Utah State University
DigitalCommons@USU

All Graduate Plan B and other Reports | Graduate Studies

5-1976

A Computer Approach to Performing MPPA Testing

James W. Shaw

Follow this and additional works at: https://digitalcommons.usu.edu/gradreports

Part of the Animal Sciences Commons

Recommended Citation

Shaw, James W., "A Computer Approach to Performing MPPA Testing" (1976). *All Graduate Plan B and other Reports*. 460.
https://digitalcommons.usu.edu/gradreports/460

This Report is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Plan B and other Reports by an authorized administrator of DigitalCommons@USU. For more information, please contact digitalcommons@usu.edu.

A COMPUTER APPROACH TO PERFORMING MPPA TESTING

by

James W. Shaw

A report submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Animal Science

Plan B

Approved:

UTAH STATE UNIVERSITY
Logan, Utah

1976

TABLE OF CONTENTS

	Page
LIST OF FIGURES	iii
ABSTRACT	iiii
INTRODUCTION	1
SECTION I	6
PERFORMANCE/WEAN	
SECTION II	35
PERFORMANCE/WEAN/TERML	
SECTION III	37
PERFORMANCE/MPPA/IMPPAL	
SECTION IV	50
PERFORMANCE/MPPA/LPMPPA	
SECTION V	60
PERFORMANCE/MPPA/MERGE	
SECTION VI	67
PERFORMANCE/MPPA/MPPA	
SECTION VII	80
PERFORMANCE/MPPA/CARDP AND SUMMARY	
APPENDIX	
SYSTEM DATA FLOW	
DATA COLLECTION FORM	
FILES USED	
SAMPLE PROGRAM RUN DECKS	

LIST OF FIGURES

Figure		Page
1.	PERFORMANCE/WEAN PROGRAM FLOW CHART	9
2.	CARD LAYOUT PERFORMANCE/WEAN	17
3.	PERFORMANCE/MPPA/IMPPAL PROGRAM FLOW CHART	40
4.	CARD LAYOUT PERFORMANCE/MPPA/IMPPAL	45
5.	PERFORMANCE/MPPA/LPMPPA PROGRAM FLOW CHART	52
6.	CARD LAYOUT PERFORMANCE/MPPA/LPMPPA	56
7.	PERFORMANCE/MPPA/MERGE PROGRAM FLOW CHART	62
8.	PERFORMANCE/MPPA/MPPA PROGRAM FLOW CHART	69
9.	CARD LAYOUT PERFORMANCE/MPPA/MPPA	71
10.	PERFORMANCE/MPPA/CARDP PROGRAM FLOW CHART	82
11.	CARD LAYOUT PERFORMANCE/MPPA/CARDP	84
12.	SYSTEM DATA FLOW (ABSTRACT)	88
13.	INITIAL HERD RUN (ABSTRACT)	89
14.	UPDATE A HERD (ABSTRACT)	90
15.	DATA COLLECTION FORM (ABSTRACT)	91

ABSTRACT

BY

James W. Shaw, Master of Science

Utah State University, 1976

Major Professor: Norris J. Stenquist

Department: Animal Science

In the highly competitive cattle market of today a scientific method, by which a rancher can base a judgment, moderated by experience as to which animals are to be kept, as proven by superior performance, can be a valuable asset.

This record performance testing system was designed to demonstrate the value of adequately kept records. Its purpose is to help ranchers develop records of herd performance, inventory, and improve management practices by more informed decisions. The system, through its representative, provides a means by which ranchers can obtain information on new ideas in animal production, health, sanitation, nutrition, management practices, and new record testing methods showing the results of their use and meaning.

The procedure used is based on the recommendations of the Beef Improvement Federation. The three test result summaries printed are (1) Weaning; (2) Yearling; and (3) Dam. An inventory of the herd is generated showing, in addition to performance, those not producing a calf, and those calves which did not have enough data for processing.

The test results will list different amounts of valuable information. According to each individual test, the results are:

(1) Weaning: The calf's weight both actual and adjusted, (two

hundred five (205) days of age), wean weight ratio, and weight per day of age at weaning. Also provided is a ranking of best to worst, by actual and adjusted weight.

(2) Yearling: Lists weight, actual and adjusted (one hundred sixty days between wean and year dates), weight gained between wean and year tests, average daily gain, and weight per day of age at yearling. Ranking of best to worst is of two kinds by actual yearling weight and by adjusted yearling weight.

(3) Dam: Number of calves in record, total wean weight accumulated, and previous Most Probable Producing Ability (MPPA) score. There is a ranking of best dam to worst dam, based on current MPPA scores.

All test results include herd averages for most valuable data (i.e.; weight, MPPA score, etc.).

This is but one type of a computer method of handling a cattle record system. The system was programmed and placed in operation, limited only by the physical size of the computer at the time of its programming. The system is in operation on the Burroughs 6700 computer system at the Utah State University campus.

This full system was accomplished by first modifying the calf performance testing and then programming the MPPA testing routines.

(101 pages)

INTRODUCTION

In the highly competitive cattle market of today a scientific method, by which a rancher can base a judgment, moderated by experience as to which animals are to be kept as proven by superior performance, can be a valuable asset.

This record performance testing system was designed to demonstrate the value of adequately kept records. Its purpose is to help ranchers develop records of herd performance, inventory, improve management practices by more informed decisions, and aid those unable to enroll in breed association testing. The system also provides the means by which a rancher can become familiar with performance testing results and their use before advancing to more complex and detailed records of genetic estimates as provided by TPR (Total Performance Records). Also it is felt that by having a representative of the system aid in getting the data (optional) and returning the finished results gives us the opportunity to present new ideas in animal production, animal health, sanitation, nutrition, and new record testing methods and showing the results of their use and meaning. There are many methods for either calf, dam, or sire evaluation. The method dealt with here is the Most Probable Producing Ability (MPPA) for dam evaluation.

The procedure used to obtain test results after collection of the data is (1) transferring data to a medium the computor can use, (2) calf results (either weaning or yearling), (3) dam results from which information was taken during calf performance testing, and (4) returning of test

results to the rancher. The formulas used to calculate the results are those recommended by the Beef Improvement Federation. A detailed description of each individual method will be found in the section dealing with the type of testing desired; the calf testing is handled by the wean programs and the dam by MPPA programs. The goal of this project was twofold: (1) add MPPA testing capability, and (2) improve efficiency of existing performance testing programs.

MPPA is designed to give a basis for comparing dams for which there is a difference in the number of calves produced and age of dam. It is most helpful in identifying the lowest producing cows for culling. A low lifetime average based on a single calf only is not an accurate presentation, as environmental conditions or the calf's genetic potential for growth might have been below the average of what the cow would normally produce. MPPA for weaning weight ratio is computed by the following formula:

$$\text{MPPA} = \bar{H} + \frac{N*R}{1+(N-1)R}(\bar{C}-\bar{H})$$

where $\bar{H}$ = 100, or the herd average wean weight ratio,

N = number of calves included in cows average,

R = .4, the repeatability factor for weaning weight ratio

$\bar{C}$ = average for weaning weight ratio for all calves the cow has produced

The post-weaning growth period (yearling records) is independent of maternal influences and provides no basis for calculating an MPPA rating on the dam; hence, weaning records are the basis. TPR uses 100 for the herd average. All the evaluation is printed both during weaning and yearling test of her calf, and the results show the complete test results

of all calves in her record. Our results of dam evaluation are processed only once, which is during weaning performance testing (herd average is computed) and the summation and printing of each individual calf in the record for that dam is changed to a one line summary of her performance (total calves produced and total wean weight ratio, and her previous year's MPPA score).

This is but a single approach to implementing this form of cow evaluation on a computer system. The system was designed to be dynamic to allow for changes in method of testing or expansion due to increases in information needed or volume of data. To lower storage cost only the programs are kept on disk storage whereas all herd data is maintained on cards and loaded to disk when needed for yearly processing. As disk files are no longer needed, the programs release the space back to the computer system. The programs used to get MPPA were written to be used separately so that those with small amounts of data or limited access computer accounts can get results from the evaluation dam performance.

There are a few items of information which will aid the reader in following the documentation of these computer programs.

Abbreviations

P stands for performance, the directory base name for the program series.

M stands for MPPA, the second level name given to all programs used to obtain the MPPA score.

W indicates WEAN which is the second level name used in referencing programs for weaning or yearling performance of calves.

