COWLES FOUNDATION FOR RESEARCH IN ECONOMICS

AT YALE UNIVERSITY

Box 2125, Yale Station
New Haven, Connecticut

COWLES FOUNDATION DISCUSSION PAPER NO. 115, Part 3

The Construction of a Business Game for Teaching and Research Purposes

Martin Shubik

September 6, 1961

The Construction of a Business Game for Teaching and Research Purposes*

(Part 3)

The Computer Program and Allied Materials

Martin Shubik, with
James Friedman, Theodore Tarson
and Tapan Roy, programmer

---

*Research undertaken by the Cowles Commission for Research in Economics under Task NR 047-006 with the Office of Naval Research.

---

1. Introduction

This part provides flow diagrams, input and output information, the FORTRANSIT program of the game and other material for those wishing to examine, modify or rewrite the program for use with a different machine.

The program has been written for operation with an IBM 650 with minimum memory size (2000 words). FORTRANSIT was selected to facilitate the eventual translation into FORTRAN by those wishing to use the game on larger IBM machines. The price paid for utilizing FORTRANSIT was memory space. The game program was written and compiled in three separate sections. A fourth section which is, in effect, a completely separate program was also written to calculate the solutions predicted by various economic theories.

As has already been observed in Part 1 the current program gives only the needed numerical information as its output. A template is needed to provide the names for the final formats used by the players. This causes little difficulty in running the game; however, for machines with larger memories it would be desirable to have the captions printed out.

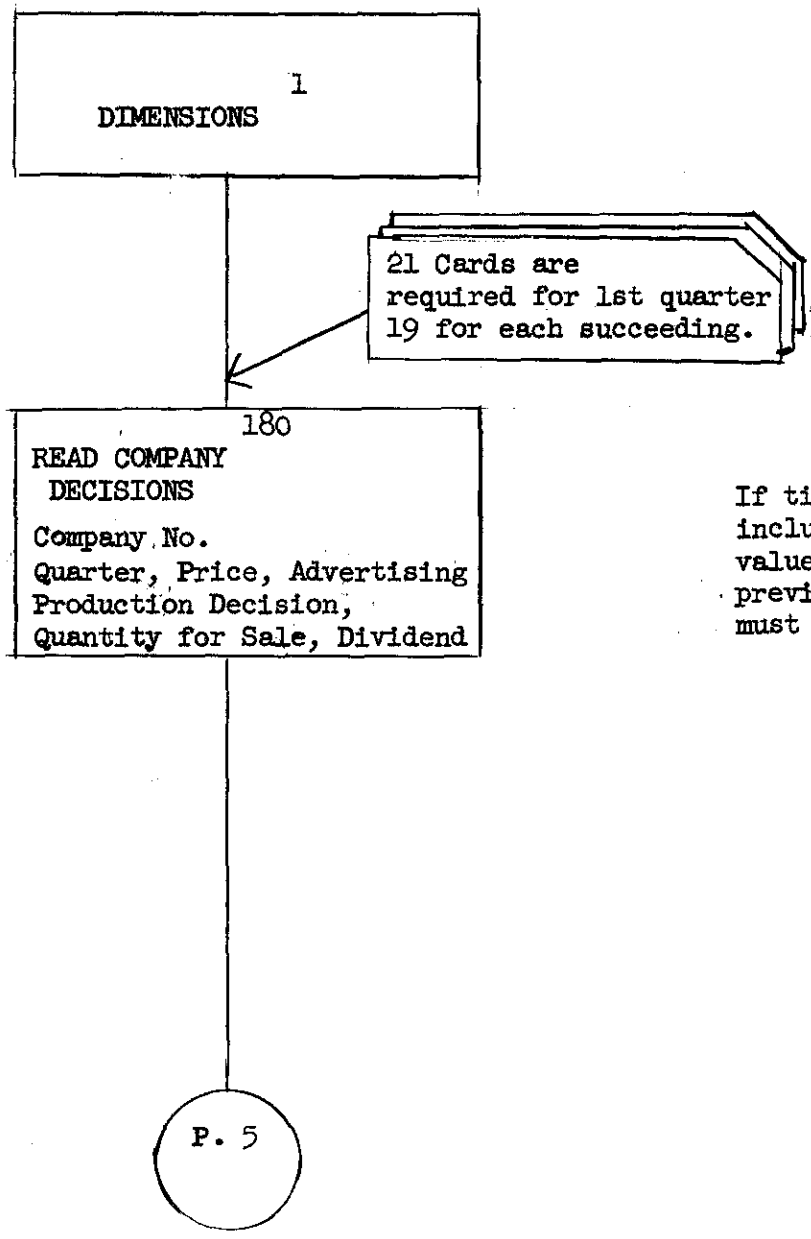
Random numbers are utilized in some parts of the program; however, they are not generated within the program itself. Each quarter, at most seven, are needed. These can be fed in on an input card, having been obtained separately. No discussion is given here of methods employed for random number generation.

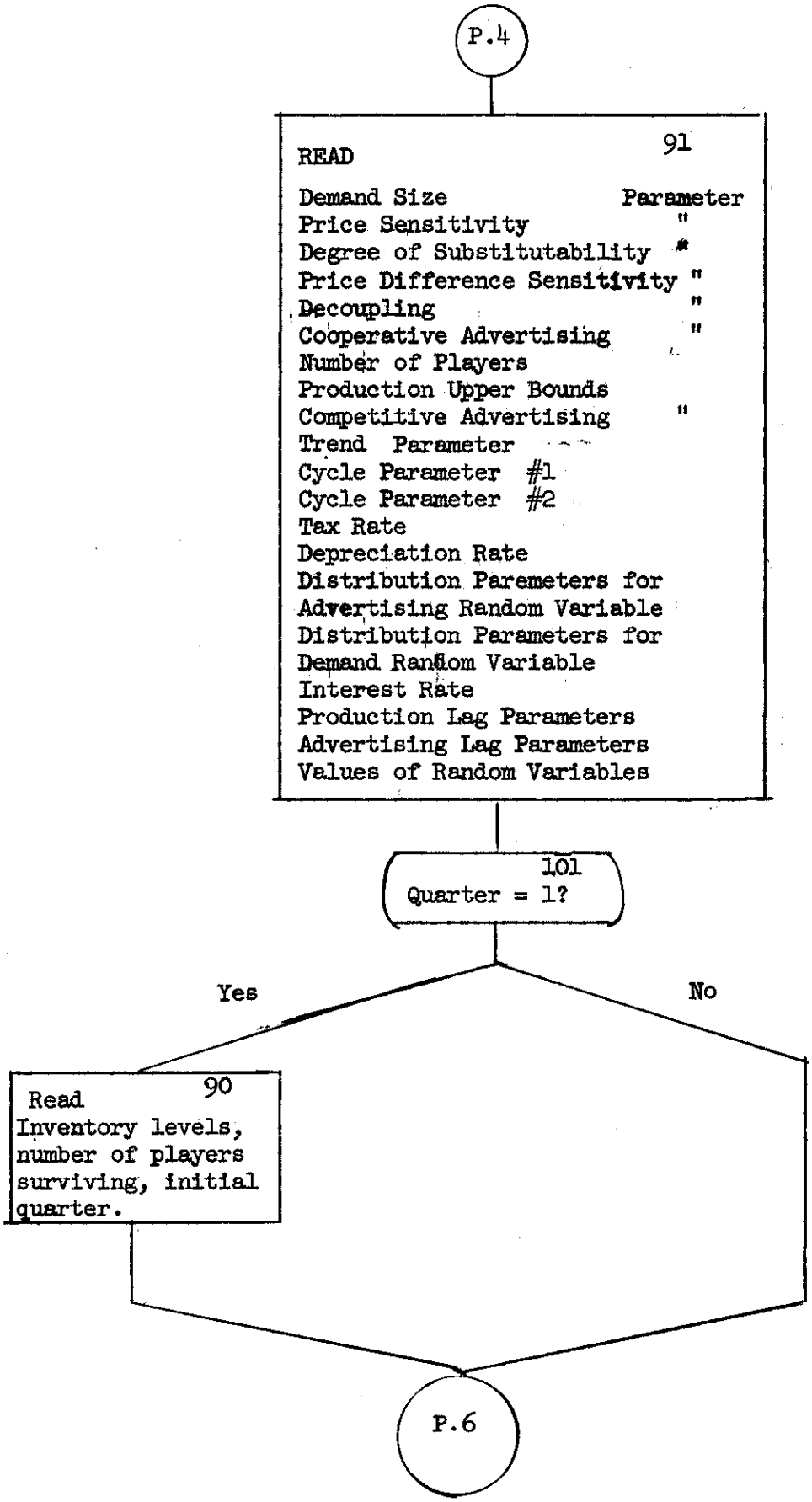
2. Flow Diagrams

The flow diagrams are presented in four parts corresponding to the three sections of the program for the game and a section for calculating solutions. They are highly aggregated with one box referring possibly to many FORTRANSIT instructions. The numbers on the flow diagrams each refer to the instruction in the FORTRANSIT program most relevant to that part of the flow diagram.

