignored, else output
C ---- the code number (PORT(M)) and the name (NAM4(M,L)) of
C ---- the port region, the amount of grain in each shipment
C ---- by time (R(N)), the total shipment(RT), unit cost(COST),
C ---- total hauling cost(RCST) and handling cost(HCSR).

IF ( RT .LE. 0.0 ) GO TO 3000
WRITE (FD,770) PORT(M), ( NAM4(M,L), L = 1, 3 ),
1           ( R(N)/OPERATION, N = 1, NOTF ),
2           RT/OPERATION, COST*OPERATION, RCST, HCSR

C ---- Update the following totals: TGSO(surplus), TOPR(port).

TGSO(I)   = TGSO(I) + RT
TOPR      = TOPR + RT

C ---- Update shipment by time to a port(TGRP).

DO 2950 N = 1, NOTF
TGRP(M,N) = TGRP(M,N) + R(N)
2950 CONTINUE
3000 CONTINUE
3900 CONTINUE
4000 CONTINUE

```

```

C ---- Write this heading:
C ---- SURPLUS REGION    SUPPLY    STORAGE    STORAGE COST
C ---- on the output file, FD.

        WRITE (FD,800)

C ---- Output the following data:
C ---- the code number (SRGN), name(NAM1(I,L)),
C ---- the amount of grain produced by this surplus region (SPLY(I)),
C ---- the amount of grain in storage (S(I,N)) per
C ---- time period and storage cost (STCT) of each surplus region.
C ---- An example of the output:
C ---- SURPLUS REGION    SUPPLY    STORAGE    STORAGE COST
C ---- 064 ABELINE, TX    5702     4424    2212    0          144642

```

C ---- Increment the total amount of storage.

```

DO 5000 I = 1, NOSR
WRITE (FD,810) SRGN(I), ( NAM1(I,L), L = 1, 3 ),
1           SPLY(I)/OPERATION,
2           ( S(I,N)/OPERATION, N = 1, NOTP ), STCT(I)
SUMG      = SUMG + STCT(I)
5000 CONTINUE
RETURN
END

```

```

C -----
C           ===== RELVTR =====
C ---- Subroutine RELVTR deals with the river regions.
C ---- It finds the cost of transportation from the river
C ---- regions to selected barge (river) points and ports.
C ---- Truck, rail and barge costs are considered.

```

```

SUBROUTINE RELVTR
DIMENSION T(4), R(4)
COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /B2/ DRGN(65) /B3/ RIVR(45)
COMMON /B4/ PORT(20)
COMMON /C1/ RDTR(65,65) /C2/ RDRL(65,65)
COMMON /C3/ RRBG(65,45) /C5/ RPBG(65,16)
COMMON /D1/ RAIL, TRUCK, BARGE, SHIP
COMMON /F3/ TL0S, RL0S, TL0R, RL0R, BL0R, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /G0/ K, OPERATION /G1/ FLOW(50000)
COMMON /GA/ SUMG, SUMT, SUMR, SUMB, SUMS, SUMH
COMMON /GB/ TOPR, TODD, TDFFD
COMMON /GC/ TGSO(65), TGRD(65), TGRF(65)
COMMON /GD/ TGRR(65,4), TGRP(65,4)
COMMON /R1/ NAM1(65,3), NAM2(65,3), NAM3(45,3)
COMMON /R2/ NAM4(20,3), NAM5(25,3)
COMMON /H1/ IDEN(65), ALFA(65), BETA(65)
DIMENSION IDN1(10), IDN2(10), IDN3(25)
INTEGER U1,U2,U3,U4,U8,U9,UF1,FD

```

```

C ---- The layout of the output.

500 FORMAT ( 20A4 )
510 FORMAT ( 20I4 )
520 FORMAT ( 10F8.3 )
700 FORMAT ( 1H1, //, 9X, 18HORIGIN/DESTN MODE, 6X, 6HSUPPLY, 14X,
    1       16HSHIPMENT BY TIME, 16X, 5HTOTAL, 6X, 4HUNIT,
    2       7X, 5HTOTAL, 5X, 8HHANDLING, /, 39X, 4I10, 4X,
    3       8HSHIPMENT, 5X, 28HCOST HAULING COST COSTS, // )
710 FORMAT ( /, 5X, 1HR, A4, 1X, 3A4, 16X, 4F10.0, / )
720 FORMAT ( 5X, 1HD, A4, 1X, 3A4, 2X, 2HT, 12X, 4F10.0,
    1       F12.0, F10.5, 2F12.0 )
730 FORMAT ( 5X, 1HD, A4, 1X, 3A4, 2X, 2HR, 12X, 4F10.0,
    1       F12.0, F10.5, 2F12.0 )
740 FORMAT ( 5X, 1HR, A4, 1X, 3A4, 2X, 2HB, 12X, 4F10.0,
    1       F12.0, F10.5, 2F12.0 )
750 FORMAT ( 5X, 1HP, A4, 1X, 3A4, 2X, 2HB, 12X, 4F10.0,
    1       F12.0, F10.5, 2F12.0 )

C ---- Read in the alfas and betas of the deficit regions.
C ---- Read in the selected (river) barge shipping
C ---- locations(IDEN(I)).

      READ  (U1,52D) ( ALFA(I), BETA(I), I = 1, NODR )
      READ  (U1,500) ( IDEN(I), I = 1, NODR )

C ---- Read in the number of barge (river) points(NRES), and
C ---- barge (port) points(NPES); and their respective code
C ---- numbers.
C ---- IDN1(I) stores the code numbers of the barge (river)
C ---- points which are N,KV,Chat,Gun,Fl.
C ---- IDN2(I) stores the code numbers of the barge (port)
C ---- points which are ND,M,B,Ch,Port.

      READ  (U1,510) NRES, NPES
      READ  (U1,500) ( IDN1(I), I = 1, NRES ), ( IDN2(I), I = 1, NPES )
      :

C ---- Read in the TRUCK mileage from the selected barge
C ---- unloading (shipping)points to each of the deficit region.

      READ  (U2,520) ( RDTR(1,J), J = 1, NODR )

C ---- RIVER regions ----> by RAIL ----> DEFICIT regions
C ---- Read in the Rail cost from each of the river
C ---- region to all the deficit regions.

      DO 1200 I = 1, MORE
      READ  (U3,520) ( RDRL(I,J), J = 1, NODR )
1200 CONTINUE

C ---- RIVER regions ----> by Barge ----> Barge (RIVER) points
C ---- Read in the barge cost(per bushel) from each of
C ---- the river region to all the barge(river) points.
C ---- NRES is the number of barge(river) points.

```

```

C ---- Barge (river) points are N,KV,Chat,Gun,Fl.

      DO 1300 I = 1, NOR
      READ  (U4,520) ( RRBG(I,J), J = 1, NRES )
1300 CONTINUE

C ---- RIVER regions ----> by Barge ----> Barge (PORT) points.
C ---- Read in the barge cost(per bushel) from each of
C ---- the river region to all the barge(port) points.
C ---- NPES is the number of barge(port) points which
C ---- are NO,M,B,Ch,Port.

      DO 1400 I = 1, NOR
      READ  (U4,520) ( RPBG(I,J), J = 1, NPES )
1400 CONTINUE

C ---- Read in the number of ports (LAKE) above the L&D 26
C ---- their code numbers(IDN3(I)).
C ---- Note: might not be used in this model.

      READ  (U4,510) LAKE
      READ  (U4,500) ( IDN3(I), I = 1, LAKE )

C ---- Print out the heading of the output file i.e.,
C ORIGIN/DESTN MODE SUPPLY SHIPMENT TOTAL UNIT TOTAL HANDLING
C                               SHIPMENT COST HAULING COST COSTS

      WRITE (FD,700) ( I, I = 1, NOTF )

C ---- Write out the code number(RIVR(I)), name (NAM3(I,L)) of
C ---- the river region, and the amount of grain (TGRR) shipped by
C ---- time (per period) to each river region.
C ---- NOTF is the number of time factor.
C ---- FD is the output file, and format 700 is used.
C ---- eg R 601 ST.PAUL,MN 0 0 0 0

      DO 3000 I = 1, NOR
      WRITE (FD,710) RIVR(I), ( NAM3(I,L), L = 1, 3 ),
1          ( TGRR(I,N)/OPERATION, N = 1, NOTF )
      NF      = I

C ---- Selected BARGE (river)points-->by TRUCK-->DEFICIT regions.
C ---- Read in the TRUCK mileage from each of the barge
C ---- unloading point to all the deficit regions.
C ---- If the cost(COST) is greater than 9999 then it is
C ---- ignored else the cost by truck(COST) is calculated.
C ---- The COST includes BET and ALF.
C ---- Subroutine GENFLO is used to calculate the hauling
C ---- truck cost(TCST) and the handling cost (HCST).
C ---- T is the amount of grain in each shipment by time.
C ---- TT is the amount of grain in total shipment.

      DO 2200 M = 1, NODR
      IDNT      = IDEN(M)
      CALL SERIAL ( IDNT, NOR, RIVR, NT )

```

```

IF ( NF .NE. NT ) GO TO 2200
NT      = M
COST    = RDTR(1,NT)
IF ( COST .GE. 9999. ) GO TO 2200
ALF    = ALFA(NT)
BET    = BETA(NT)

IF ( COST .LE. 225 ) THEN
COST = ( 0.066374 + 0.104892 * COST)
ELSE IF ( COST .GT. 225 .and. COST .LE. 245 ) THEN
COST = ( 23.67 + 0.0 * COST )
ELSE IF ( COST .GT. 245 ) THEN
COST = ( 0.68037 + 0.093976 * COST )
ELSE IF ( COST .GE. 9999. ) THEN
GOTO 2200
ENDIF

COST     = ( COST * TRUCK ) / 100.0

COST     = ( COST * BET + ALF ) / 100.0
CALL GENFLO ( COST, TLOR, TRID, T, TT, TCST, HCST )

C ---- Check that the amount of grain shipped is more than 0.
C ---- If it is, update the total hauling (truck) cost
C ---- (SUMT) and the handling cost(SUMH).

IF ( TT .LE. 0.0 ) GO TO 2200
SUMT    = SUMT + TCST
SUMH    = SUMH + HCST

C ---- Write all the information about the deficit region
C ---- which receives grain by truck.
C ---- DRGN(M) is the code number of the deficit region.
C ---- NAM2(M,L) is the name of the deficit region.
C ---- T(N) is the amount of grain received per shipment by
C ---- time. TT is the total shipment for this deficit region.
C ---- COST is the unit cost.
C ---- TCST is the total hauling(TRUCK) cost.
C ---- HCST is the handling cost.

WRITE (FD,720) DRGN(M), ( NAM2(M,L), L = 1, 3 ),
1           ( T(N)/OPERATION, N = 1, NOTF ),
2           TT/OPERATION, COST*OPERATION, TCST, HCST

C ---- Update the total amount of grain demanded(TODD) by
C ---- the deficit regions.
C ---- Update the total amount of grain shipped (TGRD(M))
C ---- to this deficit region.

TODD    = TODD + TT
TGRD(M) = TGRD(M) + TT
2200 CONTINUE

C ---- Selected BARGE (river) points-->by Rail-->DEFICIT regions.

```

```

C ---- Read in the Rail cost from each of the barge
C ---- unloading point to all the deficit regions.
C ---- If the cost(COST) is greater than 999 then it is
C ---- ignored else the cost by truck(COST) is calculated.
C ---- The COST includes BET and ALF.
C ---- Subroutine GENFLO is used to calculate the hauling
C ---- rail cost(RCST) and the handling cost (HCSR).
C ---- R is the amount of grain in each shipment by time.
C ---- RT is the amount of grain in total shipment.

      DO 2300 M = 1, NODR
      NT      = M
      COST    = RDRL(NF,NT)
      IF ( COST .GE. 999. ) GO TO 2300
      COST    = COST * RAIL / 100.0
      CALL GENFLO ( COST, RLOR, RRID, R, RT, RCST, HCSR )

C ---- Check that the amount of grain shipped is more than 0.
C ---- If it is, update the total hauling (rail) cost
C ---- (SUMR) and the handling cost(SUMH).

      IF ( RT .LE. 0.0 ) GO TO 2300
      SUMR    = SUMR + RCST
      SUMH    = SUMH + HCSR

C ---- Write all the information about the deficit region
C ---- which received grain by rail.
C ---- DRGN(M) is the code number of the deficit region.
C ---- NAM2(M,L) is the name of the deficit region.
C ---- R(N) is the amount of grain received per shipment by
C ---- time. RT is the total shipment for this deficit region.
C ---- COST is the unit cost.
C ---- RCST is the total hauling(TRuck) cost.
C ---- HCSR is the handling cost.

      WRITE (FD,730) DRGN(M), ( NAM2(M,L), L = 1, 3 ),
      1           ( R(N)/OPERATION, N = 1, NOTF ),
      2           RT/OPERATION, COST*OPERATION, RCST, HCSR

C ---- Update the total amount of grain demanded(TODD) by
C ---- the deficit regions.
C ---- Update the total amount of grain shipped (TGRD(M))
C ---- to this deficit region.

      TODD     = TODD + RT
      TGRD(M) = TGRD(M) + RT
2300 CONTINUE

C ---- Selected BARGE (river)points-->by BarGE-->RIVER regions.
C ---- Read in the BarGe cost from each of the barge
C ---- unloading point to all the river regions.
C ---- If the cost(COST) is greater than 999 then it is
C ---- ignored else the cost by barge(COST) is calculated.
C ---- The COST includes BET and ALF.
C ---- Subroutine GENFLO is used to calculate the hauling

```

C ---- barge cost(BCST) and the handling cost (HCSB).  
C ---- T is the amount of grain in each shipment by time.  
C ---- TT is the amount of grain in total shipment.

```
DO 2500 M = 1, NRES
IDNT      = IDN1(M)
CALL SERIAL ( IDNT, NORE, RIVR, NT )
IF ( NT .EQ. NF ) GO TO 2500
IF ( NT .EQ. 0 ) GO TO 2500
COST      = RRBG(NF,M)
IF ( COST .GE. 999. ) GO TO 2500
COST      = COST * BARGE
CALL GENFLO ( COST, BLOR, BRIR, T, TT, BCST, HCSB )
```

C ---- Check that the amount of grain shipped is more than 0.  
C ---- If it is, update the total hauling (barge) cost  
C ---- (SUMB) and the handling cost(SUMH).

```
IF ( TT .LE. 0.0 ) GO TO 2500
SUMB      = SUMB + BCST
SUMH      = SUMH + HCSB
```

C ---- Write all the information about the river region  
C ---- which received grain by rail.  
C ---- RIVR(NT) is the code number of the deficit region.  
C ---- NAM3(NT,L) is the name of the deficit region.  
C ---- T(N) is the amount of grain received per shipment by  
C ---- time.  
C ---- TT is the total shipment for this river region.  
C ---- COST is the unit cost.  
C ---- BCST is the total hauling(TRuck) cost.  
C ---- HCSB is the handling cost.

```
WRITE (FD,740) RIVR(NT), ( NAM3(NT,L), L = 1, 3 ),
1           ( T(N)/OPERATION, N = 1, NOTF ),
2           TT/OPERATION, COST*OPERATION, BCST, HCSB
```

C ---- Update the total amount of grain shipped(TGRR(NT,N)) to  
C ---- a river region.

```
DO 2450 N = 1, NOTF
TGRR(NT,N)= TGRR(NT,N) + T(N)
2450 CONTINUE
2500 CONTINUE
```

C ---- Selected BARGE (port)points-->by BarGE-->RIVER regions.  
C ---- Read in the BarGe cost from each of the barge  
C ---- unloading point to all the river regions.  
C ---- If the cost(COST) is greater than 999 then it is  
C ---- ignored else the cost by barge(COST) is calculated.  
C ---- Subroutine GENFLO is used to calculate the hauling  
C ---- barge cost(BCST) and the handling cost (HCSB).  
C ---- R is the amount of grain in each shipment by time.  
C ---- RT is the amount of grain in total shipment.