I/O indicates input/output of all information.

All programs have at least a two (2) level name, with all MPPA having three (3) levels with the last name in a program title indicating the function of the program referenced.

In the individual section for a given program the following documents will be contained:

A. Brief description of job performed by program.

B. General flow chart of program logic.

C. Input/output (I/O) variable list.

D. Example of any input or output of the progran if any is card or line printer.

E. Program listing obtained during compilation.

Definitions

The following definitions will aid the reader in understanding the documentation:

Format. The method of describing to the computer the way data is to appear on input or output.

Types. The nature of the data.

A or C - alphanumeric, any combination of letters and numbers.

I - integer, arithmetric value which has an exponent of zero (i.e., 100,50, etc.)

F - real, arithmetric value which may or may not have an exponent (i.e., decimal values 100.3, 5.07, etc.)

Machine relative. This has been used to describe a form by which data is communicated into and out of the computer unedited into either letters or numbers.

File: A specified device where the computer either gets data from or sends data to.

Array or dimension. Set up an area of memory for the indicated variable (means this many word of storage in a continuous section).

2: Invalid punch on a system control card which appears in column one (1).

Program language. The text a program is written in for computer use.

ALGOL - Algorithmic languages.

FORTRAN - Formula translation.

The appendix will contain information which will be valuable in understanding the general flow of data in the system. It contains flow charts of data flow in the system from pickup to return of results, form for the data collection, list of all files used by the programs along with their specification, and an example of a job control deck for each program.

The program has each input line sequenced by the computer so in the event of run-time problems its trouble will be printed by the computer operation system and will contain this sequence number located on the right hand side of the program listing.

SECTION I

PERFORMANCE/WEAN

This program performs three (3) types of performance testing: (1) weaning, (2) yearling, and (3) set up of information to do dam performance.

The weaning record consists of adjusting the wean-weight to 205 day figure, calculate weight per day of age, and the weaning weight ratio. Also, there is an average figured for the weaning weight by sex group (cows, bulls, and steers).

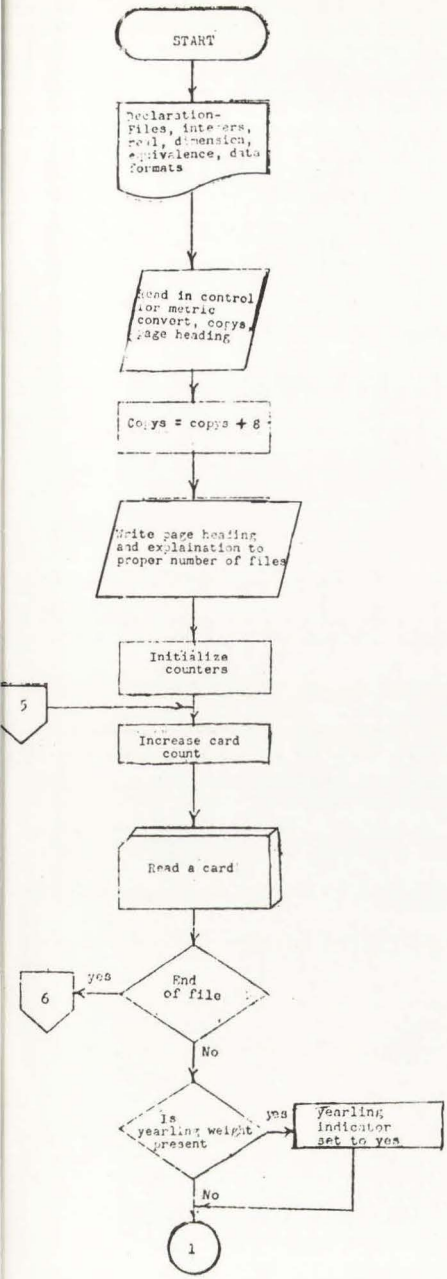
The yearling record consists of adjusting the yearling weight (365 days), finding weight gained between weaning and yearling, average daily gain, and weight per day of age at yearling. Averages are also listed for yearling weights by sex group.

Dam record is only created when weaning performance is done. It consists of a diskpack file containing information needed by the MPPA programs.

Input is by the card reader with the first input file containing the information for doing the testing. The second is a special notes file about any group of records that require explanation. The sire field is one of these as all input in this field is coded (two columns makes listing by number more practical than by name).

Output is to either/both printer or diskpack. Printer is a total of nine (9) files depending on number of copies requested (the system default file, plus eight (8) others; zero (0) defaults to one (1) copy). Headings on the pages are endemic to either weaning or yearling specification with the exemption of the heading entered by card to title output (rancher's name, date, etc.). Included are three pages in addition to those needed to print the results. These are: (1) averages both weaning

and yearling, with explanation on how to read printout results; (2) key to the heading legent; and (3) any special notes. Diskpack is written to when weaning performance is done only. It contains on record one (1) the number of records to follow, and the herd total wean weight ratio; the remaining records have just dam number with individual wean weight ratio.