PART I

Aggregate Flow Diagrams

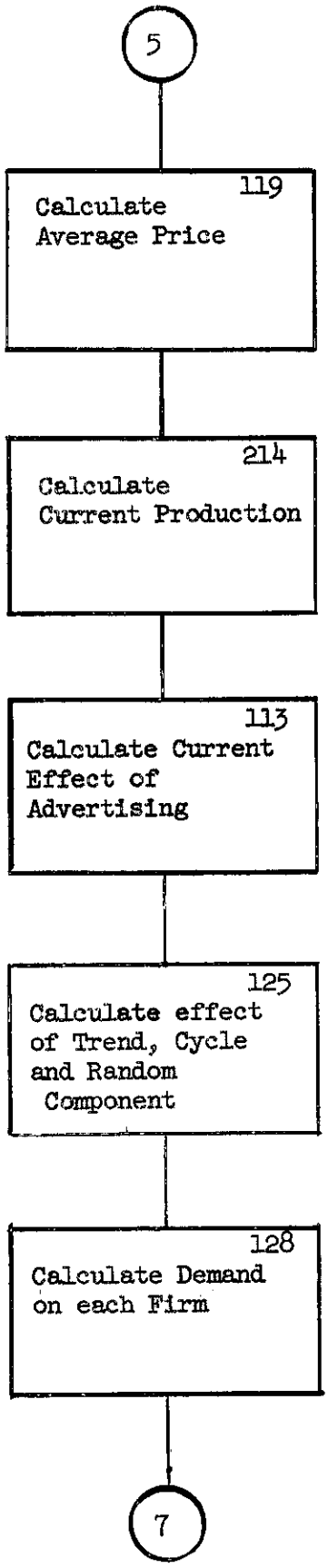




Comments

See CFDP 115 #2A for a discussion of the meaning and role of these values.

After the initial period, this is automatically updated.

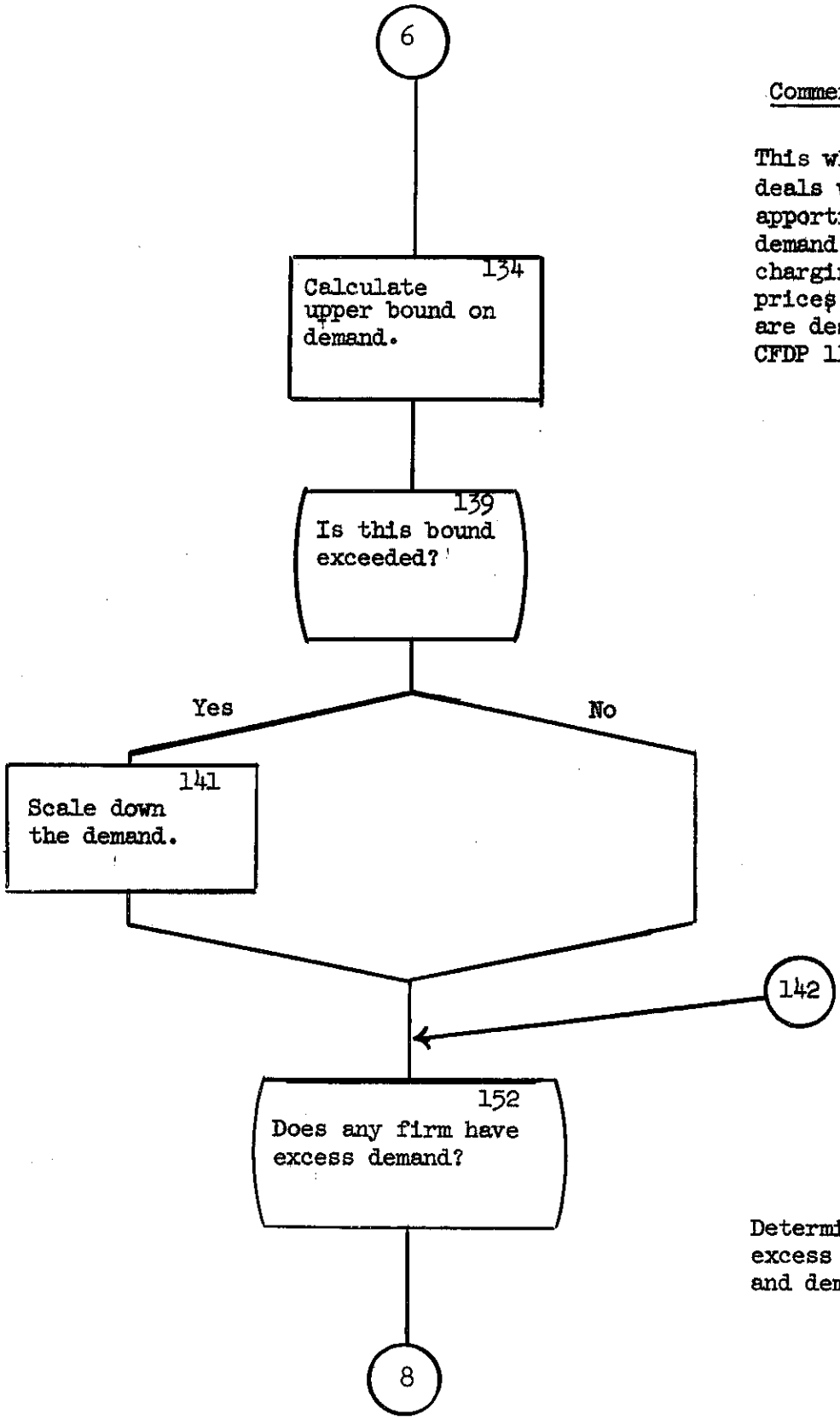


Comments

This accounts for a possible lag of up to 4 periods in the time between the production decision and production.

This includes up to 4 lags and the effect of a random variable if desired.

The numbers in the boxes refer to a key instruction in the FORTRANSIT program.