```

DO 2800 M = 1, NPES
IDNT      = IDN2(M)
CALL SERIAL ( IDNT, NOPE, .PORT, NT )
IF ( NT .EQ. 0 ) GO TO 2800
COST      = RPBG(NF,M)
IF ( COST .GE. 999. ) GO TO 2800
COST      = COST * BARGE
CALL GENFLD ( COST, BLOR, BRIP, R, RT, BCST, HCSB )

C ---- Check that the amount of grain shipped(RT) is more than 0.
C ---- If it is, update the total hauling (barge) cost
C ---- (SUMB) and the handling cost(SUMH).

IF ( RT. LE. 0.0 ) GO TO 2800
SUMB      = SUMB + BCST
SUMH      = SUMH + HCSB

C ---- Write all the information about the port region
C ---- which received grain by barge.
C ---- PORT(NT) is the code number of the port region.
C ---- NAM4(NT,L) is the name of the port region.
C ---- R(N) is the amount of grain received per shipment by
C ---- time.
C ---- RT is the total shipment for this port region.
C ---- COST is the unit cost.
C ---- BCST is the total hauling(BARGE) cost.
C ---- HCSB is the handling cost.

WRITE (FD,750) PORT(NT), ( NAM4(NT,L), L = 1, 3 ),
1           ( R(N)/OPERATION, N = 1, NOTF ),
2           RT/OPERATION, COST*OPERATION, BCST, HCSB

C ---- Update the total amount of grain shipped(TGRP(NT,N)) to
C ---- a port region.

DO 2750 N = 1, NOTF
TGRP(NT,N)= TGRP(NT,N) + R(N)
2750 CONTINUE

2800 CONTINUE
2900 CONTINUE
3000 CONTINUE
RETURN
END

C -----
C          ===== PELVTR =====
C ---- Subroutine PELVTR deals with the port regions.
C ---- It finds the cost of transportation from the port
C ---- regions to selected barge (port) points and river.
C ---- Truck, rail and barge costs are considered.

SUBROUTINE PELVTR
DIMENSION T(4)

```

```

COMMON /A1/ NOSR, NODR, NORE, NOPE, NOFR
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /B4/ PORT(2D) /B5/ FRGN(25)
COMMON /C3/ PFSP(65,45)
COMMON /D1/ RAIL, TRUCK, BARGE, SHIP
COMMON /F3/ TLOS, RLOS, TLOR, RLOR, BLOR, SLOP
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP
COMMON /GD/ K, OPERATION /G1/ FLOW(5D000)
COMMON /GA/ SUNG, SUMT, SUMR, SUMB, SUMS, SUMH
COMMON /GB/ TOPR, TODD, TOFD
COMMON /GC/ TGSO(65), TGRD(65), TGRF(65)
COMMON /GD/ TGRR(65,4), TGRP(65,4)
COMMON /R1/ NAM1(65,3), NAM2(65,3), NAM3(45,3)
COMMON /R2/ NAM4(20,3), NAM5(25,3)
DIMENSION IDN3(16)
INTEGER U1,U2,U3,U4,U8,U9,UF1,FD

```

C ---- The layout of the output file, FD.

```

500 FORMAT ( 20A4 )
510 FORMAT ( 20I4 )
520 FORMAT ( 1DF8.3 )
700 FORMAT ( 1H1, //, 9X, 18HORIGIN/DESTN MODE, 6X, 6HSUPPLY, 14X,
    1       16HSHIPMENT BY TIME, 16X, 5HTOTAL, 6X, 4HUNIT,
    2       7X, 5HTOTAL, 5X, 8HHANDLING, /, 39X, 4I10, 4X,
    3       8HSHIPMENT, 5X, 28HCOST HAULING COST COSTS, // )
710 FORMAT ( /, 5X, 1HP, A4, 1X, 3A4, 16X, 4F10.0, / )
720 FORMAT ( 5X, 1HF, A4, 1X, 3A4, 2X, 2HS , 12X, 4F10.0,
    1       F12.0, F10.5, 2F12.0 )

```

C ---- Read in the alfas and betas of the deficit regions.

C ---- Read in the selected (port) barge shipping  
locations(IDEN(I)).

```

READ (U1,52D) ( ALFA(I), BETA(I), I = 1, NODR )
READ (U1,500) ( IDEN(I), I = 1, NODR )

```

C ---- PORT regions ----> by ShIP ----> FOREIGN regions  
C ---- Read in the ship rate (PFSP(I,J)) from each port to  
C ---- all the foreign regions.

```

DO 1100 I = 1, NOPE
READ (U4,520) ( PFSP(I,J), J = 1, NOFR )
1100 CONTINUE

```

C ---- Read in the number of ports (LAKE) in the Great Lakes  
C ---- for export and their code numbers(IDN3(I)).

```

READ (U4,510) LAKE
READ (U4,500) ( IDN3(I), I = 1, LAKE )

```

C ---- Print out the heading of the output file i,e,  
C ORIGIN/DESTN MODE SUPPLY SHIPMENT TOTAL UNIT TOTAL HANDLING  
C SHIPMENT COST HAULING COST COSTS

```

        WRITE (FD,700) ( I, I = 1, NOTF )

C ---- Print out the code number (PORT(I)) and the name (NAM4(I,L))
C ---- of the port together with the total grain shipment by time
C ---- from each port region.

        DO 2000 I = 1, NOPE
          NF      = I
        WRITE (FD,710) PORT(I), ( NAM4(I,L), L = 1, 3 ),
          1           ( TGRP(I,N)/OPERATION, N = 1, NOTF )

C ---- PORT regions ---> by Ship ---> FOREIGN regions.
C ---- Read in the ship rate from each of the port region
C ---- to all the foreign regions.
C ---- If the cost(COST) is greater than 999 then it is
C ---- ignored else the cost by truck(COST) is calculated.
C ---- Subroutine GENFLO is used to calculate the hauling
C ---- shipping cost(SCST) and the handling cost (HCSS).
C ---- T is the amount of grain in each shipment by time.
C ---- TT is the amount of grain in total shipment.
C ---- SLOP is the ship loading cost.
C ---- SHIP is the factor for ship rate and can be altered.

        00 1300 M = 1, NOFR
          NT      = M
          COST   = PFSP(NF,NT)
          IF ( COST .GE. 999. ) GO TO 1300
          COST   = COST * SHIP
          CALL GENFLO ( COST, SLOP, 0.0, T, TT, SCST, HCSS )

C ---- Check that the amount of grain shipped(TT) is more than 0.
C ---- If it is, update the total hauling (ship) cost
C ---- (SUMS) and the handling cost(SUMH).

          IF ( TT .LE. 0.0 ) GO TO 1300
          SUMS   = SUMS + SCST
          SUMH   = SUMH + HCSS

C ---- Write all the information about the foreign region
C ---- which received grain by ship.
C ---- FRGN(NT) is the code number of the foreign region.
C ---- NAM5(NT,L) is the name of the foreign region.
C ---- T(N) is the amount of grain received per shipment by
C ---- time.
C ---- TT is the total shipment for this river region.
C ---- COST is the unit cost.
C ---- SCST is the total hauling (SHIP) cost.
C ---- HCSS is the handling cost.

        WRITE (FD,720) FRGN(NT), ( NAM5(NT,L), L = 1, 3 ),
          1           ( T(N)/OPERATION, N = 1, NOTF ),
          2           TT/OPERATION, COST*OPERATION, SCST, HCSS

C ---- Update the total amount of grain shipped (TOFD) to all
C ---- foreign regions and the amount shipped (TGRF(NT,N)) to

```

```
C ---- this particular foreign region.
```

```
TOFD      = TOFD + TT  
TGRF(NT) = TGRF(NT) + TT
```

```
1300 CONTINUE  
2000 CONTINUE  
RETURN  
END
```

```
C -----  
C          ***** DEMAND *****  
C
```

```
SUBROUTINE DEMAND ( IEXP )  
DIMENSION A(5)  
COMMON /A1/ NOSR, NODR, MORE, NOPE, NOFR  
COMMON /A2/ NOTP, NOTF, NDAY(4)  
COMMON /B1/ SRGN(65) /B2/ DRGN(65) /B3/ RIVR(45)  
COMMON /B4/ PORT(20) /B5/ FRGN(25)  
COMMON /E2/ DDND(65) /E3/ FDND(25,4)  
COMMON /F3/ TL0S, RL0S, TL0R, RL0R, BL0R, SLOP  
COMMON /F4/ TRID, RRID, TRIR, RRIR, BRIR, TRIP, RRIP, BRIP  
COMMON /G0/ K, OPERATION /G1/ FLOW(50000)  
COMMON /GA/ SUMG, SUMT, SUMR, SUMB, SUMS, SUMH  
COMMON /GB/ TOPR, TODD, TOFD  
COMMON /GC/ TGSO(65), TGRD(65), TGRF(65)  
COMMON /GD/ TGRR(65,4), TGRP(65,4)  
COMMON /R1/ NAM1(65,3), NAM2(65,3), NAM3(45,3)  
COMMON /R2/ NAM4(20,3), NAM5(25,3)  
INTEGER     FLOW  
INTEGER U1,U2,U3,U4,U8,U9,UF1,FD
```

```
C ---- The layout of the output.
```

```
520 FORMAT ( 10F8.3 )  
700 FORMAT ( 1H1, //, 52X, 14HDEMAND BY TIME, 17X, 5HTOTAL,  
           1           /, 39X, 4I10, 2X, 8HSHIPMENT, / )  
720 FORMAT ( 5X, 1HD, A4, 1X, 3A4, 16X, 4F10.0, F12.0 )  
730 FORMAT ( 5X, 1HF, A4, 1X, 3A4, 16X, 4F10.0, F12.0 )
```

```
C ---- Check that the number of deficit regions are greater than 0.  
C ---- If it is then read in the demand of each of the deficit  
C ---- region (DDND(I)).
```

```
IF ( NODR .LE. 0 ) GO TO 1400  
READ  (U8,520) ( DDND(I), I = 1, NODR )
```

```
C ---- Print out the heading : DEMAND BY TIME      TOTAL SHIPMENT
```

```
WRITE (FD,700) ( I, I = 1, NOTF )
```

```
C ---- Find the demand by time of each deficit region A(N).
```

```
DO 1300 I = 1, NODR
```

```

DO 1200 N = 1, NOTF
K          = K + 1
A(N)      = FLOW(K)
1200 CONTINUE

C ---- Write out the code number (DRGN(I)) and name (NAM2(I,L))
C ---- of each deficit region and their demand by time (A(N)).

K          = K + 1
A(5)      = FLOW(K)
WRITE (FD,720) DRGN(I), ( NAM2(I,L), L = 1, 3 ),
1           (A(N)/OPERATION, N = 1, 5 )
1300 CONTINUE

C ---- Check that the number of foreign regions is greater than 0.
C ---- If it is then read in the demand by time of each foreign
C ---- region FDND(I,N)).

1400 IF ( NOFR .LE. 0 ) RETURN
DO 1500 I = 1, NOFR
READ (U8,520) ( FOND(I,N), N = 1, NOTF )
1500 CONTINUE

C ---- Find the demand by time of each foreign region A(N).

00 1700 I = 1, NOFR
00 1600 N = 1, NOTF
K          = K + 1
A(N)      = FLOW(K)
1600 CONTINUE

C ---- Write out the code number (FRGN(I)) and name (NAM5(I,L))
C ---- of each foreign region and their demand by time (A(N)).

K          = K + 1
A(5)      = FLOW(K)
WRITE (FO,730) FRGN(I), ( NAM5(I,L), L = 1, 3 ),
1           (A(N)/OPERATION, N = 1, 5 )
1700 CONTINUE
RETURN
ENO

C -----
C ===== SERIAL =====
C ---- This subroutine is used to check that the data inputted
C ---- is correct.

SUBROUTINE SERIAL ( IONT, NOSR, SRGN, NF )
DIMENSION SRGN(NOSR)
INTEGER SRGN
DATA      NINE/' 999'/
600 FORMAT ( 5X, '????? ERROR IN DATA. ', A4, 'IS MISSING' )
NF      = 0
00 1100 I = 1, NOSR

```

```

      IF ( IDNT .EQ. SRGN(I) ) GO TO 1200
1100 CONTINUE
      IF ( IDNT .NE. NINE ) WRITE (FD,600) IDNT
      RETURN
1200 NF      = I
      RETURN
      END

C----- **** GENFLD ****
C ---- Calculate the amount of grain flow per time period,
C ---- total hauling cost(TCST) and handling cost (HCST).

SUBROUTINE GENFLD ( COST, COUT, CRIN, A, AT, TCST, HCST )
DIMENSION A(4)
COMMON /A2/ NOTP, NOTF, NDAY(4)
COMMON /G0/ K, OPERATION /G1/ FLOW(50000)
INTEGER FLOW
AT      = 0.0
TCST   = 0.0
HCST   = 0.0
DO 1100 N = 1, NOTF
      K = K + 1
      A(N) = FLOW(K)
      AT = AT + A(N)
1100 CONTINUE

C ---- To find the total hauling cost(TCST) e.g., for Abilene,
C ---- TCST = 5246*0.20952*1000

      TCST     = AT * COST * 1000.0

C ---- COUT and CRIN are the loading and unloading costs.
C ---- The handling cost, for Abilene,
C ---- HCST = 5246 * (6.47 + 6.59) * 10