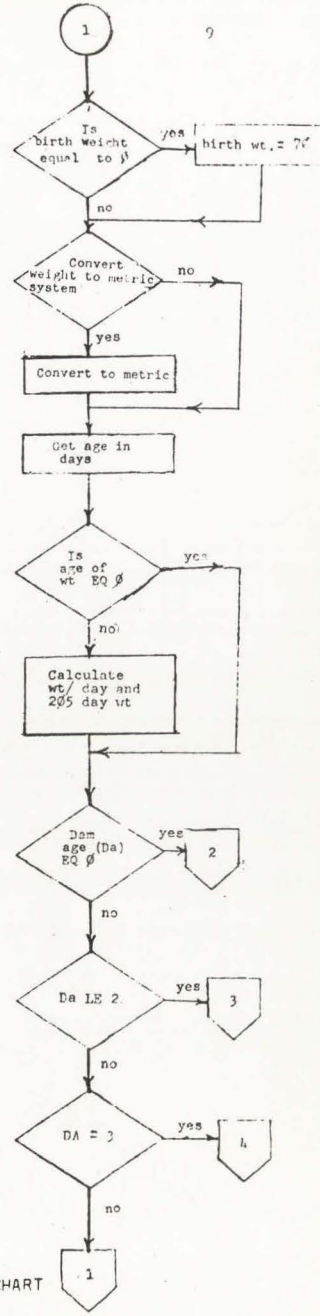
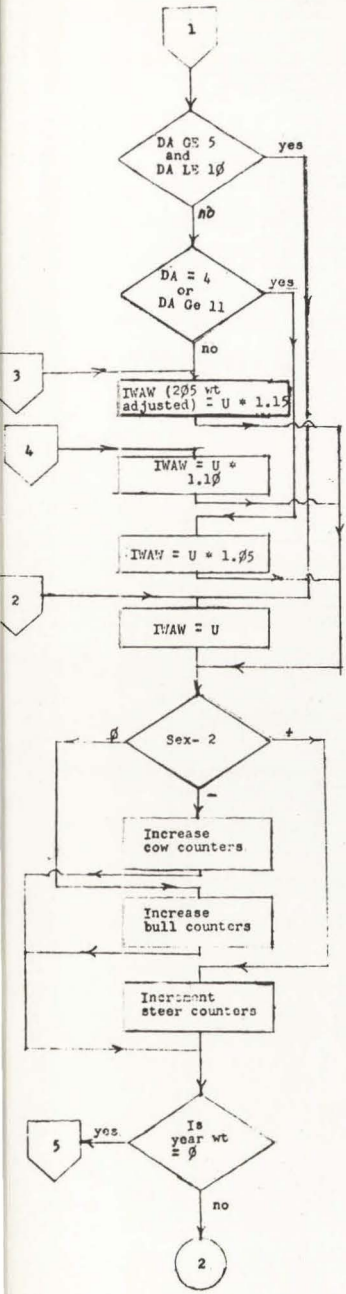
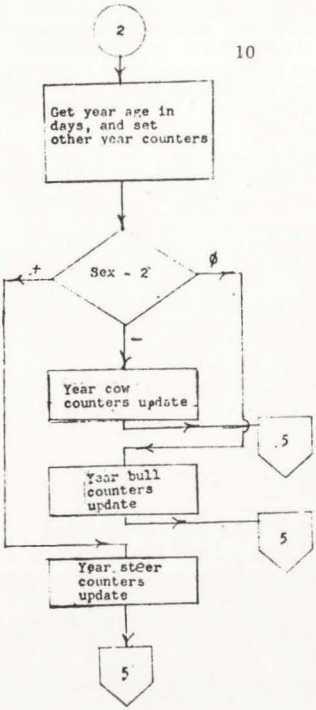
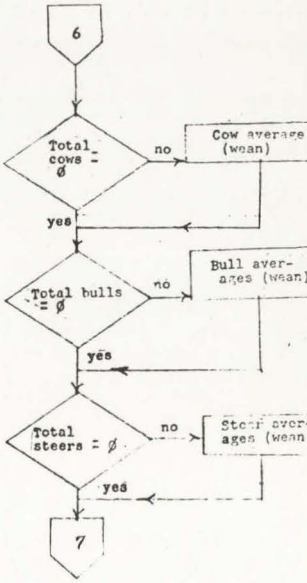
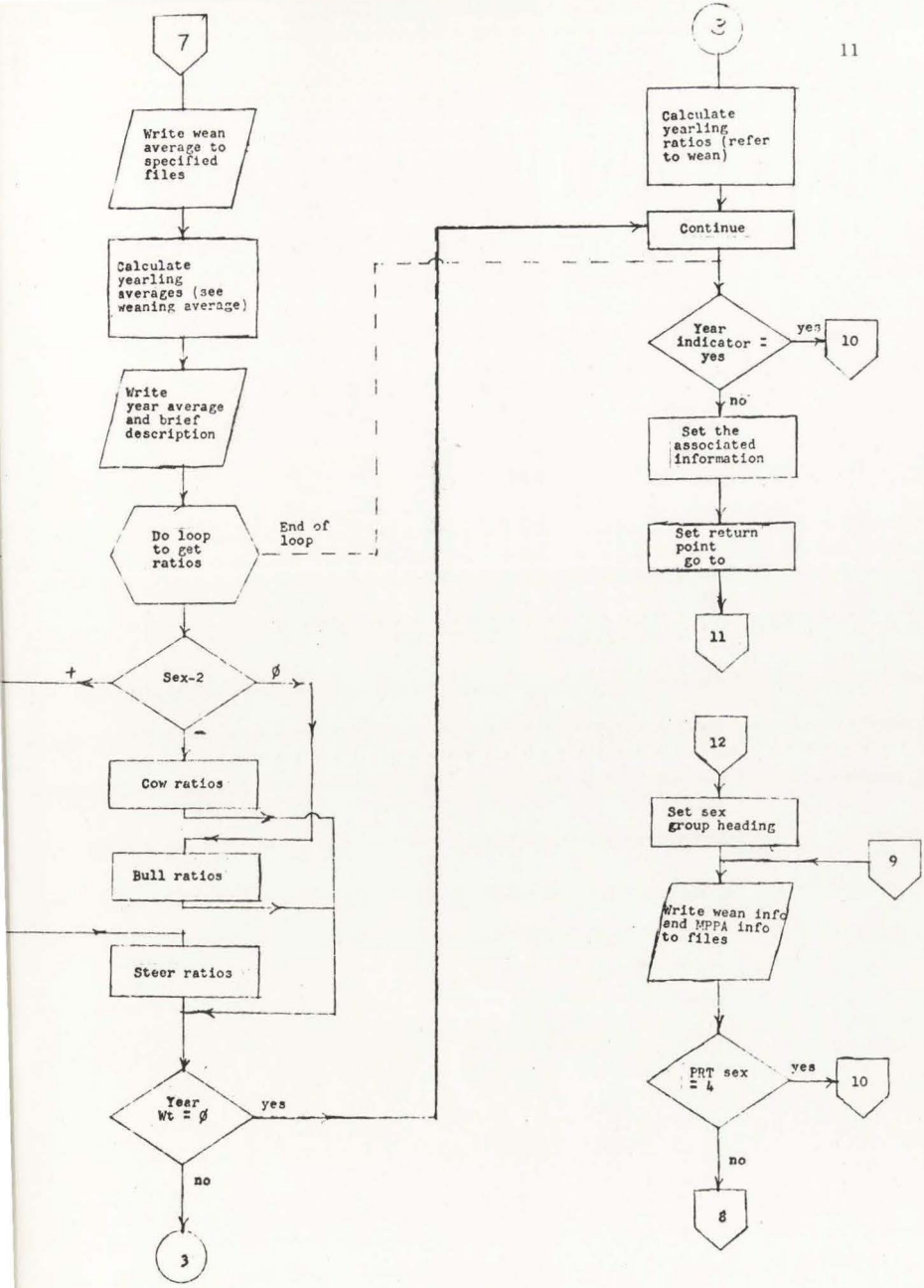


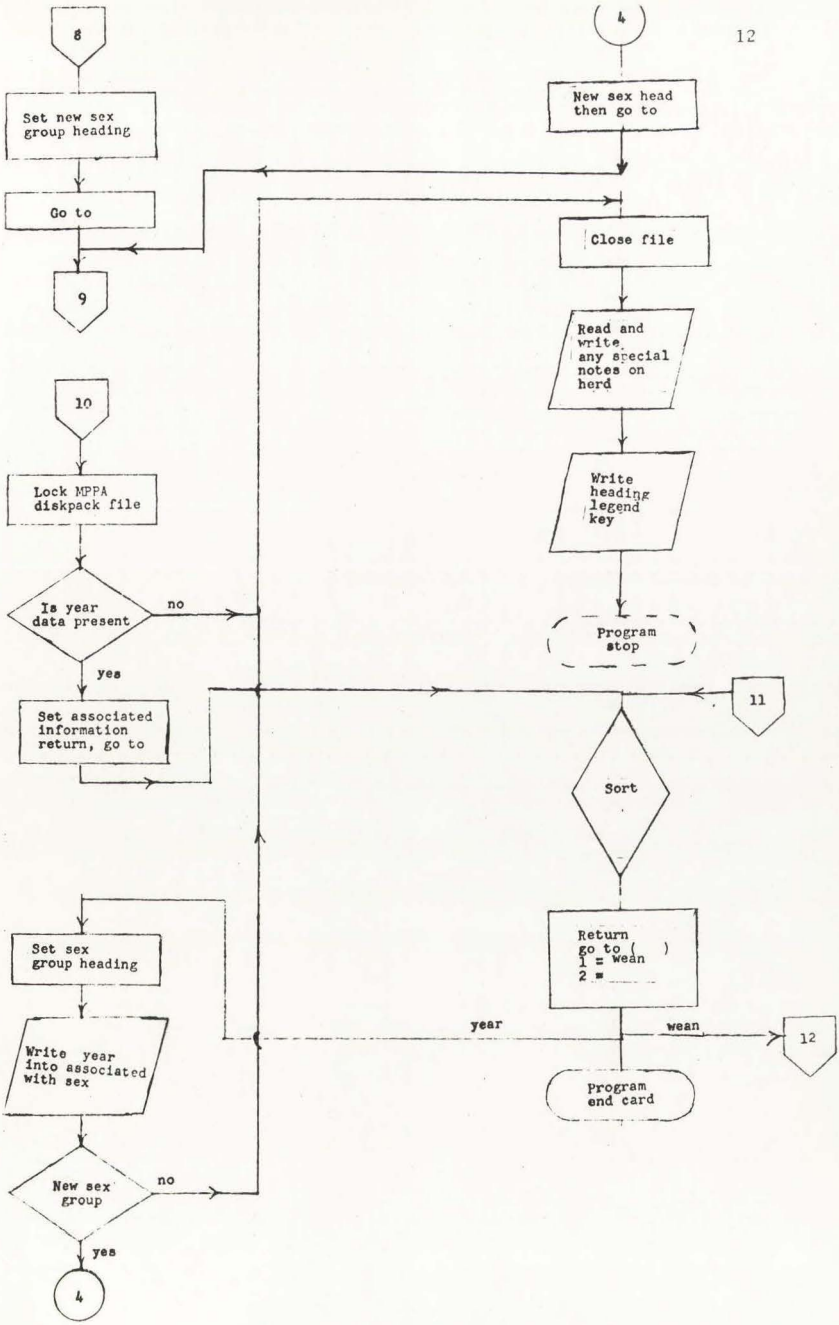
Figure 1. PERFORMANCE/MEAN PROGRAM FLOW CHART