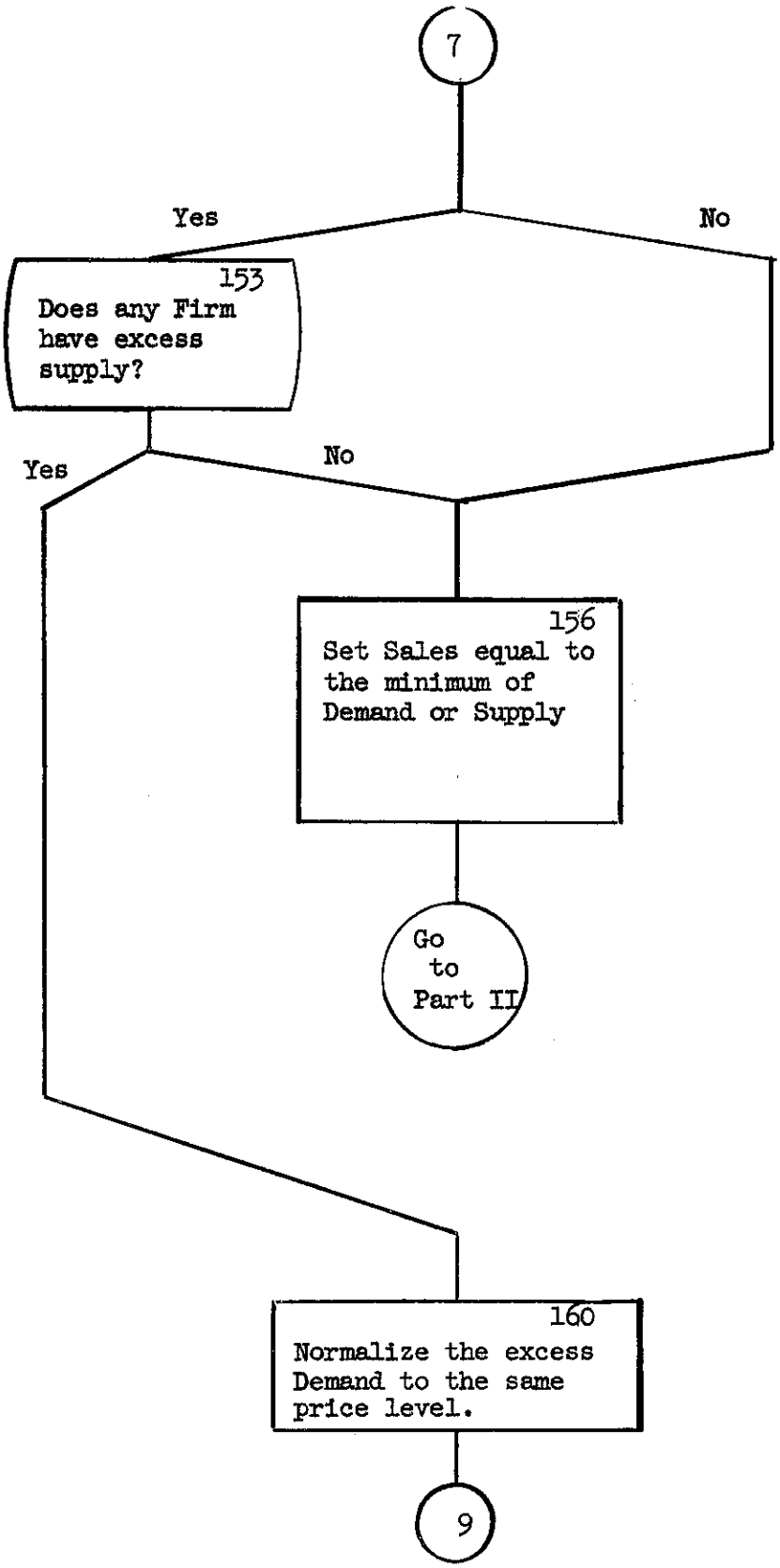


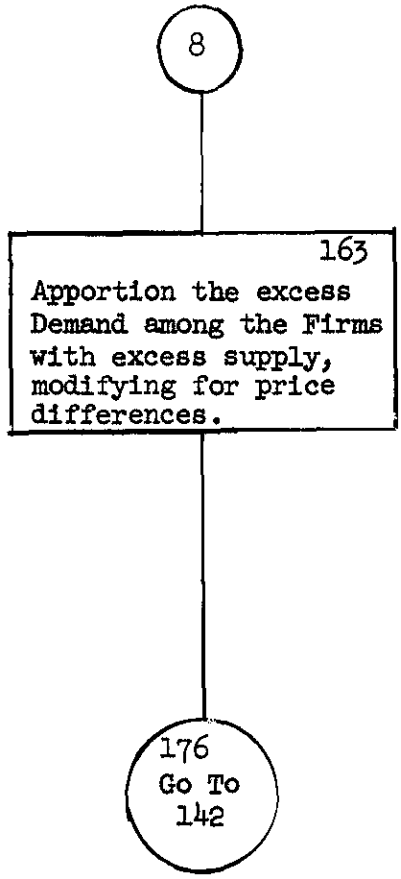
Comments

This whole process deals with the re-apportionment of demand between firms charging various prices. The details are described in CFDP 115, #2A.

Determination of excess supply and demand.

Comments

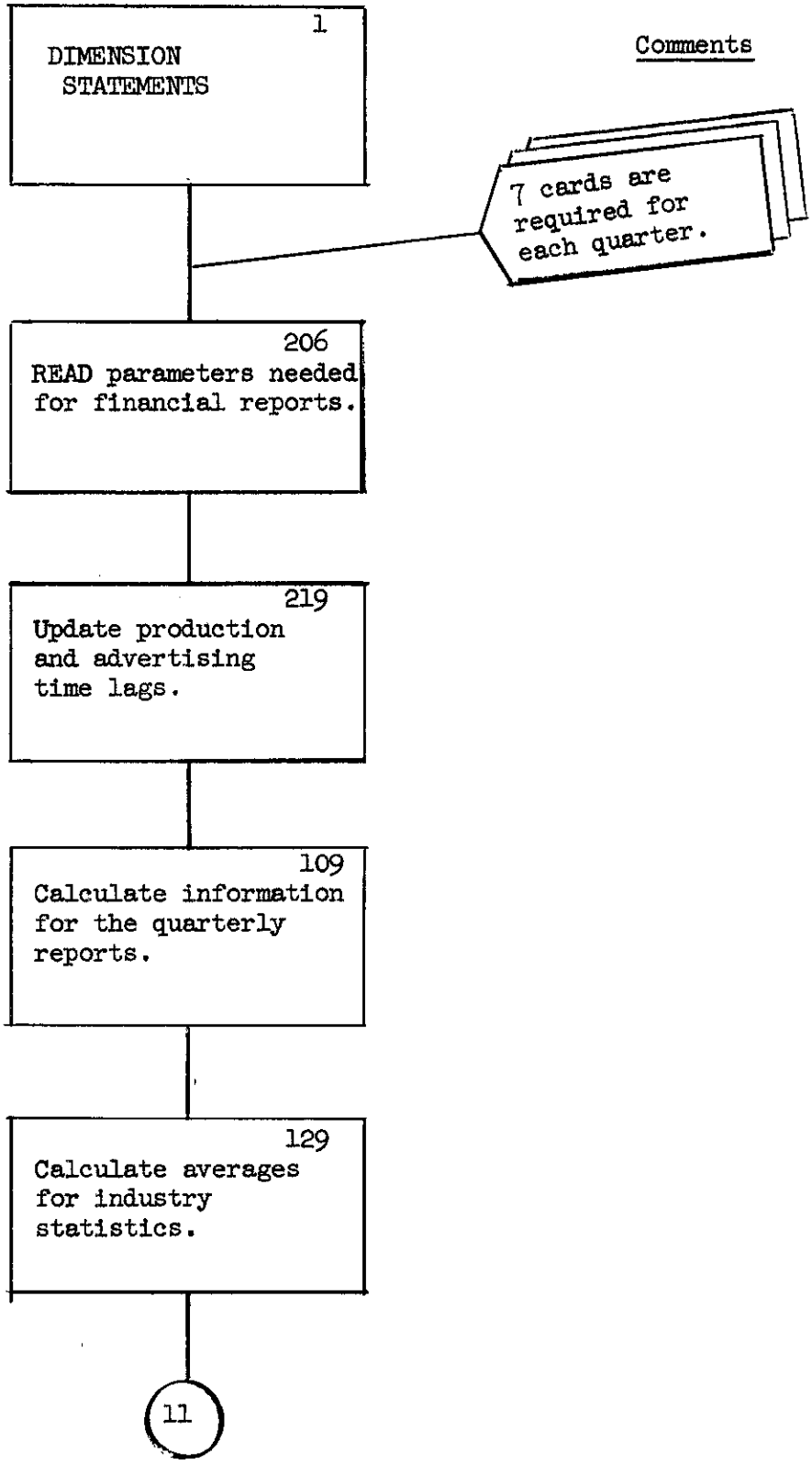

7
Yes
No
153
Does any Firm have excess supply?
Yes
No
156
Set Sales equal to the minimum of Demand or Supply
Go to Part II
160
Normalize the excess Demand to the same price level.
9

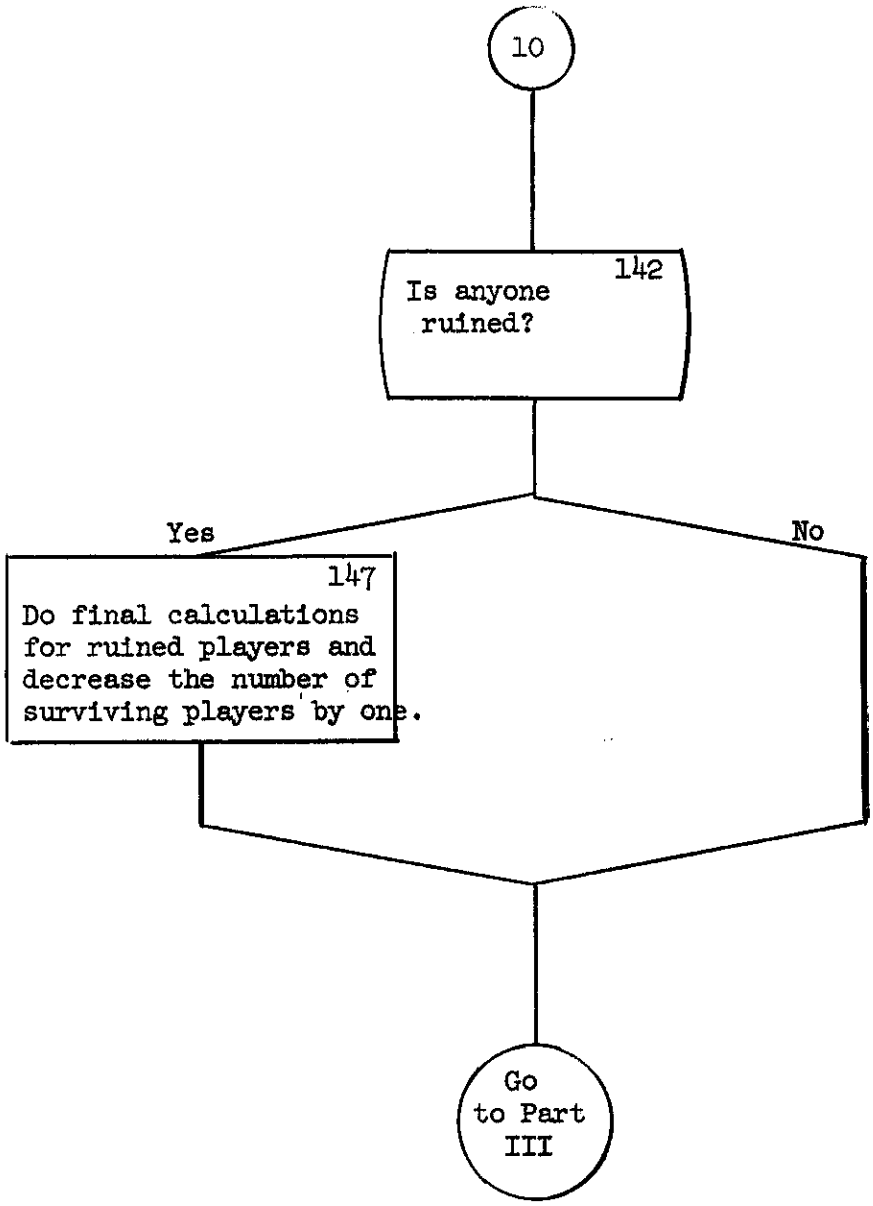


Comments

The program goes to instruction number 142 to re-evaluate excess supply and demand conditions.