      HCST     = AT * ( COUT + CRIN ) * 10.00
      RETURN
      END

```

C

## F1.SFT

C Note :

- (1) Do not use this file to run the program.  
This is a documented version of the actual file used.
- (2) The letter C at the beginning of a sentence implies that it is a comment or an explanation. Lines beginning with a C are not present in the actual file used.
- (3) Explanation appears before the actual code.
- (4) Any new changes made must be in the same position as the old entry.
- (5) If the entries on a line does not exceed the 80th column, insert a digit zero, 0, on the 80th column.
- (6) The format used for codes in the model is :  
4 fields i.e. \*ccc where c is a digit or a letter,  
e.g. \*601 where \* is a space.

C-----

C      Explanation for f1.sft.

C      The explanation for each line of f1.sft is given below.

C-----

C      This line show the title heading used in the output file, file06\_2.sft.  
C      file06\_2.sft is used to display the intermediate results.  
C      If a new title is to be needed, change this sentence.

SOFT WHEAT SHIPMENT BAIE COMEAU ADDED, FULLER, GRANT, TEH & FELLIN

0

C-----

C      This line shows the number of regions involved in this model.  
C      If there were any changes in the number of regions, the corresponding  
C      number must be changed.  
C      e.g. if the number of surplus regions were increased to 42 then change 41  
C      to 42.

C      There are limitations on the number of regions that are allowed in  
C      this program :

C           Maximum number of surplus regions allowed = 70  
C           Maximum number of deficit regions allowed = 70  
C           Maximum number of river regions allowed = 45  
C           Maximum number of port regions allowed = 20  
C           Maximum number of foreign regions allowed = 25

C      In this model, the

C           Number of surplus regions = 41  
C           Number of deficit regions = 18  
C           Number of river regions = 43  
C           Number of port regions = 20  
C           Number of foreign regions = 25

1

C      The format used is :

C      4 fields i.e. \*\*dd where d is a digit.

C e.g. \*\*21 where \* is a space.

41 18 43 20 25

0

C-----

C This line shows the number of periods and days per period in this  
C model.

C There are 3 periods per year.

C The number of days in each period :

C        number of days in the 1st period = 121

C        number of days in the 2nd period = 121

C        number of days in the 3rd period = 123

C If the number of periods were different, alter 3 to the new number and  
C add the number of days in the new period.

C The format used for the number of days is :

C 4 fields i.e. \*ddd where d is a digit.

C e.g. \*121 where \* is a space.

3 121 121 123

0

C-----

C The 3 lines show the codes for the 41 surplus regions in this model.

C For example, 111 is the code number for Alexandria LA , and

C 085 is the code number for Ames IA.

C If there were any changes in the number of surplus regions, the code of  
C the new surplus region would be added to the end of the list.

C If there were a deletion, then the code involved would be deleted from  
C this list.

111 085 171 154 351 133 261 251 092 094 144 131 099 134 141 065 201 132 312 191  
192 142 181 638 123 212 093 135 182 331 104 143 242 709 713 311 152 231 103 102  
313

0

C-----

C This line shows the codes for the 18 deficit regions in this model.

C For example, 211 is the code number for Atlanta GA,

C 163 is the code number for Battle Cr MI.

C If there were any changes in the number of deficit regions, the code of  
C the new deficit region would be added to the end of the list.

C If there were a deletion, then the code involved would be deleted from  
C this list.

211 163 098 153 391 281 183 222 715 096 053 381 291 151 043 045 371 241

0

C-----

C These 3 lines show the codes for the 43 river barge ports in this  
C model.

C For example, 601 is the code number for St Paul MN, and  
C 602 is the code number for Winona MN.

C If there were any changes in the number of river barge ports, the code  
C of the new river port would be added to the end of the list.  
C If there were a deletion, then the code involved would be deleted from  
C this list.

601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620  
621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640  
641 642 643 0

C-----

C This line shows the codes for the 20 ports in this model.  
C For example, 701 is the code number for Mobile AL, and  
C 702 is the code number for New Orleans LA.

C If there were a change in the number of port regions, the code of  
C the new port would be added to the end of the list.  
C If there were a deletion, then the code involved would be deleted from  
C this list.

701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720

C-----

C These 2 lines codes for the 25 foreign regions in this model.  
C For example, 801 is the code number for Scandinavia and  
C 802 is the code number for NC Europe.

C If there were any changes in the number of foreign regions, the code of  
C the new foreign region would be added to the end of the list. Make sure  
C that the number of regions does not exceed the maximum number allowed  
C in this region. The maximum number of foreign regions is 25.  
C If there were a deletion, then the code involved would be deleted from  
C this list.

801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820  
821 822 823 824 825 0

C-----

C This line shows the loading costs by truck, rail and  
C barge (cents/bushel) in this model.  
C If there were any changes in the loading costs, then alter the  
C corresponding loading cost.

C Truck Load Country Elevation = 10.422  
C Rail Load Country Elevation = 10.368  
C Truck Load River Location = 8.940  
C Rail Load River Location = 9.330  
C Barge Load River Location = 7.758  
C Ship Load Port Location = 5.660

C The format used is :  
C 8 fields i.e. dddd.ddd where d is a digit.  
C If there is no digit in front of the number, then add spaces.  
C e.g. 10.422 becomes \*\*10.422 where \* is a space.

10.422 10.368 8.940 9.330 7.758 5.660 0

C-----

C This line shows the unloading costs by truck, rail and barge  
C (cents/bushel) in this model.  
C If there were any changes in the unloading cost, then change the  
C affected cost.

C River Unload Truck Surplus = 8.520  
C River Unload Rail Surplus = 6.774  
C River Load Truck Deficit = 6.450  
C River Load Rail Deficit = 6.318  
C River Unload Barge = 11.076  
C Port Unload Truck = 6.763  
C Port Unload Rail = 4.082  
C Port Unload Barge = 8.097

C The format used is :  
C 8 fields i.e. dddd.ddd where d is a digit.  
C If there is no digit in front of the number, then add spaces.  
C e.g. 8.520 becomes \*\*\*8.520 where \* is a space.

8.520 6.774 6.450 6.318 11.076 6.763 4.082 8.097 0

C-----

C The quantity of storage available in each surplus region.  
C If a new surplus region were added, then add one value to the end of  
C the list.  
C If there were deletion, then erase one from this list.

C The format used is :  
C 8 fields i.e. dddddd.d where d is a digit.  
C e.g. 999999.9

999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.9  
999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.9  
999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.9  
999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.999999.9  
999999.9 0

C-----

C The grain storage cost, rail, truck, barge, and ship adjustment  
C factors and the period chosen to be blocked are as follows:

C storage cost factor (cents/bushel/year) = 11.566  
C rail cost multiplier = 1.000

C                   truck mileage multiplier                           = 1.000  
C                   barge cost multiplier                           = 1.000  
C                   ship rate multiplier                           = 1.000  
C                   period blocked                                   = 1st period.

C       If there is a need to test the effect of an increase or a decrease in  
C       cost of a type of transport, then alter the value of the multiplier.

C       The format used is :  
C       8 fields i.e. dddd.ddd where d is a digit.  
C       If there is no digit in front of the number, then add spaces.  
C       e.g. 1.000 becomes \*\*\*1.000 where \* is a space.

11.566   1.000   1.000   1.000   1.000   1.000                   0

C-----

C       Conversion factor and choice of measurement

C       The conversion factor                                   = 1.000  
C       Choice of measurement                                   = 1.000  
C       Conversion factor for short ton is 2000/60 = 33.333  
C       and for metric ton is 2204/60 = 36.733

C       Choice of measurement can be one of these:

C           1.000 is for bushel  
C           2.000 is for short ton  
C           3.000 is for metric ton

1.000   1.000   0

C-----

C       Alphas and betas associated with each surplus region and represent  
C       truck cost intercept and per unit costs. (currently not being used).

5.654	0.191	5.654	0.191	6.486	0.208	6.339	0.205	5.654	0.191
6.339	0.205	5.654	0.191	6.460	0.207	7.246	0.233	7.246	0.233
6.339	0.205	6.508	0.208	7.055	0.224	6.508	0.208	6.339	0.205
5.654	0.191	6.434	0.207	7.246	0.233	5.654	0.191	6.508	0.208
5.654	0.191	5.654	0.191	6.486	0.208	6.339	0.205	5.654	0.191
6.339	0.205	5.654	0.191	6.460	0.207	7.246	0.233	7.246	0.233
6.339	0.205	6.508	0.208	7.055	0.224	6.508	0.208	6.339	0.205
5.654	0.191	6.434	0.207	7.246	0.233	5.654	0.191	6.508	0.208
6.508	0.208								0

C-----

C       These lines show the codes for barge shipping ports, with each  
C       code number (barge loading port) associated with a surplus region from  
C       it receives grain.

C       If there were any changes in the number of surplus regions, then there  
C       must be an associated addition or deletion of the code representing the  
C       barge loading point for that surplus region.

C Adding a surplus region means that the code of the barge loading port  
C linked to the surplus region must be included at the end of the list.

C A deletion means that the code of the barge loading port linked to the  
C surplus region must be removed.

C If the surplus region is not linked with a barge loading point then its  
C code number is 999. 999 means that the route from the surplus region to  
C the barge loading port is not feasible. Thus there is no link between  
C the 2 regions.

C e.g., the 1st entry is the surplus region, Alexandria LA.

C Alexandria LA is NOT linked to any barge loading port.

C e.g., Ames IA (the 2nd code number) is linked to barge loading region,  
C Clinton IA (605).

999 605 625 618 999 608 999 618 612 613 619 615 621 608 618 636 628 606 640 634  
635 619 624 638 604 627 607 621 625 639 632 616 999 605 643 640 618 999 632 632  
639 0

C-----

C Alphas and betas associated with each deficit region and truck cost  
C intercept and per unit cost. (currently not used).

5.866	0.195	5.654	0.191	6.200	0.202	6.200	0.202	6.508	0.208
5.866	0.195	5.654	0.191	6.200	0.202	6.200	0.202	6.508	0.208
5.866	0.195	5.654	0.191	6.200	0.202	6.200	0.202	6.508	0.208
7.246	0.233	5.654	0.191	6.200	0.202				0

C-----

C The line below shows the codes for barge shipping ports with each code  
C number (barge unloading port) associated with a deficit region to which  
C it sends grain.

C For example, the 1st deficit region, Atlanta GA (211) is linked to  
C Chattanooga TN (627) and  
C the 2nd deficit region, Battle Cr (163) is NOT linked to  
C any barge unloading port.

C If there were any changes in the number of deficit regions, then there  
C must be associated addition or deletion of the code representing the  
C barge-unloading point for that deficit deficit region.  
C Adding a deficit region means that the code of the barge-unloading port  
C linked to the deficit region must be included at the end of the list.

C A deletion means that the code of the barge-unloading port linked to  
C the deficit region must be removed.

627 999 621 618 999 618 626 999 999 999 631 999 618 618 613 613 999 999 0

C-----

C The number of shipping points in this model.

C           Number of barge receiving points = 5  
C           Number of ports which receive from barge = 5

5    5

0

C-----

C       The codes for the above shipping points in this model.

C       The 5 river shipping locations are :

C           Buffalo, NY (645)  
C           Knoxville, TN (626)  
C           Chatanoo, TN (627)  
C           Guntersv, AL (628)  
C           Florence, AL (629)

C       The 5 river port shipping points are:

C           New Orle, LA (702)  
C           Mobile, AL (701)  
C           Galvestn, TX (703)  
C           Chicago, IL (710)  
C           Portland, OR (713)

645 626 627 628 629 702 701 703 710 713

0

C-----

F2.SFT

C Stores all the truck mileages in this model.  
C F2.SFT is the 2nd file in this model.