1
DA GE 5 and DA LE 1Ø
yes
no
DA = 4 or DA Ge 11
yes
no
3
IWAW (2Ø5 wt adjusted) = U * 1.15
4
IWAW = U * 1.1Ø
IWAW = U * 1.Ø5
2
IWAW = U
Sex- 2
Ø
+
-
Increase cow counters
Increase bull counters
Increment steer counters
Is year wt = Ø
yes
5
no
2


2
Get year age in days, and set other year counters
Sex - 2
+
Ø
-
Year cow counters update
5
Year bull counters update
5
Year steer counters update
5


6
Total cows = Ø
no
Cow average (wean)
yes
Total bulls = Ø
no
Bull aver-ages (wean)
yes
Total steers = Ø
no
Steer aver-ages (wean
yes
7


7
Write wean average to specified files
Calculate yearling averages (see weaning average)
Write year average and brief description
Do loop to get ratios
End of loop
Sex-2
+
Ø
-
Cow ratios
Bull ratios
Steer ratios
Year Wt = Ø
yes
no
3
3
Calculate yearling ratios (refer to wean)
Continue
Year indicator = yes
yes
10
no
Set the associated information
Set return point go to
11
12
Set sex group heading
9
Write wean info and MPPA info to files
PRT sex = 4
yes
10
no
8


8
Set new sex group heading
Go to
9
10
Lock MPPA diskpack file
Is year data present
no
yes
Set associated information return, go to
Set sex group heading
Write year into associated with sex
New sex group
no
yes
4
4
New sex head then go to
Close file
Read and write any special notes on herd
Write heading legend key
Program stop
11
Sort
Return go to ( )
1 = wean
2 =
year
wean
12
Program end card

NAME VARIABLE	I/O STATUS	INFORMATION	FORMAT SPECIFICATION
ITAG	I/O	Calf tag number	A5
IBR	I/O	Calf breed	A2
IS	I/O	Calf sex	I1
IM (1,2,3)	I/O	Calf--Month[2]	I2
ID (1,2,3)	I/O	Calf--day[1]	I2
IYR (1,2,3)	I/O	Calf--year[1]	I2
IBWT	I/O	Calf birth weight	I3
IAWW	I/O	Actual wean weight	I3
IWS	I/O	Wean conformation score	I3
IDANO	I/O	Dam number	A6
IDA	I/O	Dam age	I2
ISIRE	I/O	Sire number[3]	A2
IP	I/O	Problems[3]	A2
IFYWT	I/O	Year actual weight	I4
NYS	I/O	Year conformation score	I3
YPROB	I/O	Year problems[3]	A2
CWAW	O	Herd wean average cow	F6.2
BWAW	O	Herd wean average bull	F6.2
SWAW	O	Herd wean average steer	F6.2
CYAW	O	HERD year average cow	F6.2

BYAW	O	Herd year average bull	F6.2
SYAW	O	Herd year average steer	F6.2
KM	I/O	Control metric conversion	I1
COPYS	I	Control number printout copys	I1
WORD	I/O	Page heading[4]	13A6
AWWRH	O	Herd wean weight ratio total	F10.0
NC	O	Number of records	I4
IWAWST	O	Adjusted wean weight	I3
IADW	O	Age in days at weaning	I3
IWW	O	Wean weight ratio	I3
ICR	O	Wean conformation ratio	I3
WPD	O	Weight per day	F6.2
QTAG	O	Associated calf tag	A5
QWW	O	Ranking wean weight ratio	I3
LTAG	O	Tag number	A5
LWW	O	Associated weaning weight	I3
IYAW	O	Adjusted year weight	I4
NADY	O	Age in days at year	I3
IYW	O	Year weight ration	I3
NCR	O	Year conformation ratio	I3

NDBW	0	Days between wean and year	I3
LGA	0	Weight gain between wean and year	I4
ADG	0	Average daily gain	F6.2
WDA	0	Weight per day of age yearling	F6.2
Sex	0	Sex group heading	7A6

1. Birth, wean or year; month, day, and year date.
2. Birth weight of zero (0) or blank will default to seventy (70).
3. Sire number is coded with key printed along with any special notes (indicated in problems column by *).
4. Heading placed in by first data card which contains both control characters along with the heading. The complete format for this card is 2I1,13A6.

COLUMN	INFORMATION
1 - 5	Calf tag number (A5)
6 - 7	Calf breed (A2)
8	Calf sex (I1) 1 = Cow 2 = Bull 3 Steer
9 - 14	Birthdate MMDDYY (3I2)
15 - 17	Birth weight (I3, blank - 70, date)
18 - 23	Wean date MMDDYY (3I2)
24 - 26	Wean weight (I3)
27 - 29	Conformation score (I3)
30 - 35	Dam number (A6, right justified)
36 - 37	Dam age (I2)
38 - 39	Sire code (A2)
40 - 41	Problems (A2)
42 - 47	Yearling data MMDDYY (3I2)
48 - 51	Yearling weight (I4)
52 - 54	Yearling conformation score (I3)
55 - 56	Yearling problems (A2)

The format for the heading card with metric conversion, number of copys, and title; is in the following order KM COPYS TITLE, with a one (1) indicating metric conversion desired, number of copies with nine (9) maximum, and title of report not to exceed seventy two (72) characters and spaces.

Birth weight is optional as the program will assume seventy (70) if the card is punched with zero (0) or left blank otherwise it will use the value punched.