PART II. Output

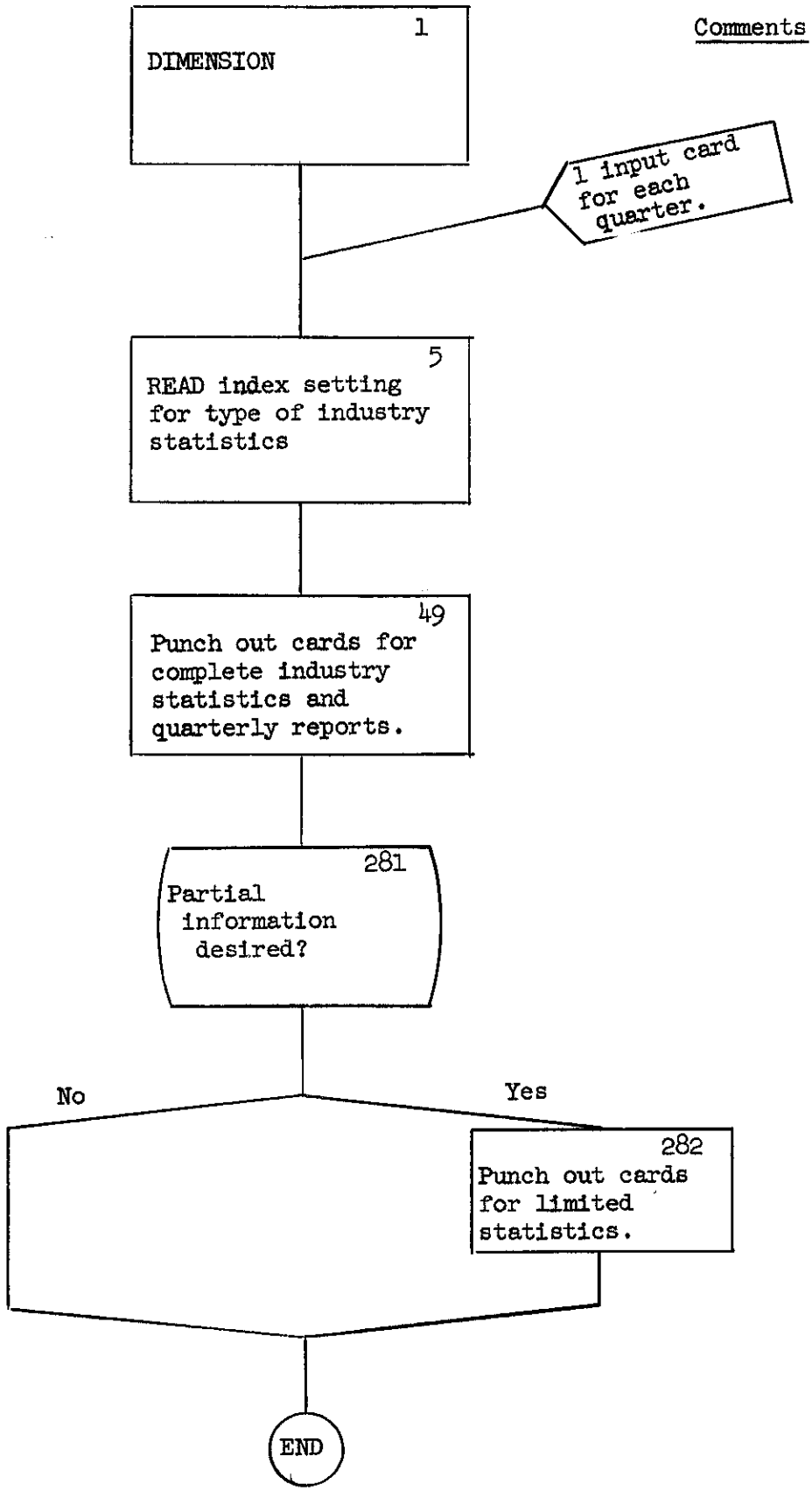




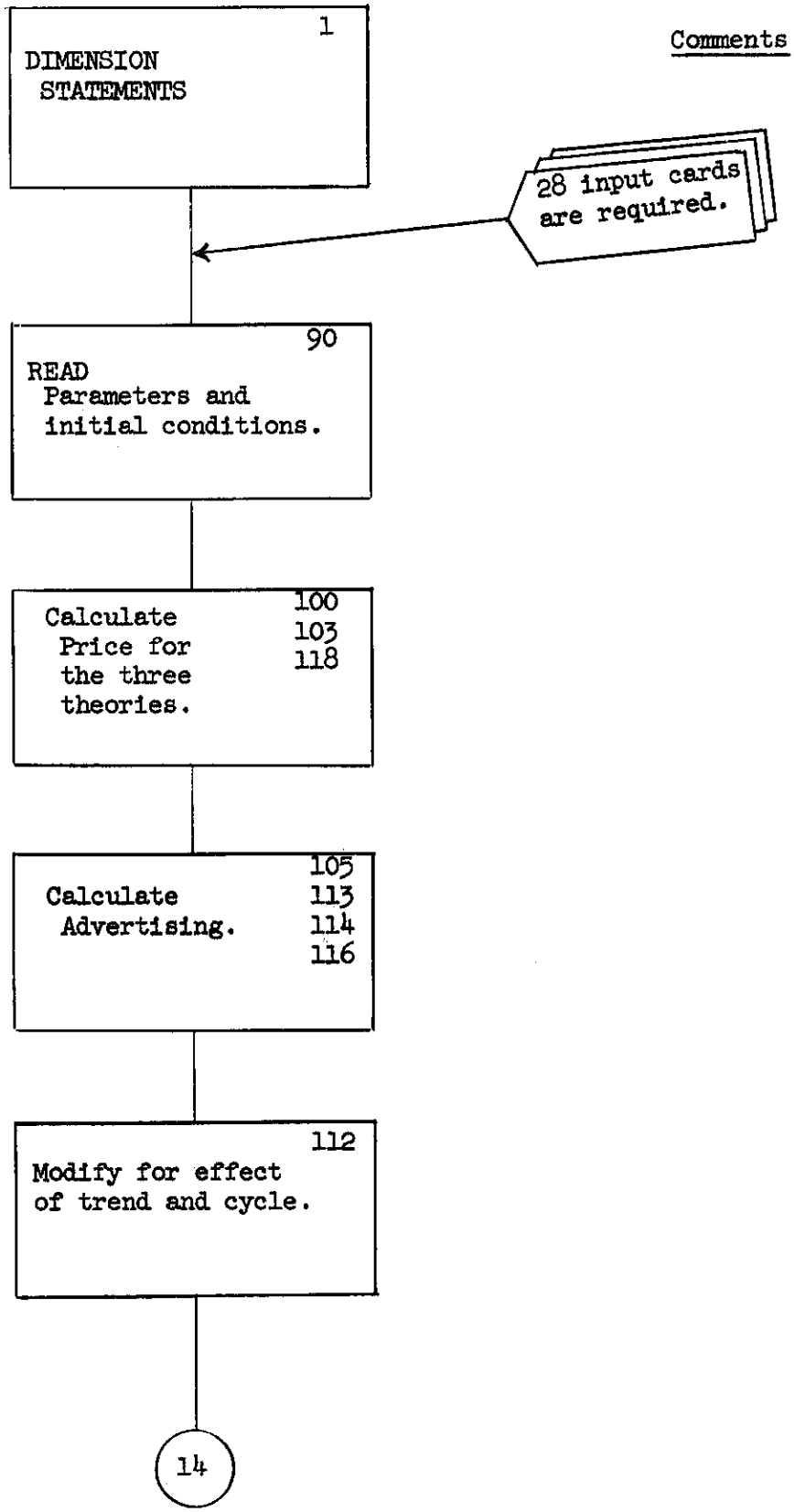
Comments

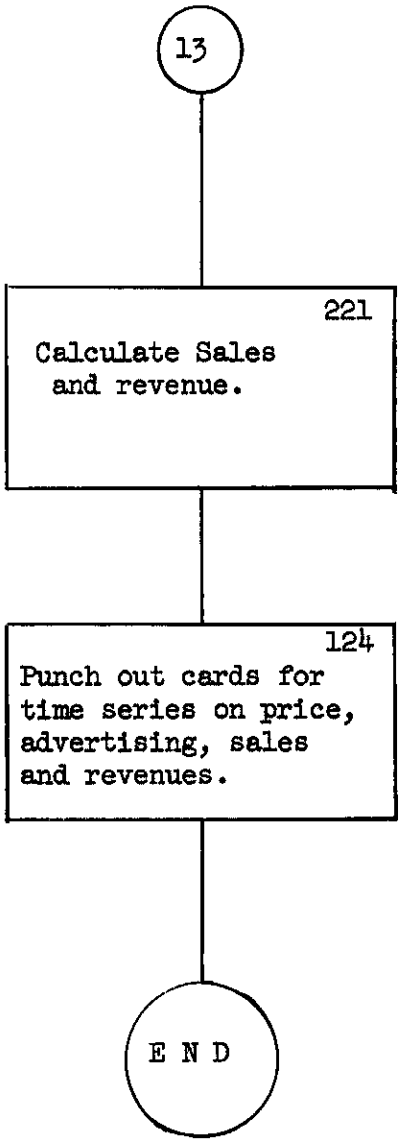
After a player is ruined, dummy moves must be entered each quarter for him as the output formats are not redimensioned.

PART III. Output



PART IV. Theoretical Solutions




13
221
Calculate Sales and revenue.
124
Punch out cards for time series on price, advertising, sales and revenues.
E N D

Comments

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES		
CONO	6	x.x	Company number	i	1, 2, 3, 4, 5, 6		
ADVT	(6,4)	xx,xxx,xxx	Advertising expenditure for player I in period J	a_i	0,0,0	0,0,0	0,0,0
					0,0,0	0,0,0	0,0,0
Q	(6,4)	xx,xxx,xxx	Production decision for player I in period J	q_i	0,0,0,x	0,0,0,x	0,0,0,x
					0,0,0,x	0,0,0,x	0,0,0,x
DVD	6	xx,xxx,xxx	Current dividend payment		x	x	x
					x	x	x
PROD	6	xx,xxx,xxx	Current output	q_i	-	-	-
					-	-	-
	6	xxx,xxx.xx	Price	p_i	x	x	x
					x	x	x
DINV	6	xx,xxx,xxx	Initial inventory level	s_i	100,000.	100,000.	100,000.
					0	0	0
ACSL	6	xx,xxx,xxx	Actual sales		-	-	-
					-	-	-
R1 (I)	4	xx.		-			
R1 (1)	1	xx.	Current size of trend (initially = 1.0)		1.0		
R1 (2)	1	xx.	Number of players surviving	$\hat{n}$	3.0		

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES
R1 (3)	1	xx	Current Quarter		1.0
R1 (4)	1	x.xxx	Cycle Amplitude Parameter	λ	0.0
DUM 1 DUM 2	1	xx	Dummies		-
VLC4 (I)	18	xxx,xxx,xxx			
VLC4 (1) - (6)	6	xxx,xxx,xxx	Short term assets		-
VLC4 (7) (12)	6	xxx,xxx,xxx	Total assets		-
VLC4 (13) - (18)	6	xxx,xxx,xxx	Present value of accumulated dividend payments		-
ASET	6	xxx,xxx,xxx	Long term assets		20,000,000 | 20,000,000 | 20,000,000 - | - | -
CASH	6	xxx,xxx,xxx	Initial cash position		20,000,000 | 20,000,000 | 20,000,000
BQ	6	-	Upper limit on production	Q_1	Not in program computation, but told to the players 500,000
H	(6,4)	xxx.xxx	Weighting factors for lagged effect of advertising		0,0,0,1. | 0,0,0,1. | 0,0,0,1. - | - | -

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES
E	6	xx,xxx,xxx	Advertising effect $E(1)= \sum_{j=1}^{4} H(I,J)* ADVT(IJ)$		-
L	(6,4)	x.xxx	Weighting factor for lag in production.		0,0,0,1. | 0,0,0,1. | 0,0,0,1. - | - | -
DEM	6	xx,xxx,xxx	Demand	d_i	-
S	6	xx,xxx,xxx	Supply		-
DIFE	6	-	Temporary subscripted variable used in demand computation.		-
TR	6	x.xxx	Random variables associated with advertising effect.		0.0 | 0.0 | 0.0 - | - | -
QSLS1 —QSLS6	-	-	Quantity offered for sale.		Not used in current program.
ALPHA	1	xx,xxx,xxx	Demand size parameter	α	1,000,000.
BETA	1	x,xxx,xxx	Price sensitivity parameter.	β	1,000.
SUB	1	x.xxx	Degree of substitutability, parameter 1		1.0 (always has this value in this program)
GAMMA	1	x.xxx	Degree of substitutability, parameter 2	γ	1.0

PROGRAM SYMBOL	DIMEN-SION and RANGE	SIGNI-FICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHE-MATICAL SYMBOL	INITIAL VALUES
DECPR	1	-	"Decoupling" parameter		Not used in program.
ETA	1	x.xxxxx	"Cooperative" advertising parameter	η	0.0001
NPLRS	1	x	Number of players	n	3
THETA	1	x.xxx	"Competitive" advertising parameter	Θ	0.5