C In this model, four sets of truck mileages must be estimated.  
C This includes  
C (1) miles from each surplus region to each deficit region  
C (2) miles from each surplus region to a barge-loading river location  
C (3) miles from each surplus region to each port location  
C (4) miles from each deficit region to a barge-unloading location.

C Diagrammatically,

(3)

surplus regions -----> port

(2)

surplus regions -----> river barge  
loading regions

(4)

port barge -----  
unloading regions -----> deficit regions

(1)

-----> deficit regions

C A zero '0' must be inserted on the 80th column on each line.  
C The letter 'C' at the start of a statement is an explanation or  
C a comment. It is not in the actual file used in the programs.

C The format used is :  
C 8 fields i.e. dddd.ddd where d is a digit.  
C If there is no digit in front of the number, then add spaces.  
C e.g. 100.443 becomes \*100.443 where \* is a space.

C-----

C First part

C (1) Each surplus region is linked to each of the deficit regions.  
C In this model there are 41 surplus regions and 18 deficit regions.

C surplus region 1 -----> deficit region 1  
C -----> deficit region 2  
C .  
C .  
C -----> deficit region 18

C surplus region 2 -----> deficit region 1  
C -----> deficit region 2  
C .  
C .  
C -----> deficit region 18

C .  
C .  
C .

C      surplus region 41    -----> deficit region 1  
C                             -----> deficit region 2  
C .  
C .  
C                             -----> deficit region 18

C-----

C      Second part

C      (2) Each surplus region is linked to a corresponding river barge-loading location.

C      surplus region 1    -----> corresponding river barge loading region  
C      surplus region 2    -----> corresponding river barge loading region  
C .  
C .  
C      surplus region 41    -----> corresponding river barge loading region

C-----

C      Third part

C      (3) Each surplus region is linked to all the port locations for export.  
C      There are 20 ports in this model.

C      surplus region 1    -----> port region 1 for export  
C                             -----> port region 2 for export  
C .  
C .  
C                             -----> port region 20 for export

C      surplus region 2    -----> port region 1 for export  
C                             -----> port region 2 for export  
C .  
C .  
C                             -----> port region 20 for export

C .  
C .  
C .

C      surplus region 41    -----> port region 1 for export  
C                             -----> port region 2 for export  
C .  
C .  
C                             -----> port region 20 for export

C-----

C      Fourth part

C      (4) Each barge-unloading river location is linked to a  
C      corresponding deficit region.

C      barge-unloading location 1 -----> corresponding deficit region

C      barge-unloading location 2 -----> corresponding deficit region

C      .

C      .

C      .

C      barge-unloading location 41 -----> corresponding deficit region

C-----

C      Explanation for 1st part

C      This includes the truck mileage from each of the 41 surplus regions  
C      to each of the 18 deficit regions.

C      A 41 by 18 matrix.

C      The first surplus region is Alexandria LA(111).

C      The distances from Alexandria to all deficit regions are given on the first  
C      2 lines (see below):

C           Alexandria LA(111) to Atlanta GA(211) = 562 miles

C           Alexandria LA(111) to Battle Cr MI(163) = 978 miles

C      The next matrix is for the next surplus region and its mileages to all  
C      the deficit regions.

C      If a new surplus region were added to the end of the surplus region list,  
C      then its truck mileages to all the deficit regions would be added to the  
C      end of this 1st part.

C      If a new deficit region were added to position 10, then its truck mileage  
C      to all the deficit regions would be positioned of the 10th location in  
C      the matrix.

C      If a surplus region were deleted, then its truck mileage to all the  
C      deficit regions would be removed from this 1st part.

C      e.g., if Alexandria were not a surplus region, then the first 2 lines would  
C      be removed. The file would start with 900.

562.000	978.000	478.000	888.000	1084.000	1191.000	666.000	661.000	1694.000	479.000
676.000	1549.000	1381.000	978.000	681.000	627.000	1900.000	869.000	0.000	
900.000	475.000	425.000	652.000	684.000	815.000	824.000	1186.000	1742.000	577.000
227.000	1078.000	980.000	580.000	289.000	425.000	1434.000	996.000	0.000	
301.000	416.000	364.000	292.000	1133.000	595.000	203.000	587.000	2073.000	726.000
432.000	1613.000	785.000	382.000	598.000	691.000	1969.000	437.000	0.000	
605.000	286.000	653.000	99.000	1316.000	205.000	415.000	840.000	2319.000	984.000
592.000	1777.000	428.000	143.000	799.000	930.000	2105.000	368.000	0.000	
1574.000	1189.000	1028.000	1366.000	275.000	1529.000	1504.000	1859.000	1108.000	861.000
989.000	412.000	1694.000	1294.000	732.000	731.000	759.000	1713.000	0.000	
589.000	266.000	335.000	304.000	957.000	558.000	479.000	876.000	1984.000	666.000
275.000	1409.000	728.000	310.000	455.000	589.000	1739.000	651.000	0.000	
482.000	393.000	657.000	120.000	1371.000	313.000	315.000	690.000	2361.000	1009.000

613.000	1836.000	550.000	252.000	831.000	956.000	2160.000	223.000	0.000	
496.000	582.000	892.000	355.000	1597.000	274.000	370.000	672.000	2572.000	1223.000
900.000	2039.000	439.000	439.000	1066.000	1191.000	2395.000	188.000	0.000	
781.000	557.000	263.000	608.000	637.000	864.000	711.000	1067.000	1679.000	439.000
208.000	1105.000	1033.000	615.000	151.000	287.000	1461.000	935.000	0.000	
743.000	626.000	167.000	652.000	671.000	917.000	692.000	998.000	1633.000	347.000
136.000	1164.000	1091.000	673.000	136.000	237.000	1520.000	934.000	0.000	
449.000	245.000	427.000	182.000	1106.000	470.000	302.000	735.000	2093.000	758.000
362.000	1577.000	654.000	239.000	573.000	704.000	1904.000	510.000	0.000	
718.000	217.000	442.000	395.000	943.000	557.000	583.000	1004.000	2001.000	773.000
334.000	1337.000	722.000	323.000	510.000	646.000	1693.000	739.000	0.000	
467.000	567.000	116.000	507.000	951.000	810.000	428.000	740.000	1825.000	478.000
146.000	1439.000	987.000	569.000	419.000	443.000	1795.000	670.000	0.000	
508.000	335.000	266.000	323.000	951.000	588.000	421.000	794.000	1930.000	597.000
201.000	1436.000	762.000	344.000	412.000	543.000	1767.000	639.000	0.000	
593.000	97.000	522.000	182.000	1143.000	360.000	406.000	879.000	2174.000	853.000
457.000	1555.000	527.000	110.000	649.000	783.000	1885.000	529.000	0.000	
826.000	1078.000	516.000	1032.000	757.000	1335.000	868.000	966.000	1356.000	200.000
629.000	1211.000	1513.000	1095.000	491.000	359.000	1562.000	1110.000	0.000	
105.000	647.000	471.000	487.000	1306.000	790.000	203.000	382.000	2083.000	734.000
605.000	1786.000	982.000	601.000	771.000	798.000	2142.000	400.000	0.000	
711.000	320.000	339.000	441.000	856.000	1460.000	622.000	998.000	1918.000	664.000
282.000	1276.000	824.000	407.000	396.000	532.000	1608.000	793.000	0.000	
2425.000	1980.000	1892.000	2158.000	1194.000	2320.000	2347.000	2711.000	1205.000	1762.000
1852.000	794.000	2485.000	2086.000	1609.000	1629.000	626.000	2517.000	0.000	
374.000	765.000	330.000	659.000	1131.000	962.000	493.000	571.000	1827.000	517.000
395.000	1621.000	1152.000	749.000	633.000	641.000	1968.000	691.000	0.000	
384.000	883.000	491.000	759.000	1258.000	1062.000	488.000	453.000	1880.000	651.000
689.000	1735.000	1252.000	849.000	790.000	797.000	2086.000	691.000	0.000	
493.000	204.000	404.000	185.000	1064.000	450.000	346.000	779.000	2067.000	735.000
339.000	1533.000	624.000	206.000	550.000	681.000	1860.000	530.000	0.000	
326.000	580.000	257.000	474.000	1088.000	777.000	303.000	599.000	1902.000	552.000
287.000	1585.000	967.000	564.000	559.000	584.000	1930.000	551.000	0.000	
2360.000	1915.000	1827.000	2093.000	1119.000	2510.000	2282.000	2646.000	1111.000	1688.000
1787.000	641.000	2420.000	2021.000	1544.000	1564.000	515.000	2442.000	0.000	
826.000	300.000	525.000	478.000	959.000	642.000	682.000	1113.000	2063.000	828.000
465.000	1371.000	805.000	406.000	540.000	676.000	1704.000	824.000	0.000	
158.000	846.000	734.000	643.000	1556.000	810.000	351.000	140.000	2263.000	997.000
855.000	2036.000	984.000	769.000	1021.000	1061.000	2392.000	360.000	0.000	
697.000	498.000	198.000	549.000	719.000	805.000	627.000	983.000	1728.000	475.000
133.000	1187.000	974.000	556.000	202.000	336.000	1543.000	858.000	0.000	
461.000	401.000	249.000	375.000	944.000	654.000	391.000	747.000	1921.000	580.000
184.000	1437.000	828.000	410.000	402.000	527.000	1773.000	641.000	0.000	
242.000	474.000	372.000	350.000	1156.000	653.000	177.000	528.000	2025.000	676.000
455.000	1636.000	843.000	440.000	621.000	699.000	1992.000	419.000	0.000	
2390.000	2052.000	1843.000	2197.000	1068.000	2392.000	2320.000	2694.000	1010.000	1630.000
1805.000	1322.000	2557.000	2151.000	1540.000	1528.000	419.000	737.000	0.000	
515.000	781.000	247.000	693.000	980.000	996.000	529.000	722.000	1711.000	371.000
312.000	1471.000	1186.000	783.000	488.000	495.000	1822.000	771.000	0.000	
435.000	347.000	328.000	296.000	1023.000	591.000	342.000	722.000	2000.000	659.000
265.000	1516.000	767.000	349.000	481.000	606.000	1852.000	564.000	0.000	
428.000	751.000	935.000	491.000	1728.000	432.000	391.000	552.000	2602.000	1244.000
934.000	2207.000	575.000	610.000	1173.000	1262.000	2531.000	149.000	0.000	
775.000	117.000	710.000	318.000	1274.000	417.000	588.000	1061.000	2348.000	1041.000
645.000	1686.000	582.000	189.000	817.000	951.000	2014.000	664.000	0.000	

2619.0002281.0002072.0002426.0001293.0002621.0002549.0002903.000	964.0001859.000
2034.000 794.0002786.0002380.0001769.0001757.000	484.0002777.000 0.000
2340.0001895.0001807.0002073.0001108.0002235.0002262.0002626.0001212.0001677.000	
1767.000 721.0002400.0002001.0001524.0001544.000	621.0002431.000 0.000
512.000 327.000 267.000 69.0001192.000	322.000 325.000 798.0002195.000 598.000
467.0001655.000 803.000 385.000 373.000	507.0001680.000 415.000 0.000
259.000 785.000 834.000 512.0001646.000	619.000 308.000 341.0002450.0001097.000
845.0002139.000 786.000 644.0001095.0001161.0002466.000	178.000 0.000
628.000 904.000 355.000 827.000	897.0001130.000 663.000 803.0001562.000 290.000
544.0001377.0001320.000 906.000	492.000 438.0001728.000 905.000 0.000
462.000 639.000 114.000 575.000	949.000 878.000 452.000 717.0001783.000 434.000
179.0001444.0001059.000 641.000	417.000 441.0001795.000 694.000 0.000
2541.0002064.0001976.0002242.0001266.0002418.0002458.0002828.0001159.0001827.000	
1959.000 785.0002569.0002170.0001693.0001713.000	616.0002600.000 0.000

C-----

C Explanation for the Second part

C This links each surplus region (41) to its corresponding river barge-loading locations. Conceptually, a (1 x 41) matrix.

C Diagrammatically,  
C surplus region 1 -----> corresponding river barge-loading location  
C surplus region 2 -----> corresponding river barge-loading location  
C  
C  
C surplus region 41 -----> corresponding river barge-loading location

C e.g., 2nd surplus region, Ames IA (085) to 5th river barge region,  
C Clinton IA(605) = 176 miles

C e.g., 3rd surplus region, Bowling G KY(171) to 25th river barge region,  
C Nashville TN(625) = 061 miles

C If a new surplus region were added to the end of the surplus region list,  
C then its truck mileage to the corresponding river barge region would be  
C added to the end of this list.

C If a new surplus region were added to position 10, then its truck mileage  
C to the corresponding river barge-loading location would be at position 10  
C in this list.

C If a surplus region were deleted, then its truck mileage to the  
C river barge-loading location would be removed from this list.  
C e.g., if Alexandria LA were not a surplus region, then 999 would be deleted.

C If a river barge-loading location were deleted, make sure that no surplus  
C region is linked to it.

999.000 176.000 61.000 183.000 999.000 166.000 999.000 443.000 74.000 79.000	
67.000 58.000 77.000 97.000 153.000 316.000 34.000 49.000 10.000 55.000	
131.000 111.000 84.000 25.000 97.000 272.000 66.000 97.000 2.000 65.000	
2.000 31.000 999.000 407.000 47.000 137.000 299.000 999.000 150.000 157.000	
132.000	0.000

C-----  
C       Explanation for the Third part

C       (3) Each surplus region(41) is linked to all 20 port locations for  
C       export.

C       Diagrammatically,

C       surplus region 1    -----> port region 1 for export  
C                              -----> port region 2 for export

C                             .  
C                             .  
C                             -----> port region 20 for export

C       surplus region 2    -----> port region 1 for export  
C                              -----> port region 2 for export

C                             .  
C                             .  
C                             -----> port region 20 for export