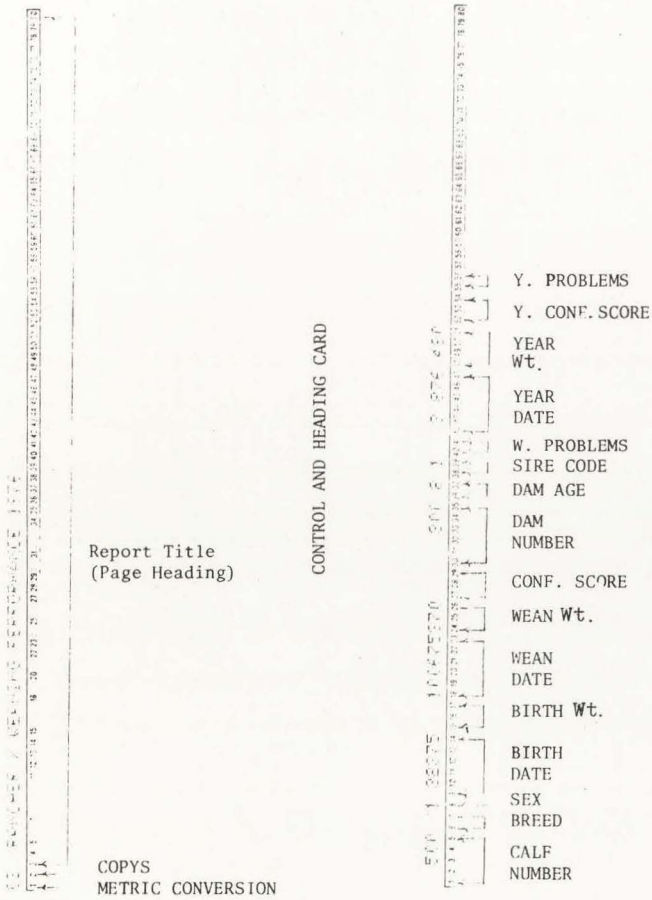


Figure 2. CARD LAYOUT PERFORMANCE/WEAN

FIELD AREA FOR NOTES

SPECIAL NOTES CARDS

OPANCHEK X EXAMPLE YEAR

CWAN	SWAN	SWAN
423.00	455.00	7.00
CYAN	SYAN	SYAN
0.00	0.00	0.00

THE MEANING OF EACH ABBREVIATION LISTED AT THE HEAD OF A COLUMN CAN BE FOUND AT THE END OF THIS LISTING. WHILE READING THE LISTING, A SINGLE ANIMAL IS REPRESENTED ON EACH LINE BUT ONLY AS FAR OVER AS THE "STAGE" COLUMN. THAT IS TO SAY THAT THE LAST FOUR COLUMNS ON THE RIGHT ARE INDEPENDENT FROM THE LINE TO THEIR LEFT. DO NOT FEAR OR ASSOCIATE THE LAST FOUR COLUMNS WITH THE OTHER INFORMATION OF THE SAME LINE. AS IS INDICATED ON THE ABBREVIATION PAGE "CWAN" EQUALS THE ADJUSTED WEANING WEIGHT RATIO (INDEX PERCENTAGE) AND IS LISTED ON THE RIGHT HAND SIDE OF THE PAGE FROM THE BEST CALF TO THE POOREST WITH ASSOCIATED TAG NUMBER IN THE PREVIOUS COLUMN. "SWAN" IS A LISTING OF ACTUAL WEANING WEIGHTS PUT IN DESENDING ORDER WITH ITS ASSOCIATED TAG NUMBER IN THE PRECEEDING COLUMN. THE LAST FOUR COLUMNS ARE FOR THE CONVENIENCE OF THE OWNER (RANCHER) PLACING THE ANIMALS IN A RANK FROM THE BEST TO THE POOREST. FOR MORE DETAILS ABOUT AN ANIMAL THAT IS LISTED IN THE RANK COLUMNS GO TO THE APPROPRIATE LINE IN THE LEFT SECTION. FOR THE CONVENIENCE OF STUDYING ANIMALS BY SEX GROUPS, THERE ARE FOUR MAJOR DIVISIONS: HEIFERS, BULLS, STEERS AND UNBORN CALVES. THE "UNBORN CALVES" LISTING IS TO COMPLETE AN INVENTORY OF ALL COWS. AN ASTERICK (*) IN THE PR (PROBLEMS) COLUMN IS AN INDICATION THAT ADDITIONAL NOTES FOR THAT ANIMAL CAN BE FOUND AT THE BACK OF THE LISTING UNDER "EXTRA NOTES". A BIRTHDATE OR AN ESTIMATED BIRTHDATE IS ABSOLUTELY ESSENTIAL FOR THIS PROGRAM TO OPERATE. THE SIRE COLUMN CAN BE DECODED BY GOING TO THE "EXTRA NOTES" SECTION LOCATED AT THE END OF THE LISTING.

ORANCHER X EXAMPLE YEAR

WEANLING HEIFERS

TAGNO	BRS	BIRTHD	BWT	WEANDA	AWW	IWS	DAMNO	DA	SIRE	WAW	ADW	IWW	ICR	WPD	PR	I	RANK BY AWW QTAG	QAW	RANK BY IWW LTAG	LWW
751	1	4 775	100	21676	590	0	171	4	1	438	316	103	0	1.55		1	752	660	753	110
752	1	4 875	105	21676	660	143	32	5	1	466	315	110	119	1.76		1	751	590	752	110
753	1	41575	80	21676	570	142	732	2	1	467	308	110	118	1.59		1	753	570	751	103
754	1	42575	85	21676	505	133	721	3	1	411	298	97	111	1.41		1	754	505	756	102
755	1	61375	84	21676	430	132	724	3	1	405	249	95	110	1.39		1	756	470	754	97
756	1	62175	95	21676	470	142	29	11	1	434	241	102	118	1.56		1	755	430	755	95
757	1	62975	82	21676	415	132	40	10	1	374	233	88	110	1.43		1	757	415	758	91
758	1	73075	100	21676	385	132	609	6	1	389	202	91	110	1.41		1	758	385	757	88

BRANCHER X EXAMPLE YEAR

										WEANLING		BULLS						I	RANK BY AWW		RANK BY IWW	
TAGNO	BR	S	BIRTHD	BWT	WEANDA	AWW	IWS	DAMNO	DA	SIRE	WAW	ADW	IWW	ICR	WPD	PR		I	QTAG	QWW	LTAG	LWW
175	2	4	975	110	21676	700	143	1207	4	1	519	314	111	103	1.88			I	175	700	175	111
275	2		41475	90	21676	650	142	0400	5	1	461	309	99	102	1.81			I	275	650	575	100
375	2		62475	90	21676	465	132	0395	5	1	413	230	88	95	1.58			I	475	490	475	99
475	2		71275	95	21676	490	133	82	5	2	463	220	99	96	1.80			I	375	465	275	99
575	2		9 275	85	21676	400	142	875	6	3	469	168	100	102	1.88			I	575	400	375	88