AR	1	x.xxx	Trend growth rate per period	T	1.01
OMEGA	1	xxx.xx	(first) cycle parameter	ω	0.0
W	1	xxx.xx	(second) cycle parameter	ν	1.5708 ($\simeq \frac{\Pi}{2}$)
TAX	1	x.xxx	Tax rate		0.5
DEP	1	x.xxx	Depreciation rate		0.05
SIG 1	1	xxx.xxx	Distribution parameters for random variables		0.0
AMU 1	1	xxx.xxx	Effecting advertising		0.0

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES		
SIG 2	1	x.xxx	Distribution parameters for random variables.		0.0		
AMU 2	1	xxx.xxx	Effecting overall economy.		0.0		
ROE	1	x.xxx	Rate of interest	ρ	.01		
RNVR 1	1	x.xxx	Random variable effecting overall economy.	ξ	0.0		
C	6	xx,xxx.xx	Production cost constants.	c_i	99.	99.	99.
					-	-	-
..Y	6		Inventory carrying charge	I_i	2.0	2.0	2.0
					-	-	-
FCST	6		Fixed costs (administrative overheads)		500,000	500,000	500,000
					-	-	-
R	6	xx,xxx,xxx	Ruin levels		0.0	0.0	0.0
VLC1(I)	18	xx,xxx,xxx.			-		

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES
VLC1(1) - 6			Net sales billed.		-
VLC1(7) - 12			Direct cost of sales		-
VLC1(13) - 18			Depreciation charges		-
VLC2(I)	18	xx,xxx,xxx			-
VLC2(1) - 6			Gross Margin		-
VLC2(7) - 12			Inventory carrying charges		-
VLC2(13) - 18			Profit before tax		-
VLC3(I)	26	xx,xxx,xxx			-
VLC3(1) - 6			Tax reserves.		-
VLC3(7) - 12			Profit after tax		-

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES
VLC3(13) - 18			Value of inventory		-
VLC3(19)			Average price.		-
VLC3(20)			Average sales		-
VLC3(21)			Average advertising		-
VLC3(22)			Average production		-
VLC3(23)			Average inventory		-
3(24)			Average profits		-
VLC3(25)			Average dividend payment		-
VLC3(26)			Interest rate (as a %)		-
BU	1	x.xxx	Financial availability parameter.		1.0

PROGRAM SYMBOL	DIMENSION and RANGE	SIGNIFICANT FIGURES	DESCRIPTION OF VARIABLE PARAMETER OR CONSTANT	MATHEMATICAL SYMBOL	INITIAL VALUES
INDEX	1	x	Index to select display of industry statistics		1
LAMBDA	1	x.xxx	R1(4)		1
PRICE	20.3		P		
REV	20.3		VLC3(7-12)		
QUANT	20.3		PROD		
RAN	7.20		RAN(1-6,I) = RNVR(6) RAN(7,I) = RNVR1		
NSURV	20		R1(2)		
QTR	20		R1(3)		
COST	1		C		
R1	1		R1(1)		
M	1	xx.	Number of quarters played or 20, whichever is least.		x
Q	1	xx.	Number of initial quarter being fed in Part IV.		x

ALPHABETIC LIST OF SYMBOLS

Program Symbol	Description of Variable, Parameter or Constant
ACSL	Actual sales.
ADVT	Advertising expenditure for player I in period J .
ALPHA	Demand size parameter.
AMU1	Distribution parameter for random variables affecting advertising.
AMU2	Distribution parameter for random variables affecting overall economy.
ANEW	(Second) cycle parameter.
AR	Trend growth rate per period.
ASET	Long Term Assets
BETA	Price sensitivity parameter.
BQ	Upper limit on production.
BU	Financial availability parameter.
C	Production cost constants.
CARY	Inventory carrying charge.
CASH	Initial Cash Position
CONO	Company number.
Cost	C
DECPR	Decoupling parameter.
DEM	Demand
DEP	Depreciation rate.
DIFE	Temporary subscripted variable used in demand computation.
DINV	Initial inventory level.
DVD	Current dividend payment.