C       surplus region 41    -----> port region 1 for export  
C                              -----> port region 2 for export

C                             .  
C                             .  
C                             -----> port region 20 for export

C       There are 41 surplus regions and 20 port locations thus a (41 by 20)  
C       matrix.

C       e.g., 1st surplus region, Alexandria LA(064) to 1st port region,  
C       Mobile AL           (701) = 309 miles

C       e.g., 1st surplus region, Alexandria LA(064) to 2nd port region,  
C       New Orleans LA (702) = 189 miles

C       The next matrix is for the next surplus region and its mileages to all  
C       the port regions.

C       If a new surplus region were added to the end of the surplus region list,  
C       then its truck mileages to all the port regions would be added to the list

C       If a new surplus region were added to position 10, then its truck mileage  
C       to all the port regions would be positioned at the 10th position in the  
C       matrix.

C       If a surplus region were deleted, then its truck mileages to all the  
C       port regions would be removed.

C       e.g., if Alexandria LA (111) were removed as a surplus region, then the  
C       first 2 lines (below) would be removed. The file would start with 984.

C       If a port region were deleted, make sure that there is no surplus  
C       region still linked to the deleted port region.

309.000 189.000 229.000 450.000 585.000 839.000 1185.000 993.000 1092.000 869.000  
 1226.000 2385.000 2316.000 2060.000 1694.000 1638.000 993.000 1092.000 869.000 1226.000  
 984.000 1011.000 987.000 1106.000 1241.000 1174.000 983.000 547.000 592.000 322.000  
 369.000 1724.000 1788.000 1830.000 1742.000 1791.000 547.000 592.000 322.000 369.000  
 498.000 578.000 829.000 1036.000 1171.000 575.000 693.000 411.000 526.000 393.000  
 851.000 2348.000 2336.000 2365.000 2073.000 2048.000 411.000 526.000 393.000 851.000  
 882.000 979.000 1216.000 1425.000 1560.000 635.000 312.000 182.000 322.000 383.000  
 852.000 2413.000 2476.000 2532.000 2319.000 2299.000 182.000 322.000 383.000 852.000  
 1600.000 1540.000 1340.000 1352.000 1480.000 1868.000 1700.000 1264.000 1310.000 1038.000  
 913.000 1116.000 1096.000 1167.000 1108.000 1157.000 1264.000 1310.000 1038.000 913.000  
 723.000 786.000 965.000 1162.000 1297.000 834.000 685.000 310.000 383.000 189.000  
 572.000 2110.000 2147.000 2164.000 1984.000 1964.000 310.000 383.000 189.000 572.000  
 800.000 908.000 1158.000 1367.000 1502.000 490.000 390.000 302.000 441.000 461.000  
 930.000 2491.000 2554.000 2591.000 2361.000 2341.000 302.000 441.000 461.000 930.000  
 831.000 966.000 1257.000 1478.000 1613.000 427.000 149.000 474.000 615.000 681.000  
 1146.000 2694.000 2762.000 2791.000 2572.000 2538.000 474.000 615.000 681.000 1146.000  
 829.000 841.000 849.000 968.000 1103.000 1061.000 986.000 607.000 674.000 419.000  
 539.000 1857.000 1828.000 1857.000 1679.000 1655.000 607.000 674.000 419.000 539.000  
 733.000 727.000 747.000 866.000 1001.000 1032.000 1030.000 676.000 743.000 488.000  
 653.000 1918.000 1887.000 1903.000 1633.000 1609.000 676.000 743.000 488.000 653.000  
 674.000 768.000 992.000 1201.000 1336.000 666.000 583.000 258.000 357.000 231.000  
 700.000 2261.000 2315.000 2332.000 2093.000 2073.000 258.000 357.000 231.000 700.000  
 862.000 923.000 1100.000 1268.000 1403.000 933.000 725.000 289.000 334.000 61.000  
 433.000 1980.000 2050.000 2089.000 2001.000 2014.000 289.000 334.000 61.000 433.000  
 506.000 533.000 671.000 839.000 974.000 756.000 908.000 582.000 681.000 443.000 811.000  
 811.000 2191.000 2162.000 2133.000 1850.000 1801.000 582.000 681.000 443.000 811.000  
 651.000 714.000 893.000 1090.000 1225.000 790.000 701.000 357.000 454.000 206.000  
 637.000 2146.000 2174.000 2191.000 1930.000 1910.000 357.000 454.000 206.000 637.000  
 823.000 917.000 1117.000 1324.000 1459.000 771.000 518.000 104.000 202.000 155.000  
 624.000 2185.000 2248.000 2310.000 2176.000 2156.000 104.000 202.000 155.000 624.000  
 621.000 527.000 311.000 385.000 520.000 1103.000 1387.000 1108.000 1195.000 941.000  
 1110.000 2047.000 1978.000 1722.000 1356.000 1300.000 1108.000 1195.000 941.000 1110.000  
 296.000 401.000 692.000 913.000 1048.000 394.000 722.000 630.000 757.000 627.000  
 1079.000 2537.000 2509.000 2391.000 2083.000 2035.000 630.000 757.000 627.000 1079.000  
 808.000 861.000 1016.000 1192.000 1327.000 977.000 811.000 395.000 442.000 179.000  
 480.000 1968.000 2014.000 2031.000 1918.000 1898.000 395.000 442.000 179.000 480.000  
 2486.000 2459.000 2259.000 2251.000 2379.000 2730.000 2501.000 2065.000 2071.000 1835.000  
 1456.000 334.000 335.000 887.000 1205.000 1301.000 2065.000 2071.000 1835.000 1456.000  
 265.000 276.000 476.000 697.000 832.000 681.000 991.000 778.000 876.000 657.000  
 1072.000 2443.000 2402.000 2156.000 1827.000 1765.000 778.000 876.000 657.000 1072.000  
   97.000 108.000 409.000 630.000 765.000 661.000 1007.000 878.000 993.000 804.000  
 1226.000 2559.000 2490.000 2246.000 1880.000 1824.000 878.000 993.000 804.000 1226.000  
 718.000 806.000 1001.000 1208.000 1343.000 710.000 564.000 220.000 318.000 187.000  
 656.000 2217.000 2271.000 2288.000 2067.000 2047.000 220.000 318.000 187.000 656.000  
 377.000 440.000 649.000 858.000 993.000 632.000 821.000 594.000 692.000 480.000  
 903.000 2355.000 2336.000 2221.000 1902.000 1878.000 594.000 692.000 480.000 903.000  
 2411.000 2375.000 2127.000 2098.000 2226.000 2655.000 2426.000 1990.000 1996.000 1760.000  
 1381.000 320.000 335.000 880.000 1111.000 1195.000 1990.000 1996.000 1760.000 1381.000  
 948.000 1011.000 1190.000 1357.000 1492.000 1037.000 808.000 372.000 418.000 142.000  
 334.000 1895.000 1958.000 2126.000 2063.000 2043.000 372.000 418.000 142.000 334.000  
 365.000 509.000 837.000 1058.000 1193.000 211.000 690.000 798.000 933.000 832.000  
 1297.000 2776.000 2759.000 2629.000 2263.000 2207.000 798.000 933.000 832.000 1297.000  
 764.000 791.000 875.000 994.000 1129.000 977.000 927.000 548.000 615.000 360.000  
 583.000 1939.000 1910.000 1939.000 1728.000 1704.000 548.000 615.000 360.000 583.000

584.000 647.000 826.0001023.0001158.000 754.000 768.000 424.000 521.000 273.000  
697.0002182.0002175.0002192.0001921.0001901.000 424.000 521.000 273.000 697.000  
437.000 517.000 775.000 983.0001118.000 522.000 696.000 469.000 584.000 446.000  
898.0002376.0002359.0002333.0002025.0001991.000 469.000 584.000 446.000 898.000  
2414.0002321.0002073.0002044.0002172.0002722.0002568.0002132.0002138.0001902.000  
1523.000 279.000 207.000 752.0001010.0001094.0002132.0002138.0001902.0001523.000  
399.000 375.000 451.000 631.000 766.000 804.0001048.000 796.000 895.000 658.000  
1005.0002270.0002212.0002025.0001711.0001655.000 796.000 895.000 658.0001005.000  
608.0002321.000 867.0001074.0001209.000 715.000 701.000 362.000 460.000 264.000  
726.0002244.0002254.0002271.0002000.0001980.000 362.000 460.000 264.000 726.000  
772.000 918.0001221.0001443.0001578.000 274.000 266.000 644.000 784.000 832.000  
1301.0002862.0002925.0002921.0002602.0002558.000 644.000 784.000 832.0001301.000  
1025.0001119.0001319.0001526.0001661.000 931.000 584.000 140.000 25.000 278.000  
617.0002268.0002331.0002441.0002348.0002328.000 140.000 25.000 278.000 617.000  
2639.0002546.0002298.0002269.0002397.0002947.0002807.0002371.0002377.0002141.000  
1762.000 222.000 48.000 599.000 917.0001037.0002371.0002377.0002141.0001762.000  
2400.0002373.0002173.0002178.0002306.0002644.0002415.0001979.0001985.0001749.000  
1370.000 283.000 354.000 902.0001212.0001296.0001979.0001985.0001749.0001370.000  
788.000 882.0001117.0001326.0001461.000 682.0001015.000 128.000 265.000 277.000  
746.0002307.0002370.0002410.0002195.0002175.000 128.000 265.000 277.000 746.000  
593.000 739.0001055.0001277.0001412.000 96.000 484.000 696.000 835.000 825.000  
1294.0002855.0002877.0002769.0002450.0002392.000 696.000 835.000 825.0001294.000  
458.000 381.000 333.000 501.000 636.000 917.0001182.000 919.0001018.000 780.000  
1062.0002198.0002129.0001888.0001562.0001506.000 919.0001018.000 780.0001062.000  
452.000 479.000 598.000 766.000 901.000 751.000 971.000 654.000 753.000 515.000  
884.0002199.0002168.0002091.0001783.0001759.000 654.000 753.000 515.000 884.000  
2569.0002519.0002271.0002242.0002370.0002813.0002584.0002148.0002154.0001918.000  
1539.000 143.000 289.000 841.0001159.0001279.0002148.0002154.0001918.0001539.000

C-----

C       Explanation for the Fourth part

C       (4) Each barge-unloading river location is linked to a  
C       corresponding deficit region.

C       There are 18 deficit regions, thus a 1 by 18 matrix.  
C       Diagrammatically,

C       barge-unloading location 1       -----> corresponding deficit region

C       barge-unloading location 2       -----> corresponding deficit region

C       .

C       .

C       barge-unloading location 18      -----> corresponding deficit region

C       e.g., barge-unloading river location, Chattanooga TN(627) to  
C       1st deficit region Atlanta GA (211)      = 176 miles

C       e.g., barge-unloading river location, Cairo IL(621) to  
C       3rd deficit region Cabool MO (098)      = 193 miles

C       If a new deficit region were added to the end of the deficit region list,  
C       then its truck mileage to the corresponding river barge-unloading region  
C       would be added to the end of this list.

C If a new deficit region were added to position 10, then its truck mileage  
C to the corresponding river barge-unloading region would be at position  
C 10 in this list.

C If a deficit region were deleted, then its truck mileage to the  
C river barge-unloading region would be removed from this list.  
C e.g., if Atlanta GA (211) were not a deficit region, then 176.000  
C would be deleted.

C If a barge-unloading river location were deleted, make sure that there  
C is no deficit region linked to it.

176.000 999.000 193.000 92.000 999.000 387.000 2.000 999.000 999.000 66.000  
141.000 999.000 574.000 171.000 65.000 197.000 999.000 999.000 0.000

C-----

C

F3.sft

C This is the 3rd file used in Soft Wheat and it deals with the  
C rail costs (cents/bushel).

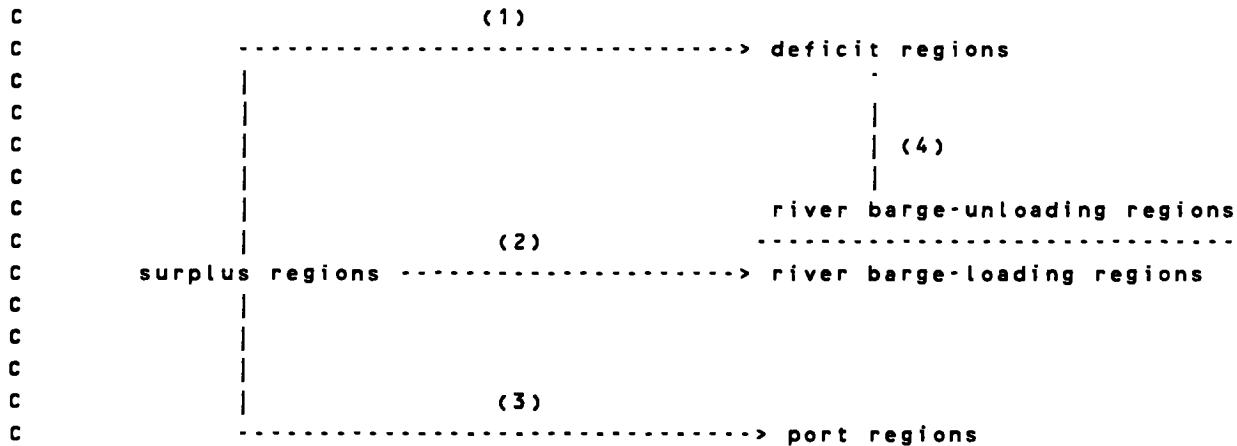
C Note: 999.999 indicates an infeasible rail cost.

C In this model, 4 sets of rail cost must be found.

C This includes

- C (1) rail cost from each surplus region to each deficit region
- C (2) rail cost from each surplus region to each river  
C barge-loading region
- C (3) rail cost from each surplus region to each port region
- C (4) rail cost from each barge-unloading river location to  
C each deficit region

C Diagrammatically,



C-----

C First part

C (1) Each surplus region is linked to each of the deficit regions.  
C In this model there are 41 surplus regions and 18 deficit regions.  
C Each surplus region is linked to all deficit regions(18). Thus  
C resulting in a 41 by 18 matrix.

C surplus region 1 -----> deficit region 1  
C -----> deficit region 2  
C .  
C -----> deficit region 18

C surplus region 2 -----> deficit region 1  
C -----> deficit region 2  
C .  
C -----> deficit region 18

C .

C .  
 C .  
 C      surplus region 41    -----> deficit region 1  
 C                            -----> deficit region 2  
 C .  
 C .  
 C                            -----> deficit region 18

C      For example,  
 C      the rail cost from Alexandria LA(111) to 1st deficit region,  
 C      Atlanta GA (211) = 48.000 cents/bushel  
 C      the rail cost from Alexandria LA(111) to 3rd deficit region,  
 C      Cabool MO (098) = 54.000 cents/bushel

48.000	999.999	54.000	999.999	999.999	999.999	57.600	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
999.999	999.999	46.200	999.999	56.400	999.999	72.600	999.999	999.999	999.999	52.200	
48.600	999.999	999.999	999.999	33.000	42.000	999.999	999.999	999.999	999.999	0.000	
27.000	51.600	39.000	25.200	999.999	53.400	22.200	54.600	999.999	999.999	999.999	
38.400	999.999	64.800	30.600	999.999	999.999	999.999	45.600	999.999	999.999	0.000	
47.400	39.600	999.999	23.400	999.999	23.400	999.999	79.200	999.999	999.999	999.999	
999.999	999.999	51.000	19.200	999.999	999.999	999.999	52.800	999.999	999.999	0.000	
999.999	999.999	80.400	999.999	29.400	999.999	999.999	999.999	999.999	106.200	76.800	
83.400	58.800	999.999	999.999	67.200	66.600	78.600	999.999	999.999	999.999	0.000	
37.200	30.000	42.600	39.600	999.999	66.000	48.000	78.000	999.999	999.999	999.999	
30.600	999.999	63.000	46.200	44.400	54.600	999.999	56.400	999.999	999.999	0.000	
45.000	999.999	999.999	999.999	999.999	39.600	34.800	78.600	999.999	999.999	999.999	
999.999	999.999	66.600	27.600	999.999	999.999	999.999	34.800	999.999	999.999	0.000	
999.999	999.999	999.999	999.999	999.999	40.800	999.999	52.800	999.999	999.999	999.999	
999.999	999.999	52.200	53.400	999.999	999.999	999.999	15.600	999.999	999.999	0.000	