KEY TO SIRE CODE

1=272
2=272
3=372

```
HEADING LEGEND:
 CWA=COW (HEIFER) AVERAGE OF THE ADJUSTED WEANING WEIGHTS
 BWA=BULL AVERAGE OF THE ADJUSTED WEANING WEIGHTS
 SWA= STEER AVERAGE OF THE ADJUSTED WEANING WEIGHTS
 CYA=COW (HEIFER) AVERAGE OF THE ADJUSTED YEARLING WEIGHTS
 BYA=BULL AVERAGE OF THE ADJUSTED YEARLING WEIGHTS
 SYA=STEER AVERAGE OF THE ADJUSTED YEARLING WEIGHTS
 TAGNO=TAG NUMBER
 BR=BREED
      1=HEREFORD
      2=ANGUS
      3=CHAROLAIS
      4=SIMMENTAL
      5=HEREFORD X ANGUS
      6=HEREFORD X CHAROLAIS
      7=HEREFORD X ANGUS X CHAROLAIS
      8=LIMOUSIN X CHAROLAIS
      9=HEREFORD X SIMMENTAL
 S=SEX:
      1=COW 2=BULL 3=STEER
 BIRTHD=BIRTH DATE
 B.T=BIRTH WEIGHT
 WEANDT=WEANING DATE
 AWA=ACTUAL WEANING WEIGHT
 IAS=MEAN CONFORMATION SCORE
      1= 2=MIDDLE 3=
 DAMNO=DAM NUMBER
 DAGE=DAM AGE
 SIRENUM=SIRE NUMBER
 AWA=ADJUSTED WEAN WEIGHT
 ADWEAN IN DAYS AT WEANING
 IWA=WEAN WEIGHT RATIO
 ICSWEAN CONFORMATION RATIO
 WPDWEAN=WT PER DAY OF AGE AT WEANING
 PROBLEMS:
      =SPECIAL NOTE 1=SOLD 2=OPEN 3=CALF DIED 4=COW DIED 5=CALF SICK 6=COW SICK 7=ASSISSTANCE AT CALVING
 LTAG=ASSOCIATED TAG NUMBER
 LWA=RANKING OF WEANING WEIGHT RATIO
 STIDTAG NUMBER
 SWA=SAME

 TAGNO TAG NUMBER
 FYA=FINAL YEARLING WEIGHT
 YEARDT=YEARLING DATE WEIGHED
 IAY ADJUSTED YEARLING WEIGHT
 ADYEAR IN DAYS AT YEARLING WEIGHT
 IYW=YEARLING WEIGHT RATIO
 AYS=YEARLING CONFORMATION SCORE
      1= 2=MIDDLE 3=
 ICSYEARLING CONFORMATION RATIO
 DIFWDAY=DIFFERENCE WEANING AND YEARLING WEIGHTS
 DIFWPDAY=DIFFERENCE WEANING AND YEARLING WEIGHTS
 DAYBETWEIGHT DAYS BETWEEN WEANING AND YEARLING DATES
 ADG=AVERAGE DAILY GAIN
 WT/DAY=WEIGHT PER DAY OF AGE AT YEARLING
```

```
FILE  3(KIND=DISKPACK, MAXRECSIZE=3, BLOCKSIZE=90, SAVEFACTOR=2,          00000100  C
    * SECURITYTYPE=CLASSA, TITLE='MPPA1')                                 00000200  C
FILE  4=NOTES,UNIT=READER                                                 00000300  C
                                                                                     FIB IS 0007 LONG
FILE  5=PERFORMANCE/DATAIN                                                00000400  C
                                                                                     FIB IS 0009 LONG
FILE  10=FILE10,UNIT=PRINTER                                              00000500  C
                                                                                     FIB IS 0007 LONG
FILE  11=FILE11,UNIT=PRINTER                                              00000600  C
                                                                                     FIB IS 0007 LONG
FILE  12=FILE12,UNIT=PRINTER                                              00000700  C
                                                                                     FIB IS 0007 LONG
FILE  13=FILE13,UNIT=PRINTER                                              00000800  C
                                                                                     FIB IS 0007 LONG
FILE  14=FILE14,UNIT=PRINTER                                              00000900  C
                                                                                     FIB IS 0007 LONG
FILE  15=FILE15,UNIT=PRINTER                                              00001000  C
                                                                                     FIB IS 0007 LONG
FILE  16=FILE16,UNIT=PRINTER                                              00001100  C
                                                                                     FIB IS 0007 LONG
FILE  17=FILE17,UNIT=PRINTER                                              00001200  C
                                                                                     FIB IS 0007 LONG
                                                                                  START OF SEGMENT 002
                                                                              FORMAT SEGMENT IS 0006 LONG
                                                                              FORMAT SEGMENT IS 0104 LONG
                                                                              FORMAT SEGMENT IS 0127 LONG
      INTEGER BIRTHD,WEANDA,YEARDA,COPYS,OWW(1000),PRTSEX,OSEX(1000),     00001300  C  0021000010
     1 LISE(1000)                                                         00001400  C  0021000010
      REAL ITAG(1000),LTAG(1000),IOR(1000),IDAND(1000)                    00001500  C  0021000010
      REAL IP(1000),GTAG(1000),ISIPE(1000),YPPOW(1000)                    00001600  C  0021000010
      DIMENSION IGUM(11),NAME(12),IAGE(3),IME(1000,3),ID(1000,3),         00001700  C  0021000010
     1IYR(1000,3),IWA(STE(1000),          IS(1000),                       00001800  C  0021000010
     2IWW(1000),IAWW(1000),IWS(1000),IDA(1000),                           00001900  C  0021000010
     3IAGE(1000),IW(1000),ICR(1000),IFYWT(1000),IYAW(1000),WADY(1000),    00002000  C  0021000010
     4IYW(1000),NYS(1000),NCR(1000),NDHW(1000),                           00002100  C  0021000010
     5LGA(1000),ADG(1000),WD(1000),                                       00002200  C  0021000010
     6PPOW(1000),WPD(1000),LWW(1000),NUYWW(1000),SEX(7)                   00002300  C  0021000010
      DATA IGUM/31,59,90,120,151,181,212,243,273,304,334/,                00002400  C  0021000010
     1      NAME/ 1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12/                     00002500  C  0021000010
      EQUIVALENCE (IAGE(1),BIRTHD),(IAGE(2),WEANDA),(IAGE(3),YEARDA)      00002600  C  0021000010
  111 FORMAT(5X,A2,I1,2(3I2,I3),(5X,A6,I2,A2,A2,3I2,I4,I3,A2)             00002700  C  0021000010
  221 FORMAT(' ',38X,F4.6, 23X,'I  RANK BY AWW',4X,'RANK BY IWW')         00002800  C  0021000010
  222 FORMAT('0','TAGNO BR S BIRTHD BWT WEANDA AWW IWS DAMNO  DA SIRE',   00002900  C  0021000010
     2   ' WAW ADW IWW ICR    WPD  PR     I  GTAG    OWW      LTAG   LWW') 00003000  C  0021000010
  232 FORMAT('0',A5,X,A2,X,I1,X,2(3I2,X,I3,X),I3,X,A6,X,I2,3X,A2,X,       00003100  C  0021000010
     *(I3,X),F6.2,2X,A2,5X,'I',X,A5,3X,I3,4X,A5,3X,I3)                    00003200  C  0021000010
  233 FORMAT('0',A5,X,A2,X,I1,X,3I2,2(X,I4),                              00003300  C  0021000010
     1(X,I3),2X,3(X,I3),6X,I4,3X,F6.2,6X,F6.2,                            00003400  C  0021000010
     1          6X,A2,4X,'I ',A5,4X,I4,4X,A5,4X,I4)                       00003500  C  0021000010
  244 FORMAT('0','TAGNO BR S YEARDA FYWT IYAW ADY IYW   NYS NCR DDH',6X,  00003600  C  0021000010
     *'GAIN    ADG        WD/DAY     PR     I  QTAG     QYWT      LTAG',  00003700  C  0021000010
     *5X,'LTAW')                                                          00003800  C  0021000010
  300 FORMAT(' ',38X,F4.6,10X,' I RANK BY FYWT',6X,'RANK BY IYW')         00003900  C  0021000010
  430 FORMAT('0',38X,'  CWAW      DAW      SWAW')                         00004000  C  0021000010
  433 FORMAT('0',38X,F6.2,2X,F6.2,3X,F6.2)                                00004100  C  0021000010
  444 FORMAT('0',38X,'  CTAW      DTAW      STAW')                        00004200  C  0021000010
  445 FORMAT('0',38X,F6.2,3X,F6.2,3X,F6.2)                                00004300  C  0021000010
  900 FORMAT(13A6)                                                        00004400  C  0021000010
  901 FORMAT(27X,13A6)                                                    00004500  C  0021000010
```

```
932   FORMAT(I*)
470 FORMAT(1*,*            HEADING LEGEND:*,/,10X,                          00004600  C  002:000010
     *'CWA=COW(HEIFER) AVERAGE OF THE ADJUSTED WEANING WEIGHTS*,/,10X,      00004700  C  002:000010
     *'BWA=BULL AVERAGE OF THE ADJUSTED WEANING WEIGHTS*,/,10X,             00004800  C  002:000010
     *'SWA=STEER AVERAGE OF THE ADJUSTED WEANING WEIGHTS*,/,10X,            00004900  C  002:000010
     *'CYA=COW(HEIFER) AVERAGE OF THE ADJUSTED YEARLING WEIGHTS*,/,10X,     00005000  C  002:000010
     *'BYA=BULL AVERAGE OF THE ADJUSTED YEARLING WEIGHTS*,/,10X,            00005100  C  002:000010
     *'SYA=STEER AVERAGE OF THE ADJUSTED YEARLING WEIGHTS*,/,10X,           00005200  C  002:000010
     *'TAG=TAG NUMBER*,/,10X,                                               00005300  C  002:000010
     * 'B=BREED*,/,15X,*1=HEREFORD*,/,15X,                                  00005400  C  002:000010
     *'2=ANGUS*,/,15X,*3=CHAROLAIS*,/,15X,*4=SEMMENTAL*,/,15X,             00005500  C  002:000010
     *'5=HEREFORD X ANGUS*,/,15X,*6=HEREFORD X CHAROLAIS*,/,15X,           00005600  C  002:000010
     *'7=HEREFORD X ANGUS X CHAROLAIS*,/,15X,*8=LIMOUSIN X CHAROLAIS*,     00005700  C  002:000010
     */,15X,*9=HEREFORD X SIMMENTAL*,                                      00005800  C  002:000010
     */,10X,*S=SEX*,/,15X,*1=COW 2=BULL 3=STEER*,/,10X,                    00005900  C  002:000010
     *'BD=BIRTH DATE*,/,10X,*W=BIRTH WEIGHT*,/,10X,                        00006000  C  002:000010
     *'WD=WEANING DATE*,/,10X,                                             00006100  C  002:000010