Program Symbol	Description of Variable, Parameter or Constant
E	Advertising effect $E(I) = \sum_{j=1}^{4} H(I,J) * ADVT(I,J)$.
ETA	"Cooperative" advertising parameter.
FCST	Fixed costs (administrative overheads)
GAMMA	Degree of substitutability parameter 2 .
H	Weighting factors for logged effect of advertising.
INDEX	Index to select display of industry statistics.
L	Weighting factor for lag in production.
LAMBDA	R 1 (4)
M	Number of quarters played or 20, whichever is least.
NPLRS	Number of players.
NSURV	R 1 (2)
OMEGA	(first) cycle parameter
P	Price
PRICE	P
PROD	Current output.
Q(Part I)	Production decision for player I in Period J.
Q(Part IV)	Number of initial quarter being fed in Part IV.
QSLS	Quantity offered for sale.
QTR	R 1 (3)
QUANT	PROD
R	Ruin levels
RAN	RAN(1-6,I) = RNVR(6) RAN(7,I) = RNVR 1

Program Symbol	Description of Variable, Parameter or Constant
REV	VLC3 (7 - 12)
RNVR	Random variables associated with advertising effect.
RNVR1	Random variable affecting overall economy.
ROE	Rate of interest.
R1	R1(1)
R1(1)	Current size of trend (initially = 1.0)
R1(2)	Number of players moving.
R1(3)	Current quarter.
R1(4)	Cycle amplitude parameter.
S	Supply
SIG1	Distribution parameter for random variables affecting advertising.
SIG2	Distribution parameter for random variables affecting overall economy.
SUB	Degree of substitutability parameter 1 .
TAX	Tax rate.
THETA	"Competitive" advertising parameter.
VLC1(1-6)	Net sales billed.
VLC1(7-12)	Direct cost of sales.
VLC1(13-18)	Depreciation charges.
VLC2(1-6)	Gross Margin
VLC2(7-12)	Inventory carrying charges.
VLC2(13-18)	Profit before tax.
VLC3(1-6)	Tax reserves.
VLC3(7-12)	Profit after tax.

Program Symbol	Description of Variable, Parameter or Constant
VLC3(13-18)	Value of inventory.
VLC3(19)	Average price.
VLC3(20)	Average sales.
VLC3(21)	Average advertising.
VLC3(22)	Average production.
VLC3(23)	Average inventory.
VLC3(24)	Average profits.
VLC3(25)	Average dividend payment.
VLC3(26)	Interest rate (as a %)
VLC4(1-6)	Short term assets.
VLC4(7-12)	Total assets.
VLC4(13-18)	Present value of accumulated dividend payments.

4. Computor Program

Part I

```
  1 DIMENSION CONO(6),       ADVT(
  116,4),Q(6,4),DVD(6),PROD(6),
  12P(6),DINV(6)
  2 DIMENSION ACSL(6),R1(4),VLC4
  21(18),ASET(6),CASH(6)
  3 DIMENSION         BQ(6),H(6,4
  31,E(6),L(6,4),DEM(6),S(6),DIFE(
  326),RNVR(6)
180 READ,CONO(1),R1(3) ,P(1),ADVT(
18011,4),Q(1,4),QSLS1  ,DVD(1)
181 READ,CONO(2),R1(3) ,P(2),ADVT(
18112,4),Q(2,4),QSLS2  ,DVD(2)
182 READ,CONO(3),R1(3) ,P(3),ADVT(
18213,4),Q(3,4),QSLS3  ,DVD(3)
183 READ,CONO(4),R1(3) ,P(4),ADVT(
18314,4),Q(4,4),QSLS4  ,DVD(4)
184 READ,CONO(5),R1(3) ,P(5),ADVT(
18415,4),Q(5,4),QSLS5  ,DVD(5)
185 READ,CONO(6),R1(3) ,P(6),ADVT(
18516,4),Q(6,4),QSLS6  ,DVD(6)
 91 READ,      ALPHA,BETA,SUB,GAMM
 911A,DECPR,ETA,NPLRS,BQ
 92 READ,THETA,AR,   OMEGA,ANEW,
 921TAX,DEP,SIG1,AMU1
 93 READ,SIG2,AMU2,ROE,L,H,DUM2
202 READ,RNVR,RNVR1
101 IF (R1(3) -1.0) 90,90,106
 90 READ,DINV,R1
106 CUMAD=0.0
107 CUMPD=0.0
117 AVPR   =0.0
108 DO116I=1,NPLRS
110 F(I)=0.0
114 PROD(I)=0.0
119 AVPR   =AVPR   +P(I)
109 DO214J=1,4
210 E(I)=E(I)+H(I,J)*ADVT(I,J)
214 PROD(I)=PROD(I)+L(I,J)*Q(I,J)
112 RNV=RNVR(I)*SIG1   +AMU1
113 E(I)=E(I)+RNV
115 CUMAD=CUMAD+E(I)
116 CUMPD=CUMPD+PROD(I)
120 AVPR   =AVPR/R1(2)
```

```
122 RNV=RNVR1   *SIG2  +AMU2
123 R1(1)=R1(1)*AR
125 TEMD2=R1(1)*(R1(4)*SINF(OM
1251 EGA*R1(3)+ANEW)+RNV)
126 DO133I=1,NPLRS
127 DEM(I)=(ALPHA   -BETA  *(P(I)
1271+GAMMA   *(P(I)-AVPR  )))
1272/R1(2)
228 IF (CUMAD) 229,229,124
229 TEMD1=1.0
230 GO TO 130
124 TEMD1=(1.0+ETA   *(SQRTF(CUMAD
1241)))
128 DEM(I)=DEM(I)*TEMD1
129 DEM(I)=DEM(I)*((1.0-THETA   )*
1291(E(I)/CUMAD)*R1(2) +  THETA)
130 DEM(I)=DEM(I)*TEMD2
    PUNCH,DEM(I)
131 IF(DEM(I)-0.0) 132,133,133
132 DEM(I)=0.0
133 CONTINUE
134 UPSIS=(ALPHA   -BETA   *AVPR)*
1341TEMD1
135 UPSIS=UPSIS*TEMD2
136 TDEMD=0.0
137 DO138I=1,NPLRS
138 TDEMD=TDEMD+DEM(I)
139 IF(TDEMD-UPSIS)142,142,140
140 DO19CI=1,NPLRS
141 DEM(I)=DEM(I)*(UPSIS/TDEMD)
    PUNCH,DEM(I)
190 CONTINUE
142 ESUP=0.0
143 EDEM=0.0
144 DO191I=1,NPLRS
145 S(I)=DINV(I)+PROD(I)/2.0
191 CONTINUE
146 DO151I=1,NPLRS
147 EXESS=DEM(I)-S(I)
148 IF(EXESS)250,250,149
149 EDEM=EDEM+EXESS
150 GO TO 151
250 ESUP=ESUP+EXESS
151 CONTINUE
152 IF(EDEM)154,154,153
153 IF(ESUP)160,154,154
154 DO158I=1,NPLRS
255 IF(DEM(I)-S(I))156,156,257
```

```
156 ACSL(I)=DEM(I)
    PUNCH,ACSL(I)
157 GO TO 158
257 ACSL(I)=S(I)
    PUNCH,ACSL(I)
158 CONTINUE
159 GO TO 177
160 ESUP=0.0
161 NDEM=0.0
162 DO171I=1,NPLRS
163 IF(DEM(I)-S(I))166,167,167
166 IF (P(I)-(ALPHA/BETA))165,171,
1661171
165 NDEM=NDEM+1
266 GO TO 171
167 DIFE(I)=DEM(I)-S(I)
168 DEM(I)=DINV(I)+PROD(I)/2.0
169 DIFE(I)=DIFE(I)*(ALPHA   /BETA
1691)   /((ALPHA   /BETA   )-P(I))
    PUNCH,DIFE(I)
170 ESUP=ESUP+DIFE(I)
171 CONTINUE
172 DO175I=1,NPLRS
272 IF(P(I)-(ALPHA/BETA))173,175,
2721175
173 IF(DEM(I)-S(I))174,175,175
174 DEM(I)=DEM(I)+(ESUP/NDEM)*(1.0
1741-(P(I)/(ALPHA   /BETA   )))
175 CONTINUE
176 GO TO 142
177 END
```