999.999	999.999	35.400	999.999	999.999	999.999	999.999	999.999	999.999	999.999	37.800	
41.400	999.999	999.999	999.999	22.200	30.600	999.999	999.999	999.999	999.999	0.000	
999.999	999.999	25.800	999.999	55.800	999.999	78.600	999.999	999.999	999.999	40.200	
28.800	999.999	999.999	999.999	27.600	38.400	999.999	999.999	999.999	999.999	0.000	
38.400	34.200	999.999	27.600	999.999	50.400	33.000	70.200	999.999	999.999	999.999	
999.999	999.999	53.400	24.600	999.999	999.999	999.999	53.400	999.999	999.999	0.000	
65.400	29.400	48.600	42.000	999.999	61.800	61.800	999.999	999.999	999.999	999.999	
37.200	999.999	63.000	34.800	999.999	999.999	999.999	82.200	999.999	999.999	0.000	
48.600	999.999	27.000	52.800	999.999	999.999	43.800	999.999	999.999	999.999	999.999	
31.800	999.999	999.999	999.999	40.800	48.600	999.999	999.999	999.999	999.999	0.000	
44.400	37.800	36.000	31.800	999.999	58.800	43.800	999.999	999.999	999.999	999.999	
24.600	999.999	60.600	34.200	999.999	999.999	999.999	999.999	999.999	999.999	0.000	
69.600	27.000	999.999	26.400	999.999	43.800	37.200	64.800	999.999	999.999	999.999	
999.999	999.999	45.000	15.000	999.999	999.999	999.999	58.200	999.999	999.999	0.000	
999.999	999.999	32.400	999.999	58.200	999.999	999.999	999.999	999.999	96.600	21.600	
35.400	93.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000	
14.400	999.999	999.999	999.999	999.999	999.999	999.999	21.600	34.200	999.999	999.999	
999.999	999.999	999.999	999.999	999.999	999.999	999.999	38.400	999.999	999.999	0.000	
62.400	37.200	40.800	49.800	999.999	69.600	63.000	999.999	999.999	999.999	999.999	
28.800	999.999	71.400	42.600	999.999	999.999	999.999	79.800	999.999	999.999	0.000	
999.999	999.999	999.999	999.999	100.200	999.999	999.999	999.999	999.999	101.400	999.999	
999.999	70.200	999.999	999.999	999.999	999.999	81.600	999.999	999.999	0.000		
27.000	999.999	999.999	999.999	999.999	999.999	41.400	46.200	999.999	999.999	999.999	

45.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
30.000	999.999	999.999	999.999	999.999	999.999	999.999	34.800	43.800	999.999	999.999	
53.400	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
46.200	27.000	48.600	22.200	999.999	49.200	36.000	69.000	999.999	999.999	999.999	
36.600	999.999	51.000	24.600	999.999	999.999	999.999	999.999	61.800	999.999	999.999	0.000
26.400	999.999	29.400	999.999	999.999	999.999	31.800	46.200	999.999	999.999	999.999	
43.800	999.999	999.999	999.999	999.999	999.999	999.999	999.999	48.600	999.999	999.999	0.000
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	107.400	999.999
999.999	69.600	999.999	999.999	132.000	131.400	81.000	999.999	999.999	999.999	999.999	0.000
999.999	34.800	63.000	999.999	999.999	999.999	66.600	999.999	999.999	999.999	999.999	999.999
51.000	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
16.200	999.999	999.999	999.999	999.999	999.999	27.000	23.400	999.999	999.999	999.999	
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
999.999	999.999	36.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	48.600
31.800	999.999	999.999	999.999	25.200	36.000	999.999	999.999	999.999	999.999	999.999	0.000
37.200	45.000	37.200	41.400	999.999	64.200	33.000	61.800	999.999	999.999	999.999	
25.200	999.999	999.999	48.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
22.800	999.999	999.999	999.999	999.999	999.999	18.600	42.600	999.999	999.999	999.999	
999.999	999.999	999.999	999.999	999.999	999.999	999.999	39.000	999.999	999.999	999.999	0.000
999.999	999.999	150.600	999.999	84.600	999.999	999.999	999.999	999.999	999.999	99.000	130.800
151.800	51.000	999.999	999.999	120.000	120.600	69.600	999.999	999.999	999.999	999.999	0.000
49.800	999.999	29.400	999.999	999.999	999.999	41.400	999.999	999.999	999.999	999.999	39.000
41.400	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	0.000
62.400	33.000	44.400	46.200	999.999	65.400	58.200	999.999	999.999	999.999	999.999	999.999
32.400	999.999	67.200	41.400	999.999	999.999	999.999	999.999	78.600	999.999	999.999	0.000
36.000	999.999	999.999	999.999	999.999	999.999	50.400	36.600	40.200	999.999	999.999	999.999
999.999	999.999	66.600	999.999	999.999	999.999	999.999	999.999	19.800	999.999	999.999	0.000
59.400	19.800	999.999	20.400	999.999	50.400	51.600	78.600	999.999	999.999	999.999	
999.999	999.999	52.200	23.400	999.999	999.999	999.999	999.999	66.000	999.999	999.999	0.000
999.999	999.999	167.400	999.999	105.000	999.999	999.999	999.999	999.999	999.999	81.000	151.200
163.800	71.400	999.999	999.999	143.400	141.000	56.400	999.999	999.999	999.999	999.999	0.000
999.999	999.999	126.600	999.999	97.800	999.999	999.999	999.999	999.999	999.999	96.000	139.800
129.600	68.400	999.999	999.999	132.000	133.800	84.000	999.999	999.999	999.999	999.999	0.000
48.000	36.600	66.000	12.600	999.999	27.000	33.600	57.000	999.999	999.999	999.999	
43.800	999.999	52.200	17.400	67.200	79.200	999.999	52.200	999.999	999.999	999.999	0.000
23.400	999.999	999.999	999.999	999.999	999.999	999.999	999.999	30.600	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	22.800	999.999	999.999	0.000
999.999	999.999	40.800	999.999	77.400	999.999	57.000	999.999	999.999	999.999	40.800	
51.000	999.999	999.999	999.999	48.000	46.800	999.999	999.999	999.999	999.999	999.999	0.000
40.800	999.999	16.800	999.999	999.999	999.999	46.800	67.200	999.999	999.999	999.999	42.600
27.600	999.999	999.999	999.999	41.400	37.200	999.999	999.999	999.999	999.999	999.999	0.000
999.999	999.999	999.999	999.999	108.600	999.999	999.999	999.999	999.999	999.999	106.800	148.800
140.400	78.600	999.999	999.999	141.000	142.800	94.200	999.999	999.999	999.999	999.999	0.000

C-----

C       Second part

C       (2) Each surplus region is linked to 43 river barge-loading locations.

C       surplus region 1       -----> river barge-loading region 1  
 C                              -----> river barge-loading region 2









999.999	18.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999																0.000
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	19.200	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999																0.000
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999

C-----

C        Third part

C        (3) Each surplus region is linked to all the port locations for export.  
C        There are 20 ports in this model.

C        surplus region 1    -----> port region 1 for export  
C                            -----> port region 2 for export

C

C                            . . . . . -----> port region 20 for export

C        surplus region 2    -----> port region 1 for export  
C                            -----> port region 2 for export

C

C                            . . . . . -----> port region 20 for export

C

C

C        surplus region 41    -----> port region 1 for export  
C                            -----> port region 2 for export

C

C                            . . . . . -----> port region 20 for export

C

C        The rail costs from each surplus region to the 20 ports. Thus, a  
C        41 by 20 matrix.

C        For example,

C        from the 1st surplus region, Alexandria LA(111) to 2nd port region,  
C        New Orleans LA(702) = 21.000 cents/bushel

999.999	21.000	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
82.800	79.800	90.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999



999.999	32.400	21.600	57.600	84.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
45.000	36.600	43.200	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
72.000	68.400	999.999	999.999	999.999	999.999	999.999	61.200	35.400	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	35.400	999.999	999.999	999.999	999.999	999.999	999.999
55.800	999.999	999.999	999.999	999.999	21.000	35.400	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	61.800	16.800	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	16.800	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	24.000	13.200	46.800	73.800	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	27.000	25.800	70.800	96.600	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
61.200	105.600	106.800	999.999	999.999	999.999	53.400	30.000	999.999	46.800			
999.999	999.999	999.999	999.999	999.999	999.999	30.000	999.999	46.800	999.999			
43.200	999.999	999.999	999.999	999.999	10.800	41.400	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	37.200	31.800	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	37.200	45.000	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999
999.999	19.200	30.000	64.200	91.200	999.999	999.999	999.999	999.999	999.999	999.999	999.999	999.999

C-----

C       Fourth part

C       (4) Each barge-unloading river location is linked to 18  
C       deficit regions.

C       barge-unloading river location 1       -----> deficit region 1  
C    -----> deficit region 2  
C    .  
C    .  
C    -----> deficit region 18

C       barge-unloading river location 2       -----> deficit region 1  
C    -----> deficit region 2  
C    .  
C    .  
C    -----> deficit region 18

C       .  
C       .  
C       .  
C       barge-unloading river location 43    -----> deficit region 1  
C    -----> deficit region 2  
C    .  
C    .  
C    -----> deficit region 18





C

F4.sft

C This is the 4th data file used in Soft Wheat deals with the  
 C barge costs and shiprates.

C The barge costs from 43 river regions to 5 river shipping locations.  
 C The 5 river shipping locations are :

C       Nashville, TN (625)  
 C       Knoxville, TN (626)  
 C       Chatanoo, TN (627)  
 C       Guntersv, AL (628)  
 C       Florence, AL (629)

C       For example,

C       the barge cost from the 1st river barge region,

C           St. Paul, MN (601) to Nashville (702) = 0.302 \$/bushel  
 C           St. Paul, MN (601) to Knoxville (701) = 0.327 \$/bushel  
 C           St. Paul, MN (601) to Chatanoo (703) = 0.299 \$/bushel  
 C           St. Paul, MN (601) to Guntersv (710) = 0.284 \$/bushel  
 C           St. Paul, MN (601) to Florence (713) = 0.269 \$/bushel

0.302	0.327	0.299	0.284	0.269	0
0.266	0.290	0.263	0.248	0.233	0
0.248	0.273	0.246	0.230	0.215	0
0.238	0.263	0.236	0.221	0.205	0
0.230	0.254	0.227	0.211	0.197	0
0.201	0.226	0.199	0.183	0.168	0
0.184	0.209	0.181	0.166	0.151	0
0.145	0.170	0.134	0.127	0.112	0
0.353	0.378	0.350	0.335	0.320	0
0.323	0.348	0.321	0.305	0.290	0
0.424	0.458	0.421	0.400	0.337	0
0.381	0.415	0.378	0.357	0.337	0
0.356	0.389	0.352	0.331	0.311	0
0.308	0.341	0.305	0.283	0.263	0
0.290	0.324	0.287	0.266	0.245	0
0.274	0.308	0.271	0.249	0.229	0
0.256	0.290	0.253	0.232	0.211	0
0.217	0.206	0.174	0.155	0.137	0
0.198	0.191	0.158	0.139	0.122	0
0.151	0.153	0.120	0.102	0.084	0
0.133	0.136	0.104	0.085	0.067	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0
0.273	0.247	0.215	0.196	0.178	0
0.264	0.241	0.208	0.190	0.172	0
0.213	0.199	0.167	0.148	0.130	0
999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	0

999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0
999.000	999.000	999.000	999.000	999.000	999.000	0

C-----

C        The barge costs from 43 river port shipping points to 5 port locations.  
C        The 5 river port shipping points are:

C              New Orle, LA (702)  
C              Mobile, AL (701)  
C              Galvestn, TX (703)  
C              Chicago, IL (710)  
C              Portland, OR (713)

C        For example, the 1st river region (St.Paul)

C              St. Paul MN(601) to New Orle, LA (702) = 0.283 \$/bushel  
C              St. Paul MN(601) to Mobile, AL (701) = 0.345 \$/bushel  
C              St. Paul MN(601) to Galvestn, TX (703) = 0.398 \$/bushel  
C              St. Paul MN(601) to Chicago, IL (710) = 0.234 \$/bushel  
C              St. Paul MN(601) to Portland, OR (713) = 999    \$/bushel

0.283	0.345	0.398	0.234	999.000	0
0.247	0.308	0.362	0.197	999.000	0
0.229	0.291	0.344	0.180	999.000	0
0.219	0.281	0.334	0.170	999.000	0
0.210	0.272	0.325	0.161	999.000	0
0.182	0.244	0.297	0.133	999.000	0
0.164	0.226	0.280	0.116	999.000	0
0.125	0.187	0.240	0.109	999.000	0
0.334	0.273	0.449	0.300	999.000	0
0.304	0.366	0.419	0.270	999.000	0
0.398	0.429	0.555	0.352	999.000	0
0.355	0.398	0.512	0.309	999.000	0
0.329	0.379	0.486	0.283	999.000	0
0.281	0.344	0.438	0.238	999.000	0
0.264	0.331	0.421	0.060	999.000	0
0.248	0.319	0.404	0.079	999.000	0
0.230	0.306	0.387	0.099	999.000	0
0.158	0.309	0.430	999.000	999.000	0
0.147	0.281	0.398	999.000	999.000	0
0.120	0.214	0.319	999.000	999.000	0
0.074	0.200	0.267	999.000	999.000	0
0.069	0.187	0.253	999.000	999.000	0
0.053	0.149	0.207	999.000	999.000	0
0.057	0.157	0.216	999.000	999.000	0
0.150	0.281	0.393	999.000	999.000	0
0.123	0.216	0.327	999.000	999.000	0
0.104	0.170	0.273	999.000	999.000	0
0.094	0.144	0.242	999.000	999.000	0

0.083	0.118	0.213	999.000	999.000		0
0.098	0.229	0.287	999.000	999.000		0
0.093	0.217	0.273	999.000	999.000		0
0.063	0.144	0.188	999.000	999.000		0
999.000	999.000	999.000	999.000	999.000		0
0.042	0.122	0.176	999.000	999.000		0
0.034	0.102	0.153	999.000	999.000		0
999.000	999.000	999.000	999.000	999.000		0
999.000	999.000	999.000	999.000	0.170		0