     *'WA=ACTUAL WEANING WEIGHT*,/,10X,*CWS=WEAN CONFORMATION SCORE*,/,    00006200  C  002:000010
     *15X,*1=P 2=MIDDLE 3=*,/,10X,*DAMNO=DAM NUMBER*,/,10X,*DA=DAM AGE*    00006300  C  002:000010
     *,/,10X,*SIRE=SIRE NUMBER*,/,10X,*AW=ADJUSTED WEAN WEIGHT*,/,10X,     00006400  C  002:000010
     *'AD=AGE IN DAYS AT WEANING*,/,10X,*WR=WEAN WEIGHT RATIO*,/,10X,      00006500  C  002:000010
     *'CF=WEAN CONFORMATION RATIO*,/,10X,*WD=WEIGHT PER DAY OF AGE AT      00006600  C  002:000010
     *WEANING*,                                                            00006700  C  002:000010
     *,/,10X,*PR=PROBLEMS*,/,15X,*1=SEE SPECIAL NOTE 1=SGLD 2=OPEN 3=DAL   00006800  C  002:000010
     *F EXCD 4=COW DIED 5=CALF SICK 6=COW SICK 7=ASSISSTANCE AT CALVING*   00006900  C  002:000010
     *,/,10X,                                                              00007000  C  002:000010
     *'ATG=ASSOCIATED TAG NUMBER*,/,10X,*RW=RANKING OF WEANING WEIGHT      00007100  C  002:000010
     *RATIO*,/,10X,*DTAG=TAG NUMBER*,/,10X,*RW=RANK*,///,                  00007200  C  002:000010
     *'TAG=TAG NUMBER*,/,10X,*FYWT=ACTUAL YEARLING WEIGHT*,/,10X,          00007300  C  002:000010
     *'YRDA=YEARLING DATE WEIGHED*,/,10X,*YAW=ADJUSTED YEARLING WEIGH      00007400  C  002:000010
     *TT*,/,10X,*AD=AGE IN DAYS AT YEARLING WEIGHT*,/,10X,                 00007500  C  002:000010
     *'YRWR=YEARLING WEIGHT RATIO*,/,15X,*NS=YEARLING CONFORMATION SCORE   00007600  C  002:000010
     *'YRCF=YEARLING CONFORMATION RATIO                                    00007700  C  002:000010
     *'YRDA=YEARLING DATE WEIGHED*,/,10X,*DAYS=YEARLING WEIGHT             00007800  C  002:000010
     *'YRAD=AGE IN DAYS AT YEARLING WEIGHT*,/,10X,                         00007900  C  002:000010
     *'YRGD=DAILY GAIN BETWEEN WEANING AND YEARLING DATES*,/,10X,          00008000  C  002:000010
     *'ADG=AVERAGE DAILY GAIN*,/,10X,                                      00008100  C  002:000010
     *'WT/DAY=WEIGHT PER DAY OF AGE AT YEARLING*)                          00008200  C  002:000010
920   FORMAT(0*,*THE MEANING OF EACH ABBREVIATION LISTED AT THE HEAD OF     00008300  C  002:000010
     1A COLUMN CAN BE FOUND AT THE END OF THIS LISTING. WHILE READING*,/   00008400  C  002:000010
     2*0*,*NOTE,                                                           00008500  C  002:000010
     3' LISTING: A SINGLE ANIMAL IS REPRESENTED ON EACH LINE BUT ONLY*,    00008600  C  002:000010
     4' AS FAR,                                                            00008700  C  002:000010
     5  * GOES AS THE "DTAG" COLUMN. THAT IS TO SAY THAT THE LAST*,/        00008800  C  002:000010
     6*0* "FOUR COLUMNS "*,                                                00008900  C  002:000010
     7      *ON THE RIGHT ARE INDEPENDENT FROM THE LINE TO THEIR LEFT. D   00009000  C  002:000010
     8E NOT READ OR ASSOCIATE THE LAST FOUR COLUMNS WITH THE OTHER*,/*0*   00009100  C  002:000010
     9 *INFORMATION *,                                                     00009200  C  002:000010
     9      *ON THE SAME LINE. AS IS INDICATED ON THE ABBREVIATION PAGE    00009300  C  002:000010
     9"LWR" EQUALS THE ADJUSTED WEANING WEIGHT RATIO*,/*0*,                00009400  C  002:000010
     9 *(INDEX PERCENTAGE) *,                                              00009500  C  002:000010
     9  *AND IS LISTED ON THE RIGHT HAND SIDE OF THE PAGE FROM THE REST    00009600  C  002:000010
     9CALF TO THE POOREST WITH ASSOCIATED TAG NUMBER*,/*0*,                00009700  C  002:000010
     9 *IN THE PREVIOUS COLUMN. *,                                         00009800  C  002:000010
     9     *"GWA" IS A LISTING OF ACTUAL WEANING WEIGHTS PUT IN DESENDING  00009900  C  002:000010
     9ORDER WITH ITS ASSOCIATED TAG*,/*0*,                                 00010000  C  002:000010
     9 *NUMBER IN THE PRECEEDING COLUMN. THE *,                            00010100  C  002:000010
     9*LAST FOUR COLUMNS ARE FOR THE CONVENIENCE OF THE OWNER (RANCHER)    00010200  C  002:000010
     9PLACING THE ANIMALS IN A RANK*,/*0*,*FROM *,                         00010300  C  002:000010
     9*THE BEST TO THE POOREST. FOR MORE DETAILS ABOUT AN ANIMAL THAT IS   00010400  C  002:000010
     9 LISTED IN THE RANK COLUMNS GO TO THE APPROPRIATE LINE*,/*0*,        00010500  C  002:000010
                                                                           00010600  C  002:000010
```

```
     9*IN THE LEFT SECTION. FOR THE CONVENIENCE OF STUDYING ANIMALS BY S 00010700  C  002:0000:0
     9EX GROUPS, THERE ARE FOUR MAJOR DIVISIONS:*,/,*0*,                 00010800  C  002:0000:0
     9*HEIFERS, BULLS, STEERS AND UNBORN CALVES. THE "UNBORN CALVES" LIS 00010900  C  002:0000:0
     9TING IS TO COMPLETE AN INVENTORY OF ALL*,/,*0*,                    00011000  C  002:0000:0
     9*COWS. AN ASTERISK (*) IN THE PR (PROBLEMS) COLUMN IS AN INDICATIO 00011100  C  002:0000:0
     9N THAT ADDITIONAL NOTES FOR THAT ANIMAL*,/,*0*,                    00011200  C  002:0000:0
     9    *CAN BE FOUND AT THE BACK OF THE LISTING UNDER "EXTRA NOTES".  00011300  C  002:0000:0
     9A BIRTHDATE OR AN ESTIMATED BIRTHDATE IS ABSOLUTELY ESSENTIAL FOR* 00011400  C  002:0000:0
     9,/,*0*,*THIS*,                                                     00011500  C  002:0000:0
     9    * PROGRAM TO OPERATE. THE SIRE COLUMN CAN BE DECODED BY GOING T 00011600  C  002:0000:0
     9O THE "EXTRA NOTES" SECTION LOCATED AT THE END OF THE LISTING.*)   00011700  C  002:0000:0
 1332 FORMAT(2I1,13A6)                                                   00011800  C  002:0000:0
 1333 FORMAT(1*,I1,13A6)                                                 00011900  C  002:0000:0
 1400 FORMAT(A6,I3)                                                      00012000  C  002:0000:0
 1401 FORMAT(3F10.0)                                                     00012100  C  002:0000:0
      READ(5,1332) KK, COPYS, WORD                                       00012200  C  002:0000:0
      WRITE (6,1333)KK,    WORD                                          00012300  C  002:0008:2
                                                                                       FEB IS 0006 LONG
      COPYS=COPYS*8                                                      00012400  C  002:0014:2
      IF(COPYS .LE. 9)GO TO 70                                           00012500  C  002:0015:5
      DO 71 I=10,COPYS                                                   00012600  C  002:0016:5
71    WRITE(6,1333)KK,    WORD                                           00012700  C  002:0018:0
C     INITIALIZE COUNTERS                                                00012800  C  002:0020:3
70    NCOPS=0                                                            00012900  C  002:0022:3
      A**SH = 0                                                          00013000  C  002:0024:1
      ICOW=0                                                             00013100  C  002:0026:5
      IBULL=0                                                            00013200  C  002:0027:5
      ISTEER=0                                                           00013300  C  002:0029:1
      ITC=0                                                              00013400  C  002:002C:5
      ITB=0                                                              00013500  C  002:002E:3
      ITBN=0                                                             00013600  C  002:002F:1
      IBC=0                                                              00013700  C  002:002F:5
      IBB=0                                                              00013800  C  002:0032:3
      ITS=0                                                              00013900  C  002:0031:1
      ITC=0                                                              00014000  C  002:0031:5
      ITB=0                                                              00014100  C  002:0032:3
      ITS=0                                                              00014200  C  002:0033:1
      NCC=0                                                              00014300  C  002:0033:5
      NBULL=0                                                            00014400  C  002:0034:3
      NSTEER=0                                                           00014500  C  002:0035:1
      NTC=0                                                              00014600  C  002:0035:5
      NTB=0                                                              00014700  C  002:0036:3
      NTS=0                                                              00014800  C  002:0037:1
      KCLG=0                                                             00014900  C  002:0037:5
      KBLG=0                                                             00015000  C  002:0038:3
      KSLG=0                                                             00015100  C  002:0039:1
      KCLGA=0                                                            00015200  C  002:0039:5
      KBLGA=0                                                            00015300  C  002:003A:3