Part II

```
  1 DIMENSION CONO(6),       ADVT(
  116,4),Q(6,4),DVD(6),PROD(6),
  12P(6),DINV(6)
  2 DIMENSION ACSL(6),R1(4),VLC4
  21(18),ASET(6),CASH(6)
  4 DIMENSION C(6),CARY(6),FCST(6)
  41,R(6)
  3 DIMENSION VLC1(18),VLC2(18),VL
  31C3(26)
206 READ,C,CARY,FCST,R
103 READ,DEP,TAX,ROE,BU,NPLRS,INDE
1031X
100 IF(R1(3) -1.0)101,101,104
101 READ,ASET,CASH,DUM1,DUM2
104 AVPR=0.0
105 DO 106 I=1,NPLRS
106 AVPR=AVPR+P(I)
107 AVPR=AVPR/R1(2)
217 DO220I=1,NPLRS
218 DO220J=1,3
219 Q(I,J)=Q(I,J+1)
220 ADVT(I,J)=ADVT(I,J+1)
108 DO 121 I=1,NPLRS
109 DINV(I)=DINV(I)-ACSL(I)+PROD(I
1091)
110 VLC1(I)=P(I)*ACSL(I)
111 VLC1(I+6)=C(I)*ACSL(I)
112 VLC1(I+12)=DEP*ASET(I)
113 VLC2(I)=VLC1(I)-VLC1(I+6)-VLC1
1131(I+12)
114 VLC2(I+6)=CARY(I)* (DINV(I)-((
1141PROD(I)-ACSL(I))/2.0))
115 VLC2(I+12)=VLC2(I)-ADVT(I,4)-V
1151LC2(I+6)-FCST(I)
116 VLC3(I)=VLC2(I+12)*TAX
117 VLC3(I+6)=VLC2(I+12)-VLC3(I)
118 CASH(I)=CASH(I)+VLC3(I+6)-DVD(
1181I)-C(I)*(PROD(I)-ACSL(I))
119 VLC3(I+12)=C(I)*DINV(I)
120 VLC4(I)=CASH(I)+VLC3(I+12)
121 VLC4(I+6)=VLC4(I)+ASET(I)
122 TSLS=0.0
123 TINV=0.0
124 TPRO=0.0
125 TDVD=0.0
126 TPROD=0.0
```

```
127 TADVT=0.0
128 DO 134 I=1,NPLRS
129 TSLS=TSLS+ACSL(I)
130 TINV=TINV+DINV(I)
131 TPRO=TPRO+VLC3(I+6)
132 TDVD=TDVD+DVD(I)
133 TPROD=TPROD+PROD(I)
134 TADVT=TADVT+ADVT(I,4)
135 TSLS=TSLS/R1(2)
136 TINV=TINV/R1(2)
137 TPRO=TPRO/R1(2)
138 TDVD=TDVD/R1(2)
139 TPROD=TPROD/R1(2)
140 TADVT=TADVT/R1(2)
240 DO 148 I=1,NPLRS
141 VLC4(I+12)=(1.0+ROE)*VLC4(I+12
1411)+DVD(I)
142 IF (R(I)-CASH(I)-BU*VLC4(I+12)
1421 ) 143,143,147
143 IF (R(I)-CASH(I)) 148,148,144
144 VLC4(I+12)=VLC4(I+12)+CASH(I)-
1441R(I)
145 CASH(I)=R(I)
146 GO TO 148
147 CASH(I)=CASH(I)+BU*VLC4(I+12)
247 VLC4(I+12)=(1.0-BU)*VLC4(I+12)
148 CONTINUE
349 VLC3(19)=AVPR
350 VLC3(20)=TSLS
351 VLC3(21)=TADVT
352 VLC3(22)=TPROD
353 VLC3(23)=TINV
354 VLC3(24)=TPRO
355 VLC3(25)=TDVD
356 VLC3(26)=100.0*ROE
176 OUT=999999.9
177 DO 181 I=1,NPLRS
178 IF (R(I)-CASH(I)) 181,181,179
179 R1(2)=R1(2)-1.0
280 F=FIXF(-10)
180 PUNCH,CONO(I),R1(3),OUT
181 CONTINUE
357 END
```

Part III

```
  1 DIMENSION CONO(6),       ADVT(
  116,4),Q(6,4),DVD(6),PROD(6),
  12P(6),DINV(6)
  2 DIMENSION ACSL(6),R1(4),VLC4
  21(18),ASET(6),CASH(6)
  4 DIMENSION C(6),CARY(6),FCST(6)
  41,R(6)
  3 DIMENSION VLC1(18),VLC2(18),VL
  31C3(26)
  5 READ,NPLRS,INDEX
249 ZERO=0.0
 49 F=FIXF(-10)
149 PUNCH,P,VLC3(19)
250 F=FIXF(-10)
150 PUNCH,ACSL,VLC3(20)
251 F=FIXF(-10)
151 PUNCH,ADVT(1,4),ADVT(2,4),ADVT
1511(3,4),ADVT(4,4),ADVT(5,4),ADVT
1512(6,4),VLC3(21)
252 F=FIXF(-10)
152 PUNCH,PROD,VLC3(22)
253 F=FIXF(-10)
153 PUNCH,DINV,VLC3(23)
254 F=FIXF(-10)
154 PUNCH,VLC3(7),VLC3(8),VLC3(9),
1541VLC3(10),VLC3(11),VLC3(12),VLC
15423(24)
255 F=FIXF(-10)
155 PUNCH,DVD,VLC3(25)
256 F=FIXF(-10)
156 PUNCH,ZERO,ZERO,ZERO,VLC3(26)
257 F=FIXF(-10)
157 PUNCH,VLC4(13),VLC4(14),VLC4(1
15715),VLC4(16),VLC4(17),VLC4(18)
158 DO 175 I=1,NPLRS
259 F=FIXF(-10)
159 PUNCH,ZERO,ZERO,CONO(I),ZERO
1591,R1(3)
260 F=FIXF(-10)
160 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1601,VLC1(1)
261 F=FIXF(-10)
161 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1611,VLC1(I+6)
262 F=FIXF(-10)
162 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
```

```
1621,VLC1(I+12)
263 F=FIXF(-10)
163 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1631,VLC2(I)
264 F=FIXF(-10)
164 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1641,ADVT(I,4)
265 F=FIXF(-10)
165 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1651,VLC2(I+6)
266 F=FIXF(-10)
166 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1661,FCST(I)
267 F=FIXF(-10)
167 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1671,VLC2(I+12)
268 F=FIXF(-10)
168 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1681,VLC3(I)
269 F=FIXF(-10)
169 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1691,VLC3(I+6)
270 F=FIXF(-10)
170 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1701,CASH(I)
272 F=FIXF(-10)
172 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1721,VLC3(I+12)
271 F=FIXF(-10)
171 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1711,VLC4(1)
273 F=FIXF(-10)
173 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1731,ASET(I)
274 F=FIXF(-10)
174 PUNCH,ZERO,ZERO,ZERO,ZERO,ZERO
1741,VLC4(I+6)
175 CONTINUE
281 GO TO (182,205),INDEX
182 DO191I=1,NPLRS
 82 F=FIXF(-10)
282 PUNCH,ZERO,ZERO,CONO(I),ZERO
2821,R1(3)
283 F=FIXF(-10)
183 PUNCH,P(I),ZERO,ZERO,ZERO,ZERO
1831,VLC3(19)
284 F=FIXF(-10)
184 PUNCH,ACSL(I),ZERO,ZERO,ZERO,Z
1841ERO,VLC3(20)
285 F=FIXF(-10)
```

```
185 PUNCH,ADVT(I,4),ZERO,ZERO,ZERO
1851,ZERO,VLC3(21)
286 F=FIXF(-10)
186 PUNCH,PROD(I),ZERO,ZERO,ZERO,
1861ZERO,VLC3(22)
 87 F=FIXF(-10)
187 PUNCH,DINV(I),ZERO,ZERO,ZERO,Z
1871ERO,VLC3(23)
 88 F=FIXF(-10)
287 PUNCH,VLC3(I+6),ZERO,ZERO,ZERO
2871,ZERO,VLC3(24)
288 F=FIXF(-10)
188 PUNCH,DVD(I),ZERO,ZERO,ZERO,ZE
1881RO,VLC3(25)
289 F=FIXF(-10)
189 PUNCH,VLC4(I+12)
290 F=FIXF(-10)
190 PUNCH,ZERO,ZERO,ZERO,VLC3(26)
191 CONTINUE
205 END
```