999.000	999.000	999.000	999.000	0.145		0
999.000	999.000	999.000	999.000	0.113		0
999.000	999.000	999.000	999.000	0.063		0
999.000	999.000	999.000	999.000	0.052		0
0.000	999.000	999.000	999.000	999.000		0
999.000	999.000	999.000	999.000	0.000		0

C-----

C       The number of river shipping ports above L&D 26

13		0
----	--	---

C-----

C       The codes for the river shipping points.

601	602	603	604	605	606	607	609	610	611	612	613	614	0
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	---

C-----

C       Shiprates from 20 ports to 25 world demand locations

C       For example, the shiprate from the 1st port

C              Mobile, AL (701) to Scandinavia (801) = 0.475 \$/bushel

C              Mobile, AL (701) to N. C. Europe(802) = 0.374 \$/bushel

0.475	0.374	0.320	0.344	0.453	0.538	0.480	0.495	0.477	0.734	
1.024	0.418	0.511	0.519	0.828	0.461	0.632	0.434	0.707	999.999	
0.285	0.082	0.306	0.161	0.215					0.000	
0.475	0.374	0.320	0.344	0.453	0.538	0.480	0.495	0.477	0.734	
1.024	0.418	0.511	0.519	0.828	0.461	0.632	0.434	0.707	999.999	
0.285	0.082	0.306	0.161	0.215					0.000	
0.490	0.387	0.332	0.355	0.465	0.553	0.499	0.508	0.494	0.751	
1.045	0.435	0.520	0.525	0.840	0.461	0.621	0.417	0.696	999.999	
0.277	0.059	0.272	0.177	0.229					0.000	
0.490	0.387	0.332	0.355	0.465	0.553	0.499	0.508	0.494	0.750	
1.045	0.435	0.520	0.525	0.840	0.461	0.621	0.417	0.696	999.999	
0.277	0.059	0.272	0.177	0.229					0.000	
0.499	0.394	0.341	0.363	0.472	0.562	0.507	0.514	0.504	0.761	
1.056	0.445	0.525	0.525	0.856	0.358	0.855	0.417	0.967	999.999	
0.313	0.048	0.273	0.169	0.238					0.000	
0.434	0.374	0.332	0.348	0.505	0.462	0.535	0.552	0.399	0.645	
0.970	0.352	0.842	0.799	1.153	1.002	0.931	0.883	0.955	999.999	
0.339	0.166	0.301	0.148	0.148					0.000	
0.375	0.326	0.287	0.307	0.449	0.437	0.482	0.490	0.367	0.605	
0.973	0.330	0.577	0.547	0.902	0.947	0.887	0.838	0.982	999.999	

0.347	0.202	0.312	0.171	0.169						0.000
0.740	0.740	0.624	0.636	0.934	0.839	0.999	0.974	0.711	1.157	
1.545	0.705	1.527	1.722	2.236	2.184	2.094	1.931	2.125	999.999	
0.932	0.660	0.872	0.637	0.541						0.000
0.794	0.794	0.677	0.690	0.988	0.892	1.052	1.027	0.765	1.190	
1.599	0.759	1.580	1.775	2.290	2.237	2.147	1.984	2.178	999.999	
0.986	0.714	0.926	0.691	0.595						0.000
0.853	0.762	0.736	0.749	1.047	0.951	1.111	1.091	0.824	1.250	
1.659	0.819	1.640	1.834	2.350	2.296	2.101	2.043	2.238	999.999	
0.995	0.775	0.978	0.750	0.654						0.000
0.867	0.716	0.690	0.763	1.061	0.965	1.125	1.105	0.838	1.264	
1.671	0.831	1.654	1.848	2.362	2.310	2.218	2.057	2.192	999.999	
1.059	0.787	0.933	0.704	0.550						0.000
0.832	0.753	0.775	0.760	0.881	0.901	0.689	0.652	0.835	1.211	
1.483	0.746	0.456	0.536	0.375	0.291	0.431	0.228	0.499	999.999	
0.317	0.570	0.257	0.447	0.630						0.000
0.820	0.769	0.761	0.747	0.868	0.888	0.694	0.657	0.821	1.197	
1.413	0.727	0.461	0.541	0.380	0.296	0.430	0.233	0.252	999.999	
0.312	0.554	0.449	0.434	0.612						0.000
0.764	0.716	0.707	0.692	0.815	0.831	0.882	0.877	0.761	1.138	
1.340	0.774	0.741	0.697	0.647	0.513	0.461	0.403	0.554	999.999	
0.246	0.492	0.382	0.376	0.549						0.000
0.735	0.687	0.678	0.663	0.785	0.801	0.613	0.725	0.682	1.107	
1.517	0.822	0.489	0.576	0.416	0.289	0.427	0.225	0.492	999.999	
0.212	0.457	0.200	0.344	0.496						0.000
0.794	0.746	0.736	0.720	0.844	0.853	0.613	0.914	0.737	1.257	
1.483	0.805	1.049	0.982	0.715	0.514	0.457	0.395	0.485	999.999	
0.216	0.451	0.361	0.334	0.487						0.000
0.816	0.665	0.708	0.714	0.840	0.874	0.770	0.765	0.817	1.055	
1.468	0.818	0.846	1.173	1.148	1.132	1.503	1.055	1.612	999.999	
0.940	0.812	0.904	0.772	0.801						0.000
0.859	0.708	0.741	0.757	0.883	0.917	0.813	0.817	0.860	1.098	
1.510	0.861	0.889	1.216	1.191	1.175	1.546	1.098	1.655	999.999	
0.983	0.855	0.947	0.815	0.844						0.000
0.963	0.811	0.854	0.860	0.987	1.021	0.918	0.912	0.964	1.202	
1.615	0.965	0.993	1.320	1.295	1.279	1.650	1.202	1.759	999.999	
1.087	0.959	1.051	0.919	0.948						0.000
0.871	0.733	0.763	0.647	0.895	0.925	0.826	0.820	0.872	1.110	
1.522	0.873	0.901	1.228	1.203	1.187	1.558	1.110	1.666	999.999	
0.995	0.867	0.959	0.827	0.856						0.000

C-----  
C       Number of ports at Great Lakes for export

8

0

C-----  
C       Codes for the above ports

- C       (708) Toledo, OH
- C       (709) Saginaw, MI
- C       (710) Chicago, IL
- C       (711) Duluth, MN
- C       (717) Toledo, BC
- C       (718) Saginaw, BC
- C       (719) Chicago, BC
- C       (720) Duluth, BC

708 709 710 711 717 718 719 720

0

C-----

C

F8.sft

C This is the 5th data file in Soft wheat deals with  
 C supplies and demands.

C-----  
 C These lines show the amount of surplus produced by each surplus region.  
 C For example, the 1st surplus region,  
 C Alexandria LA (111) produces 4.937 million/bushel.

4.937	0.308	12.691	4.028	0.014	2.086	0.513	2.838	3.455	5.854
3.330	1.089	13.576	7.745	10.402	4.445	2.338	9.458	32.769	10.648
5.035	8.277	10.099	53.656	3.145	6.015	4.804	13.898	2.973	32.398
20.437	2.127	9.579	0.783	14.799	51.760	11.270	7.433	1.050	12.240
									0.000
5.603									

C-----  
 C These lines show the amount of corn demanded by the 56 deficit regions.  
 C For example, the 1st deficit region,  
 C Atlanta GA (211) requires 7.113 million/bushel.

7.113	1.266	0.763	0.210	1.406	10.767	22.048	0.759	4.181	0.264
2.528	7.393	12.279	2.041	1.926	3.144	0.593	5.026		0.000

C-----  
 C The remaining lines show the amount demanded by 25 foreign regions  
 C (demand per quarter).  
 C For example, the 1st foreign region, Scandinavia  
 C in the 1st quarter requires 0.000 million bushel  
 C in the 2nd quarter requires 0.000 million bushel  
 C in the 3rd quarter requires 0.000 million bushel  
 C in the 4th quarter requires 0.057 million bushel

0.000	0.000	0.000	0.057	0.000
0.004	0.007	0.015	0.012	0.000
0.292	1.437	2.216	3.246	0.000
0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000
0.000	0.115	0.480	0.119	0.000
7.516	9.146	4.334	10.776	0.000
18.060	16.210	18.561	18.387	0.000
0.056	0.050	0.043	0.047	0.000
0.258	1.465	2.958	0.743	0.000
0.006	0.037	0.315	0.312	0.000
0.000	0.000	0.000	0.000	0.000
3.010	5.990	3.952	2.184	0.000
6.598	6.598	6.598	6.598	0.000
2.338	2.338	2.338	2.338	0.000
1.099	1.236	1.292	1.300	0.000
13.234	16.842	19.282	15.859	0.000
9.333	7.764	8.030	8.333	0.000
0.000	5.948	13.633	20.912	0.000
0.000	0.000	0.000	0.000	0.000

0.019	0.140	0.013	0.017	0.000
0.019	0.140	0.013	0.017	0.000
0.042	0.076	0.308	0.264	0.000
1.050	0.831	1.559	1.095	0.000
1.378	0.967	1.332	0.892	0.000
C-----				

C

F9.sft

C The last and 6th data file with the location names and code numbers  
C of all regions in the Soft Wheat model.

C-----

C 41 Surplus regions

C Code number Location name State

111	Alexandria	LA	0
085	Ames	IA	0
171	Bowling Green	KY	0
154	Cambridge	OH	0
351	Casper	WY	0
133	Champaign	IL	0
261	Charleston	WV	0
251	Charlotvi	VA	0
092	Chillicothe	MO	0
094	Clinton	MO	0
144	Columbus	IN	0
131	De Kalb	IL	0
099	Dexter	MO	0
134	Effingham	IL	0
141	Fort Wayne	IN	0
065	Ft Worth	TX	0
201	Gadsden	AL	0
132	Galesburg	IL	0
312	Moses Lake	WA	0
191	Greenwood	MS	0
192	Hattiesburg	MS	0
142	Indy	IN	0
181	Jackson	TN	0
638	Lewiston	ID	0
123	Madison	WI	0
212	McRae	GA	0
093	Moberly	MO	0
135	Mt Vernon	IL	0
182	Nashville	TN	0
331	Pendleton	OR	0
104	Pine Bluff	AR	0
143	Princeton	IN	0
242	Rocky Mtn	NC	0
709	Saginaw	MI	0
713	Salem	OR	0
311	Spokane	WA	0
152	Springfld	OH	0
231	Sumter	SC	0
103	Texarkana	AR	0
102	Walnut Ri	AR	0
313	Wenatchee	WA	0

C-----

C 18 Deficit regions

C	Code number	Location name	State	
	211	Atlanta	GA	0
	163	Battle Cr	MI	0
	098	Cabool	MO	0
	153	Chillico	OH	0
	391	Denver	CO	0
	281	Du Bois	PA	0
	183	Knoxville	TN	0
	222	Lake City	FL	0
	715	Los Angel	CA	0
	096	OklahomaC	OK	0
	053	Rolla	MO	0
	381	Salt Lake	UT	0
	291	Syracuse	NY	0
	151	Tiffin	OH	0
	043	Topeka	KS	0
	045	Wichita	KS	0
	371	Winnemucca	NV	0
	241	Winston/S	NC	0

C-----

C      43 River regions

C	Code number	Location name	State	
	601	ST. PAUL,	MN	0
	602	WINONA,	MN	0
	603	MCGREGOR,	IA	0
	604	DUBUQUE,	IA	0
	605	CLINTON,	IA	0
	606	BURLINGT,	IA	0
	607	HANNIBAL,	MO	0
	608	ST LOUIS,	MO	0
	609	SIOUX CT,	IA	0
	610	OMAHA,	NE	0
	611	NE CITY,	NE	0
	612	ST JOSEP,	MO	0
	613	KANSAS C,	MO	0
	614	GLASCO,	MO	0
	615	OTTAWA,	IL	0
	616	PEORIA,	IL	0
	617	BEARDSTO,	IL	0
	618	CINCINNA,	OH	0
	619	LOUISVIL,	KY	0
	620	EVANSVIL,	IN	0
	621	CAIRO,	IL	0
	622	HICKMAN,	KY	0
	623	OSCEOLA,	TN	0
	624	MEMPHIS,	TN	0
	625	NASHVILL,	TN	0
	626	KNOXVILL,	TN	0
	627	CHATTANOO,	TN	0

628	GUNTERSV,	AL	0
629	FLORENCE,	AL	0
630	CATOOSA,	OK	0
631	MUSKOGEE,	OK	0
632	PINEBLUF,	AR	0
633	DES ARC,	AR	0
634	GREENVIL,	MS	0
635	VICKSBUR,	MS	0
636	MONROE,	LA	0
637	LEWISTON,	ID	0
638	CENTRL F,	WA	0
639	PASCO,	WA	0
640	ROOSEVEL,	WA	0
641	THE DALL,	OR	0
642	NEW ORLE,	LA	0
643	PORTLAND,	OR	0

C-----

C      20 Port regions

Code number	Location name	State	
701	MOBILE,	AL	0
702	NEW ORLE,	LA	0
703	GALVESTN,	TX	0
704	CORPUS C,	TX	0
705	BROWNSVI,	TX	0
706	CHARLEST,	SC	0
707	BALTIMOR,	MD	0
708	TOLEDO,	OH	0
709	SAGINAW,	MI	0
710	CHICAGO,	IL	0
711	DULUTH,	MN	0
712	SEATTLE,	WA	0
713	PORTLAND,	OR	0
714	SAN FRAN,	CA	0
715	LONG BEA,	CA	0
716	SAN DIEG,	CA	0
717	TOLEDO,	BC	0
718	SAGINAW,	BC	0
719	CHICAGO,	BC	0
720	DULUTH,	BC	0

C-----

C      25 Foreign regions

Code number	Location name	
801	SCANDINAVIA	0
802	N.C. EUROPE	0
803	S.W. EUROPE	0
804	ISLANDS	0
805	ADRIATIC	0

806	USSR	0
807	E BLOCK EURO	0
808	E MEDITERRAN	0
809	N AFRICA	0
810	RED SEA	0
811	E AFRICA	0
812	W AFRICA	0
813	PERSIAN GULF	0
814	W ASIA	0
815	SE ASIA	0
816	TAIWAN	0
817	KOREA	0
818	JAPAN	0
819	CHINA	0
820	CANADA	0
821	W MEXICO	0
822	E MEXICO	0
823	WS AMERICA	0
824	CENT AMERICA	0
825	CARIBBEAN	0

\*\*\*\*\* OPTIMAL SOLUTION \*\*\*\*\*

TOTAL COST 350577734.

SOFT WHEAT SHIPMENT BATE COMEAU ADDED, FULLER, GRANT, TEH & FE

UNIT = 1 (THOUSAND BUSHEL)

SUPPLY 40994.

DEMAND 406107.

632 633 633 11712

## 1 Network Generator For Grain Shipment Problem

SOFT WHEAT SHIPMENT BAIE COMEAU ADDED, FULLER, GRANT, TEH &amp; FE

1

ORIGIN/DESTN	MODE	SUPPLY	SHIPMENT BY TIME				TOTAL SHIPMENT	UNIT COST	TOTAL HAULING COST	HANDLING COSTS
			1	2	3	4				
S 111 Alexander LA		4937.								
P 702 NEW ORLE, LA R		3234.	1703.	00.	00.		4937.	0.21000	1036770.	713397.
S 085 Ames IA		308.								
D 391 Denver CO R		00.	00.	00.	308.		308.	0.56400	173712.	52797.
S 171 Bowling G KY		12691.								
D 183 Knoxville TN R		2539.	5512.	4640.	00.		12691.	0.22200	2817402.	2175491.
S 154 Cambridge OH		4028.								
P 707 BALTIMOR, MD R		296.	501.	1608.	1623.		4028.	0.29600	1192288.	582046.
S 351 Casper WY		14.								
D 381 Salt Lake UT T		00.	00.	14.	00.		14.	0.39398	5516.	2652.
S 133 Champaign IL		2086.								
D 211 Atlanta GA R		131.	00.	860.	1095.		2086.	0.37200	775992.	357582.
S 261 Charlesto WV		513.								
D 261 Winston/S NC T		00.	00.	00.	513.		513.	0.23457	120336.	97172.
S 251 Charlottvi VA		2838.								
P 707 BALTIMOR, MD T		00.	943.	623.	1272.		2838.	0.15695	445432.	487710.
S 092 Chilcott MD		3455.								
D 096 OklahomaC OK R		66.	66.	66.	66.		264.	0.37800	99792.	45255.
D 043 Topeka KS T		481.	481.	481.	481.		1924.	0.15905	306013.	364444.
D 045 Wichita KS T		630.	190.	00.	447.		1267.	0.27651	350344.	239995.
S 094 Clinton MO		5854.								
D 098 Cabool MO T		163.	190.	00.	00.		353.	0.17583	62069.	66865.
D 391 Denver CO R		351.	351.	351.	43.		1096.	0.55800	611568.	187876.
D 053 Rolla MO T		632.	632.	632.	632.		2528.	0.14332	362305.	478854.
D 045 Wichita KS T		156.	596.	786.	339.		1877.	0.23670	444286.	355541.

S 144 Columbus IN	3330.								
D 183 Knoxville TN T		00.	00.	00.	860.	860.	0.29061	249926.	162901.
R 619 LOUISVIL, KY T		2470.	00.	00.	00.	2470.	0.07094	175225.	416738.
S 131 Dekalb IL	1089.								
S 099 Dexter MO	13576.								
R 621 CAIRO, IL T		00.	00.	7736.	5840.	13576.	0.08143	1105502.	2290543.
S 134 Effingham IL	7745.								
R 608 ST LOUIS, MO T		1646.	4181.	1490.	428.	7745.	0.10241	793158.	1306736.
S 141 Fort Wayne IN	10402.								
D 163 Battle Cr MI T		316.	316.	316.	316.	1264.	0.10241	129445.	239427.
D 291 Syracuse NY R		765.	1964.	2408.	2259.	7396.	0.45000	3328200.	1267822.
D 151 Tiffin OH T		212.	510.	510.	510.	1742.	0.11604	202150.	329970.
S 065 Ft Worth TX	4445.								
D 715 Los Angel CA R		1045.	1045.	1045.	1045.	4180.	0.96600	4037880.	716536.
P 703 GALVESTN, TX R		1.	118.	146.	00.	265.	0.29400	77910.	38293.
S 201 Gadsen AL	2338.								
D 211 Atlanta GA T		1097.	00.	872.	369.	2338.	0.11080	259051.	442864.
S 132 Galesburg IL	9458.								
R 606 BURLINGT, IA T		00.	404.	6616.	2438.	9458.	0.05206	492391.	1595754.
S 312 Moses Lake WA	32769.								
R 640 ROOSEVEL, WA T		14333.	11168.	3630.	3638.	32769.	0.01115	365471.	5528786.
S 191 Greenwood MS	10648.								
P 702 NEW ORLE, LA R		230.	4786.	4334.	1298.	10648.	0.22800	2427744.	1538636.
S 192 Hattisbur MS	5035.								
P 701 MOBILE, AL R		5035.	00.	00.	00.	5035.	0.12000	604200.	727558.
S 142 Indy IN	8277.								
D 183 Knoxville TN T		00.	00.	00.	3397.	3397.	0.33196	1127670.	643460.
D 291 Syracuse NY R		2304.	1105.	661.	810.	4880.	0.51000	2488800.	836530.
S 181 Jackson TN	10099.								

D 211 Atlanta GA R		00.	1096.	46.	314.	1456.	0.26400	384384.	249588.
R 624 MEMPHIS, TN T		00.	1965.	4026.	2652.	8643.	0.08877	767265.	1458247.
S 638 Lewiston ID		53656.							
D 371 Winona NV T		148.	148.	148.	148.	592.	0.49078	290542.	112137.
P 713 PORTLAND, OR R		4601.	13712.	16020.	18731.	53064.	0.31200	16555968.	7667748.
S 123 Madison WI		3145.							
R 604 DUBUQUE, IA T		00.	437.	00.	00.	437.	0.10241	44753.	73731.
S 212 McRae GA		6015.							
D 211 Atlanta GA R		550.	682.	00.	00.	1232.	0.16200	199584.	211189.
D 222 Lake City FL T		189.	189.	189.	189.	756.	0.14751	111519.	143202.
P 706 CHARLEST, SC R		131.	00.	1909.	1987.	4027.	0.19200	773184.	581902.
S 093 Moberly MO		4804.							
R 607 HANNIBAL, MO T		00.	467.	1429.	2908.	4804.	0.06989	335763.	810531.
S 135 Mt Vernon IL		13898.							
R 621 CAIRO, IL T		00.	1817.	3054.	9027.	13898.	0.10241	1423280.	2344871.
S 182 Nashville TN		2973.							
D 183 Knoxville TN R		2973.	00.	00.	00.	2973.	0.18600	552978.	509632.
S 331 Pendleton OR		32398.							
D 381 Salt Lake UT R		1848.	1848.	1834.	1848.	7378.	0.51000	3762780.	1266737.
P 713 PORTLAND, OR R		7190.	6598.	6612.	4619.	25019.	0.21600	5404104.	3615246.
S 104 Pine Bluff AR		20437.							
R 632 PINEBLUF, AR T		5239.	5950.	00.	9248.	20437.	0.00276	56438.	3448131.
S 143 Princeton IN		2127.							
D 183 Knoxville TN T		00.	00.	872.	1255.	2127.	0.32820	698085.	402896.
S 242 Rocky Mtn NC		9579.							
D 241 WinstonS NC T		1256.	1256.	1256.	763.	4511.	0.15695	708014.	854474.
P 706 CHARLEST, SC R		573.	2498.	1358.	639.	5068.	0.21000	1064280.	732526.
S 709 Saginaw MI		783.							
P 718 SAGINAW, BC T		00.	00.	00.	783.	783.	0.02689	21052.	134559.
S 713 Salem OR		14799.							
P 713 PORTLAND, OR T		5348.	6472.	00.	2979.	14799.	0.05101	754925.	2543208.

S 311 Spokane WA	51760.
P 713 PORTLAND, OR R	4140. 8299. 19282. 20059. 51760. 0.25800 13354079. 7479520.
S 152 Springfield OH	11270.
D 153 Chillico OH T	52. 52. 52. 52. 208. 0.07304 15192. 39999.
D 281 Dubois PA R	2691. 2691. 2691. 2691. 10764. 0.27000 2906280. 1845165.
D 151 Tiffin OH R	298. 00. 00. 00. 298. 0.17400 51852. 51083.
S 231 Sumpter SC	7433.
P 706 CHARLEST, SC R	2044. 967. 3420. 1002. 7433. 0.10800 802764. 1074069.
S 103 Texarkana AR	1050.
P 703 GALVESTN, TX R	37. 280. 26. 707. 1050. 0.31800 333900. 151725.
S 102 Walnut Ri AR	12240.
D 098 Cabool MO T	27. 00. 190. 190. 407. 0.12024 48938. 77094.
P 702 NEW ORLE, LA R	7722. 3645. 00. 466. 11833. 0.37200 4401876. 1709869.
S 313 Wenatchee WA	5603.
P 712 SEATTLE, WA T	42. 425. 3952. 1184. 5603. 0.15066 844144. 962876.

1

SURPLUS REGION	SUPPLY	STORAGE	STORAGE COST
S 111 Alexander LA	4937.	1703. 00. 00. 65297.	
S 085 Ames IA	308.	308. 308. 308. 35623.	
S 171 Bowling G KY	12691.	10152. 4640. 00. 567156.	
S 154 Cambridge OH	4028.	3732. 3231. 1623. 330234.	
S 351 Casper WY	14.	14. 14. 00. 1074.	
S 133 Champaign IL	2086.	1955. 1955. 1095. 192596.	
S 261 Charlesto WV	513.	513. 513. 513. 59334.	
S 251 Charlottvi VA	2838.	2838. 1895. 1272. 231050.	
S 092 Chilcott MO	3455.	2278. 1541. 994. 185170.	
S 094 Clinton MO	5854.	4552. 2783. 1014. 320761.	
S 144 Columbus IN	3330.	860. 860. 860. 99468.	
S 131 Dekalb IL	1089.	00. 00. 00. 00.	
S 099 Dexter MD	13576.	13576. 5840. 1268683.	
S 134 Effingham IL	7745.	6099. 1918. 428. 324070.	
S 141 Fort Wayne IN	10402.	9109. 6319. 3085. 711782.	
S 065 Ft Worth TX	4445.	3399. 2236. 1045. 256787.	
S 201 Gadsden AL	2338.	1241. 1241. 369. 109547.	
S 132 Galesburg IL	9458.	9458. 9054. 2438. 804812.	
S 312 Moses Lake WA	32769.	18436. 7268. 3638. 1127339.	
S 191 Greenwood MS	10648.	10418. 5632. 1298. 665981.	
S 192 Hattisbur MS	5035.	00. 00. 00. 00.	
S 142 Indy IN	8277.	5973. 4868. 4207. 579638.	
S 181 Jackson TN	10099.	10099. 7038. 2966. 772671.	
S 638 Lewiston ID	53656.	48907. 35047. 18879. 3954796.	

S 123 Madison WI	3145.	437.	00.	00.	16755.
S 212 McRae GA	6015.	5145.	4274.	2176.	445956.
S 093 Moberly MO	4804.	4804.	4337.	2908.	463827.
S 135 Mt Vernon IL	13898.	13898.	12081.	9027.	1347924.
S 182 Nashville TN	2973.	00.	00.	00.	00.
S 331 Pendleton OR	32398.	23359.	14913.	6467.	1719485.
S 104 Pine Bluff AR	20437.	15198.	9248.	9248.	1297759.
S 143 Princeton IN	2127.	2127.	2127.	1255.	212022.
S 242 Rocky Mtn NC	9579.	7750.	3996.	1382.	504231.
S 709 Saginaw MI	783.	783.	783.	783.	90562.
S 713 Salem OR	14799.	9451.	2979.	2979.	592701.
S 311 Spokane WA	51760.	47620.	39321.	20039.	4114536.
S 152 Springfield OH	11270.	8229.	5486.	2743.	632772.
S 231 Sumpter SC	7433.	5389.	4422.	1002.	415228.
S 103 Texarkana AR	1050.	1013.	733.	707.	94501.
S 102 Walnut Ri AR	12240.	4491.	846.	656.	230200.
S 313 Wenatchee WA	5603.	5561.	5136.	1184.	456293.

1

ORIGIN/DESTN	MODE	SUPPLY	SHIPMENT BY TIME				TOTAL SHIPMENT	UNIT COST	TOTAL HAULING COST	HANDLING COSTS
			1	2	3	4				
R 601 ST. PAUL, MN			00.	00.	00.	00.				
R 602 WINONA, MN			00.	00.	00.	00.				
R 603 MCGREGOR, IA			00.	00.	00.	00.				
R 604 DUBUQUE, IA			00.	437.	00.	00.				
P 702 NEW ORLE, LA B			00.	437.	00.	00.	437.	0.21900	95703.	69286.
R 605 CLINTON, IA			00.	00.	00.	00.				
R 606 BURLINGT, IA			00.	404.	6616.	2438.				
P 702 NEW ORLE, LA B			00.	404.	6616.	2438.	9458.	0.18200	1721356.	1499566.
R 607 HANNIBAL, MO			00.	467.	1429.	2908.				
P 702 NEW ORLE, LA B			00.	467.	1429.	2908.	4804.	0.16400	787856.	761674.
R 608 ST LOUIS, MO		1646.	4181.	1490.	428.					
P 702 NEW ORLE, LA B		1646.	4181.	1490.	428.	7745.	0.12500	968125.	1227970.	
R 609 SIOUX CT, IA			00.	00.	00.	00.				

R 610 OMAHA, NE	00.	00.	00.	00.				
R 611 NE CITY, NE	00.	00.	00.	00.				
R 612 ST JOSEP, MO	00.	00.	00.	00.				
R 613 KANSAS C, MO	00.	00.	00.	00.				
R 614 GLASCO, MO	00.	00.	00.	00.				
R 615 OTTAWA, IL	00.	00.	00.	00.				
R 616 PEORIA, IL	00.	00.	00.	00.				
R 617 BEARDSTO, IL	00.	00.	00.	00.				
R 618 CINCINNA, OH	00.	00.	00.	00.				
R 619 LOUISVIL, KY	2470.	00.	00.	00.				
P 702 NEW ORLE, LA B	2470.	00.	00.	00.	2470.	0.14700	363090.	391619.
R 620 EVANSVIL, IN	00.	00.	00.	00.				
R 621 CAIRO, IL	00.	1817.	10790.	14867.				
P 702 NEW ORLE, LA B	00.	1817.	10790.	14867.	27474.	0.07400	2033076.	4356003.
R 622 HICKMAN, KY	00.	00.	00.	00.				
R 623 OSCEOLA, AR	00.	00.	00.	00.				
R 624 MEMPHIS, TN	00.	1965.	4026.	2652.				
P 702 NEW ORLE, LA B	00.	1965.	4026.	2652.	8643.	0.05700	492651.	1370348.
R 625 NASHVILL, TN	00.	00.	00.	00.				
R 626 KNOXVILL, TN	00.	00.	00.	00.				
R 627 CHATANCO, TN	00.	00.	00.	00.				

R 628 GUNTERSV, AL	00.	00.	00.	00.				
R 629 FLORENCE, AL	00.	00.	00.	00.				
R 630 CATOOSA, OK	00.	00.	00.	00.				
R 631 MUSKOGEE, OK	00.	00.	00.	00.				
R 632 PINEBLUFF, AR	5239.	5950.	00.	9248.				
P 702 NEW ORLE, LA B	5239.	5950.	00.	9248.	20437.	0.06300	1287531.	3240286.
R 633 DES ARC, AR	00.	00.	00.	00.				
R 634 GREENVIL, MS	00.	00.	00.	00.				
R 635 VICKSBUR, MS	00.	00.	00.	00.				
R 636 MONROE, LA	00.	00.	00.	00.				
R 637 LEWISTON, ID	00.	00.	00.	00.				
R 638 CENTRL F, WA	00.	00.	00.	00.				
R 639 PASCO, WA	00.	00.	00.	00.				
R 640 ROOSEVEL, WA	14333.	11168.	3630.	3638.				
P 713 PORTLAND, OR B	14333.	11168.	3630.	3638.	32769.	0.06300	2064447.	5195525.
R 641 THE DALL, OR	00.	00.	00.	00.				
R 642 NEW ORLE, LA	00.	00.	00.	00.				
R 643 PORTLAND, OR	00.	00.	00.	00.				

1

ORIGIN/DESTN	MODE	SUPPLY	SHIPMENT BY TIME			TOTAL SHIPMENT	UNIT COST	TOTAL HAULING COST	HANDLING COSTS
			1	2	3				

P 701 MOBILE, AL	5035.	00.	00.	00.					
F 807 E BLOCK EURO S	5035.	00.	00.	00.	5035.	0.48000	2416800.	284981.	
P 702 NEW ORLE, LA	20541.	25355.	28685.	34305.					
F 807 E BLOCK EURO S	2481.	9146.	4334.	10776.	26737.	0.48000	12833760.	1513314.	
F 808 E MEDITERRAN S	18060.	16209.	18561.	17604.	7034.	0.49500	34864832.	3986564.	
F 814 W ASIA S	00.	00.	5790.	5925.	11715.	0.51900	6080085.	663069.	
P 703 GALVESTN, TX	38.	398.	172.	707.					
F 814 W ASIA S	00.	118.	146.	673.	937.	0.52500	491925.	53034.	
F 821 W MEXICO S	19.	140.	13.	17.	189.	0.27700	52353.	10697.	
F 822 E MEXICO S	19.	140.	13.	17.	189.	0.05900	11151.	10697.	
P 704 CORPUS C, TX	00.	00.	00.	00.					
P 705 BROWNSVI, TX	00.	00.	00.	00.					
P 706 CHARLEST, SC	2748.	3465.	6687.	3628.					
F 803 S.W. EUROPE S	00.	00.	00.	420.	420.	0.33200	139440.	25772.	
F 806 USSR S	00.	115.	480.	119.	714.	0.46200	329868.	40412.	
F 809 N AFRICA S	56.	50.	43.	47.	196.	0.39900	78204.	11094.	
F 810 RED SEA S	258.	1465.	2958.	743.	5424.	0.64500	3498480.	306998.	
F 811 E AFRICA S	6.	37.	315.	312.	670.	0.97000	649900.	37922.	
F 824 CENT AMERICA S	1050.	831.	1559.	1095.	4535.	0.14800	671180.	256681.	
F 825 CARIBBEAN S	1378.	967.	1332.	892.	4569.	0.14800	676212.	258605.	
P 707 BALTIMOR, MD	296.	1444.	2231.	2895.					
F 801 SCANDINAVIA S	00.	00.	00.	57.	57.	0.37500	21375.	3226.	
F 802 N.C. EUROPE S	4.	7.	15.	12.	38.	0.32600	12588.	2151.	
F 803 S.W. EUROPE S	292.	1437.	2216.	2826.	6771.	0.28700	1943277.	383299.	
P 708 TOLEDO, OH	00.	00.	00.	00.					
P 709 SAGINAW, MI	00.	00.	00.	00.					
P 710 CHICAGO, IL	00.	00.	00.	00.					
P 711 DULUTH, MN	00.	00.	00.	00.					
P 712 SEATTLE, WA	42.	425.	3952.	1184.					

F 813 PERSIAN GULF S	00.	27.	3644.	257.	3928.	0.45600	1791168.	222325.
F 815 SE ASIA S	00.	322.	00.	00.	322.	0.37500	120750.	18225.
F 818 JAPAN S	00.	00.	00.	663.	663.	0.22800	151164.	37526.
F 823 WS AMERICA S	42.	76.	308.	264.	690.	0.25700	177330.	39054.
P 713 PORTLAND, OR	35612.	46249.	45544.	50006.				
F 813 PERSIAN GULF S	3010.	5963.	308.	1927.	11208.	0.46100	5166888.	634373.
F 814 W ASIA S	6598.	6480.	662.	00.	13740.	0.54100	7433340.	777684.
F 815 SE ASIA S	2338.	2016.	2338.	2338.	9030.	0.38000	3431400.	511098.
F 816 TAIWAN S	1099.	1236.	1292.	1300.	4927.	0.29600	1458392.	278868.
F 817 KOREA S	13234.	16842.	19282.	15859.	65217.	0.43000	28043310.	3691282.
F 818 JAPAN S	9533.	7764.	8029.	7670.	32796.	0.23300	7641468.	1856254.
F 819 CHINA S	00.	5948.	13633.	20912.	40493.	0.25200	10204236.	2291904.
P 714 SAN FRAN, CA	00.	00.	00.	00.				
P 715 LONG BEA, CA	00.	00.	00.	00.				
P 716 SAN DIEG, CA	00.	00.	00.	00.				
P 717 TOLEDO, BC	00.	00.	00.	00.				
P 718 SAGINAW, BC	00.	00.	00.	783.				
F 808 E MEDITERRAN S	00.	00.	00.	783.	783.	0.81700	639711.	44318.
P 719 CHICAGO, BC	00.	00.	00.	00.				
P 720 DULUTH, BC	00.	00.	00.	00.				

1

	DEMAND BY TIME			TOTAL
	1	2	3	4 SHIPMENT
D 211 Atlanta GA	1778.	1778.	1778.	1778. 7112.
D 163 Battle Cr MI	316.	316.	316.	316. 1264.
D 098 Cabool MO	190.	190.	190.	190. 760.
D 153 Chillico OH	52.	52.	52.	52. 208.
D 391 Denver CO	351.	351.	351.	351. 1404.
D 281 Duboise PA	2691.	2691.	2691.	2691. 10764.
D 183 Knoxville TN	5512.	5512.	5512.	5512. 22048.
D 222 Lake City FL	189.	189.	189.	189. 756.
D 715 Los Angel CA	1045.	1045.	1045.	1045. 4180.
D 096 Oklahoma OK	66.	66.	66.	66. 264.
D 053 Rolla MO	632.	632.	632.	632. 2528.
D 381 Salt Lake UT	1848.	1848.	1848.	1848. 7392.
D 291 Syracuse NY	3069.	3069.	3069.	3069. 12276.
D 151 Tiffin OH	510.	510.	510.	510. 2040.

D 043 Topeka KS	481.	481.	481.	481.	1924.
D 045 Wichita KS	786.	786.	786.	786.	3144.
D 371 Winnemucca NV	148.	148.	148.	148.	592.
D 241 Winston/S NC	1256.	1256.	1256.	1256.	5024.
F 801 SCANDINAVIA	00.	00.	00.	57.	57.
F 802 N.C. EUROPE	4.	7.	15.	12.	38.
F 803 S.W. EUROPE	292.	1437.	2216.	3246.	7191.
F 804 ISLANDS	00.	00.	00.	00.	00.
F 805 ADRIATIC	00.	00.	00.	00.	00.
F 806 USSR	00.	115.	480.	119.	714.
F 807 E BLOCK EURO	7516.	9146.	4334.	10776.	31772.
F 808 E MEDITERRAN	18060.	16209.	18561.	18387.	71217.
F 809 N AFRICA	56.	50.	43.	47.	196.
F 810 RED SEA	258.	1465.	2958.	743.	5424.
F 811 E AFRICA	6.	37.	315.	312.	670.
F 812 W AFRICA	00.	00.	00.	00.	00.
F 813 PERSIAN GULF	3010.	5990.	3952.	2184.	15136.
F 814 W ASIA	6598.	6598.	6598.	6598.	26392.
F 815 SE ASIA	2338.	2338.	2338.	2338.	9352.
F 816 TAIWAN	1099.	1236.	1292.	1300.	4927.
F 817 KOREA	13234.	16842.	19282.	15859.	65217.
F 818 JAPAN	9333.	7764.	8029.	8333.	33459.
F 819 CHINA	00.	5948.	13633.	20912.	40493.
F 820 CANADA	00.	00.	00.	00.	00.
F 821 W MEXICO	19.	140.	13.	17.	189.
F 822 E MEXICO	19.	140.	13.	17.	189.
F 823 WS AMERICA	42.	76.	308.	264.	690.
F 824 CENT AMERICA	1050.	831.	1559.	1095.	4535.
F 825 CARIBBEAN	1378.	967.	1332.	892.	4569.

1

STORAGE COST	25298620.
TRUCK COST	13116200.
RAIL COST	70220272.
BARGE COST	9813835.
SHIP COST	131030400.
HANDLING CST	101200848.

GRAIN SHIPPED FROM SURPLUS REGIONS	406107.
GRAIN SHIPPED TO DEFICIT REGIONS	83680.
GRAIN SHIPPED TO FOREIGN REGIONS	322427.

**Validation of Soft Wheat Model**

<u>Port</u>	<u>FGIS Recorded Exports</u>	<u>Model Solution</u>
Million Bu.		
Mobile	5.0	5.0
New Orleans	109.4	108.9
Galveston	1.3	1.3
Corpus Christi	0	0
Brownsville	0	0
Charleston	23.4 <sup>a</sup>	6.9
Baltimore	-	16.5
Toledo	0.6	0
Saginaw	0	0
Chicago	0	0
Duluth	0.1	0
Seattle	5.6	5.6
Portland	177.4	177.4
California	0	0
San Francisco	-	-
Long Beach	-	-
San Diego	-	-
Baie Comeau	0.3	.8
Toledo	-	-
Saginaw	-	.8
Chicago	-	-
Duluth	-	-
<u>Total</u>	<u>323.2</u>	<u>322.4</u>

<sup>a</sup>Charleston and Baltimore combined.

Special adjustments made in the calibration process:

- 1) Short truck hauls are included for the river ports to move from surplus to barge or barge to deficit where the barge location and surplus deficit region are the same.