Part IV

```
  1 DIMENSION PRICE(20,3),ADVT(20,
  113),REV(20,3),QUANT(20,3),RAN(7
  12,20),NSURV(20),QTR(20)
 90 READ,ALPHA,COST,BETA,GAMMA,ETA
 901,THETA,SIG2,AMU2,Q
 91 READ,R1,AR,OMEGA,ANEW,DEP,ASET
 911,CARY,FCST,LAMBDA
 92 READ,RAN,M,TAX,NSURV
200 COST = COST + CARY/2.0
100 PRIC1  = (ALPHA + COST *BETA)/
1001(2.0 *BETA)
101 ADVT3 = 0.0
202 QTR(1) = Q - 1.0
102 DO 122 NOW = 1,M
103 PRICE(NOW,2) = (NSURV(NOW) *AL
1031PHA + COST *BETA *(NSURV(NOW)
1032+ (NSURV(NOW) - 1.0) *GAMMA))
104 PRICE(NOW,2) = PRICE(NOW,2)/
1041(BETA *(2.0 *NSURV(NOW) +
1042(NSURV(NOW) - 1.0) *GAMMA))
105 ADVT1 = (((ALPHA - COST *BETA)
1051**2) *ETA/(8.0 *BETA *NSURV
1052(NOW)))**2
106 B = (((1.0 - THETA)/NSURV(NOW)
1061) + THETA) *ETA/2.0
107 C = (1.0 - THETA) *((NSURV(NOW
1071) - 1.0)/NSURV(NOW))
108 A1 = NSURV(NOW)/((PRICE(NOW,2)
1081- COST) *(ALPHA - BETA *PRICE
1082(NOW,2)))
109 RNV = RAN(7,NOW) *SIG2 + AMU2
110 R1 = R1 *AR
111 QTR(NOW) = QTR(NOW + 1.0
112 TEMD2 = R1*(LAMBDA*SINF(OMEGA
1121*QTR(NOW) + ANEW) + RNV)
113 ADVT(NOW,1) = ADVT1 *((TEMD2)
1131**2)
114 ADVT(NOW,3) = ADVT3
115 A = A1/TEMD2
116 ADVT(NOW,2) = ((B + ETA *C +
1161SQRTF((B + ETA *C)**2 + 4.0 *A
1162*C))/(2.0 *A))**2
117 PRICE(NOW,1) = PRIC1
118 PRICE(NOW,3) = COST
119 DO 122 I=1,3
120 QUANT(NOW,I) =  (ALPHA - BETA
```

```
1201*PRICE(NOW,I)) *(1.0 + ETA *
1202SQRTF(ADVT(NOW,I))) *TEMD2/
1203NSURV(NOW)
121 REV(NOW,I) = (PRICE(NOW,I) -
1211COST) *QUANT(NOW,I) - ADVT(NOW
1212,I) - DEP *ASET - FCST
221 REV(NOW,I) = REV(NOW,I) *(1.0
2211- TAX)
122 CONTINUE
123 DO 125 NOW=1,M
224 F=FIXF(-8)
124 PUNCH,QTR(NOW),PRICE(NOW,1)
1241,PRICE(NOW,2),PRICE(NOW,3)
125 CONTINUE
126 DO 128 NOW=1,M
227 F=FIXF(-10)
127 PUNCH,QTR(NOW),ADVT(NOW,1),AD
1271VT(NOW,2),ADVT(NOW,3)
128 CONTINUE
129 DO 131 NOW=1,M
230 F=FIXF(-10)
130 PUNCH,QTR(NOW), QUANT(NOW,1),
1301QUANT(NOW,2),QUANT(NOW,3)
131 CONTINUE
132 DO 134 NOW=1,M
233 F=FIXF(-10)
133 PUNCH,QTR(NOW), REV(NOW,1),REV
1331(NOW,2),REV(NOW,3)
134 CONTINUE
135 END
```

5. Hand Simulation and "Debugging"

It is invariably highly desirable to hand compute several test cases of a program before attempting to assemble the program. Even after assembly it is a great convenience, and possibly a necessity, to have a computation to compare with the output of the machine. In general several simple cases provide the best means of checking a program. If numbers are set equal to 0 or 1 without sufficient care being exercised, there is a danger that errors may be masked by oversimplification. If, on the other hand, complicated debugging cases are used, the chances for error in the hand computation or misinterpretation of the results becomes great.

A sample of a debugging run for this program is presented. The instructions were hand computed and compared with the trace given below.

The first column of the input gives the number of the read instruction associated with the data input. The input cards each can feed in up to seven words; thus, for example, the construction 91 begins with the value of ALPHA in floating point 2100000057 or 2,100,000. Instruction 93 calls for both H and L to be read in. These are both two dimensional arrays of (6,4), hence each require (6,4) = 24 spaces. Note that the input listing prints blanks for 10-character words containing only zeroes.

The trace indicates what is taking place at each instruction in the program as it is executed; thus, for example, the instruction 119 used in the computation of average price appears as is indicated by the arrows.

Case I. Data Input

	Word 1 Cols. 1-10	Word 2 Cols.11-20	Word 3 Cols.21-30	Word 4 Cols.31-40	Word 5 Cols.41-50	Word 6 Cols.51-60	Word 7 Cols.61-70
180	1000000051	2000000051	2000000053				
181	2000000051	2000000051	4000000053				
182	3000000051	2000000051	3000000053				
183	4000000051	2000000051					
184	5000000051	2000000051					
185	6000000051	2000000051					
→ 91	2100000057	6000000054	1000000051	1000000051		1000000047	3
911	5000000056	5000000056	5000000056				
92	5000000050		1000000051	3000000051	2000000051		
921		5000000050	5000000049				
93		1000000051	5000000049				
93							
93							
93	1	1	1				
93							
93							
93				1000000051	1000000051	1000000051	

NOTE: Word eight on each data input card contains a numerical identification of the user's choosing. Whatever the identification, column 73 must contain a " + " punch (a "high" or "row 12" punch).

CASE I

Trace, first page

(Words 2-7 are all blank and therefore are not reproduced here.)

Word 1 (Columns 1-10)	Word 8 (Columns 71-80)
+	10106+
+	20107+
+	30117+
1+	40108+
+	50110+
+	60114+
→ 2000000053+	70119+ ←
1+	80109+
+	90210+
+	100214+
+	110210+
+	120214+
+	130210+
+	140214+
3000000056+	150210+
2000000056+	160214+
2000000055+	170112+
3200000056+	180113+
3200000056+	190115+
2000000056+	200116+
+	210110+
+	220114+
→ 4000000053+	230119+ ←
1+	240109+
+	250210+
+	260214+
+	270210+
+	280214+
+	290210+
+	300214+
3000000056+	310210+
2000000056+	320214+
1000000055-	330112+
2900000056+	340113+
6100000056+	350115+
4000000056+	360116+
+	370110+
+	380114+
→ 6000000053+	390119+ ←

Trace, second page

Word 1 (Columns 1-10)	Word 8 (Columns 71-80)
1+	400109+
+	410210+
+	420214+
+	430210+
+	440214+
+	450210+
+	460214+
4000000056+	470210+
2000000056+	480214+
1000000055-	490112+
3900000056+	500113+
1000000057+	510115+
6000000056+	520116+
2000000053+	530120+
1000000051+	540122+
1000000051+	550123+
1000000051+	560125+
1+	570126+
3000000056+	580127+
1000000057+	590228+
1100000051+	600124+
3300000056+	610128+
3234000056+	620129+
3234000056+	630130+
3234000056+	640000+
3234000056+	650131+
1000000051+	660133+
3000000056+	670127+
1000000057+	680228+
1100000051+	690124+
3300000056+	700128+
3085500056+	710129+
3085500056+	720130+
3085500056+	730000+
3085500056+	740131+
1000000051+	750133+
3000000056+	760127+
1000000057+	770228+
1100000051+	780124+
3300000056+	790128+
3580500056+	800129+
3580500056+	810130+
3580500056+	820000+
3580500056+	830131+
1000000051+	840133+
9900000056+	850134+

Trace, third page

Word 1 (Columns 1-10)	Word 8 (Columns 71-80)
9900000056+	860135+
+	870136+
1+	880137+
3234000056+	890138+
6319500056+	900138+
9900000056+	910138+
+	920139+
+	930142+
+	940143+
1+	950144+
2000000056+	960145+
1000000051+	970191+
2000000056+	980145+
1000000051+	990191+
2000000056+	1000145+
1000000051+	1010191+
1+	1020146+
1234000056+	1030147+
1234000056+	1040148+
1000000051+	1050151+
1085500056+	1060147+
1085500056+	1070148+
1000000051+	1080151+
1580500056+	1090147+
1580500056+	1100148+
1000000051+	1110151+
3900000056+	1120152+
1234000056+	1130255+
2000000056+	1140257+
2000000056+	1150000+
1000000051+	1160158+
1085500056+	1170255+
2000000056+	1180257+
2000000056+	1190000+
1000000051+	1200158+
1580500056+	1210255+
2000000056+	1220257+
2000000056+	1230000+
1000000051+	1240158+