      KSLGA=0                                                            00015400  C  002:003B:1
      TEARLG="NO"                                                        00015500  C  002:003B:5
      I=A**2                                                             00015600  C  002:003D:3
C     READ IN DATA                                                       00015700  C  002:003E:1
    5 NCARD=NCARD+1                                                      00015800  C  002:003E:1
      READ(5,111,END=6) ITAG(NCARD), IBR(NCARD),IS(NCARD),               00015900  C  002:003F:3
     1IM(NCARD,1),ID(NCARD,1),ITR(NCARD,1),IBWT(NCARD),IMENCARD,2),      00016000  C  002:0040:0
     1ID(NCARD,2),ITR(NCARD,2),IAWW(NCARD),IWS(NCARD),IDAND(NCARD),      00016100  C  002:0045:2
     *ID4(NCARD),ISIRE(NCARD),IP(NCARD),IM(NCARD,3),ID(NCARD,3),         00016200  C  002:0046:0
     1ITR(NCARD,3),IFY4(NCARD),NYS(NCARD),YPROB(NCARD)                   00016300  C  002:0079:4
C     CHECK FOR YEARLING DATA IF PRESENT                                 00016400  C  002:0089:0
      IF(IFY4(NCARD) .NE. 0) YEARLG="YES"                                00016500  C  002:008A:0
C     USE ONLY IN CONVERTING FROM ENGLISH TO METRIC SYSTEM               00016600  C  002:008C:3
```

```
      IF( IBWT(NCARD) .EQ. 0 ) IBWT(NCARD) = 70                         00016700  C  002:008C43
      IF(M.LT.1)GO TO 95                                                00016800  C  002:009D40
      BWT=(FLOAT(IBWT(NCARD))*0.4536)                                   00016900  C  002:009121
      IBWT(NCARD)=BWT                                                   00017000  C  002:009434
      FWA=(FLOAT(IFWA(NCARD))*0.4536)                                   00017100  C  002:009553
      IFWA(NCARD)=FWA                                                   00017200  C  002:009924
      FTW=(FLOAT(IFTW(NCARD))*0.4536)                                   00017300  C  002:009B43
      IFTW(NCARD)=FTW                                                   00017400  C  002:009E44
C     CONVERT DATES INTO NUMBER OF DAYS                                 00017500  C  002:00A043
   78 DO 11 NTIME=1,3                                                   00017600  C  002:00A345
      ILEAP=0                                                           00017700  C  002:00A240
      IAGE(NTIME)=0                                                     00017800  C  002:00A244
      ALEAP=FLOAT(IYR(NCARD,NTIME))/4.                                  00017900  C  002:00A631
      ELEAP=IYR(NCARD,NTIME)/4                                          00018000  C  002:00A734
      IF(ALEAP.EQ.ELEAP)ILEAP=1                                         00018100  C  002:00AB31
      DO 10 I=1,12                                                      00018200  C  002:00AD31
      IF(IMON(NCARD,NTIME).NE.NAME(I))GO TO 10                          00018300  C  002:00AE40
      IF(I.EQ.1)GO TO 8                                                 00018400  C  002:00B231
      IF(I.GE.3.AND.ILEAP.EQ.1)IAGE(NTIME)=1                            00018500  C  002:00B532
      IAGE(NTIME)=ICUM(I-1)+ID(NCARD,NTIME)+IAGE(NTIME)                 00018600  C  002:00B710
      GO TO 11                                                          00018700  C  002:00BC15
    8 IAGE(NTIME)=ID(NCARD,NTIME)                                       00018800  C  002:00BD12
      GO TO 11                                                          00018900  C  002:00C131
   10 CONTINUE                                                          00019000  C  002:00C134
   11 CONTINUE                                                          00019100  C  002:00C345
C     CALCULATE IN DAYS IF BIRTH YEAR & MEAN YEAR DIFFERENT             00019200  C  002:00C610
      IF(IYR(NCARD,1).EQ.IYR(NCARD,2)) GO TO 1                          00019300  C  002:00C610
      ILEAP=MOD(IYR(NCARD,1),4)                                         00019400  C  002:00C920
      IF(ILEAP.NE.0) GO TO 30                                           00019500  C  002:00D831
      IAD(NCARD)= (367-BIRTHD) + MEANDA                                 00019600  C  002:00DE10
      GO TO 2                                                           00019700  C  002:00DF42
   30 IAD(NCARD)= (366-BIRTHD) + MEANDA                                 00019800  C  002:00E135
      GO TO 2                                                           00019900  C  002:00E531
    1 IAD(NCARD)=MEANDA-BIRTHD                                          00020000  C  002:00E534
    2 IF(IAD(NCARD) .LT. 0) IAD(NCARD)=0                                00020100  C  002:00E532
      WPI(NCARD)=0.0                                                    00020200  C  002:00E924
      W=0.0                                                             00020300  C  002:00EB31
      IF(IAW(NCARD) .EQ. 0) GO TO 708                                   00020400  C  002:00EB35
      IF(IAD(NCARD) .EQ. 0) GO TO 708                                   00020500  C  002:00ED34
      WPI(NCARD)=FLOAT(IAW(NCARD)-IBWT(NCARD))/FLOAT(IADW(NCARD))       00020600  C  002:00EF35
      W=(FLOAT(IAW(NCARD)-IBWT(NCARD))/FLOAT(IADW(NCARD)))*205.*(FLOAT  00020700  C  002:00F431
     1(IBWT(NCARD)))                                                    00020800  C
C     BREAK BY DAM AGE TO GET PROPER FACTOR IN ARRIVING AT ADJUSTED     00020900  C  002:00F934
C     WEANING WEIGHT                                                    00021000  C  002:00F934
708   IF (IDA(NCARD) .EQ. 0) GO TO 15                                   00021100  C  002:00F934
      IF(IDA(NCARD).LE.2)GO TO 12                                       00021200  C  002:00FB13
      IF(IDA(NCARD).EQ.3)GO TO 13                                       00021300  C  002:00FD35
      IF(IDA(NCARD).EQ.4)GO TO 14                                       00021400  C  002:00FF33
      IF(IDA(NCARD).GE.5.AND.IDA(NCARD).LE.10)GO TO 15                  00021500  C  002:00F135
      IF(IDA(NCARD).GE.11)GO TO 14                                      00021600  C  002:00F531
   12 IWAW=W*1.15                                                       00021700  C  002:00F731
      GO TO 16                                                          00021800  C  002:00F935
   13 IWAW=W*1.10                                                       00021900  C  002:00FA32
      GO TO 16                                                          00022000  C  002:00FC35
   14 IWAW=W*1.05                                                       00022100  C  002:00FD32
      GO TO 16                                                          00022200  C  002:00FF35
   15 IWAW=W                                                            00022300  C  002:010032
C     BREAK BY CALF SEX TO GET INFORMATION FOR AVERAGE BY SEX GROUP FOR 00022400  C  002:010132
C     ADJUSTED WEAN WEIGHT AND CONFORMATION SCORE                       00022500  C  002:010132
   16 IF(IS(NCARD)-2)21,22,23                                           00022600  C  002:010132
21    IF(IWAW .NE. 0)ITC=ITC+1                                          00022700  C  002:010532
```

```
      IWAWST(NCARD)=IWAW                                                00022800  C  002:0107:5
      ICCW=ICCW+IWAW                                                    00022900  C  002:0109:3
      ICC=ICC+IWS(NCARD)                                                00023000  C  002:0108:0
      GO TO 35                                                          00023100  C  002:0100:1
22    IF(IWAW .NE. 0)ITB=ITB+1                                          00023200  C  002:0100:4
      IWAWT(NCARD)=IWAW                                                 00023300  C  002:0110:1
      IBULL=IBULL+IWAW                                                  00023400  C  002:0111:5
      ICB=ICB+IWS(NCARD)                                                00023500  C  002:0111:2
      GO TO 35                                                          00023600  C  002:0115:3
23    IF(IWAW .NE. 0)ITS=ITS+1                                          00023700  C  002:0116:0
      IWAWST(NCARD)=IWAW                                                00023800  C  002:0118:3
      ISTEER=ISTEER+IWAW                                                00023900  C  002:0114:1
      ICS=ICS+IWS(NCARD)                                                00024000  C  002:0113:4
C     RETURN FOR MORE DATA IF YEARLING WT=0 ELSE FIGURE AGE FOR YEAR IN 00024100  C  002:0110:5
C     DATS THEN CALCULATE SOME OF THE YEARLING DATA                     00024200  C  002:0110:5
   35 IF(IFYW(NCARD).EQ.0)GO TO 5                                       00024300  C  002:0110:5
      ILEAP=MOD(IYR(NCARD+1),4)                                         00024400  C  002:0111:4
      IF(ILEAP.NE.0) GO TO 60                                           00024500  C  002:0121:5
      NAD(NCARD)= (367-BIRTHD) + YEARDA                                 00024600  C  002:0122:4
      GO TO 61                                                          00024700  C  002:0126:1
   60 NAD(NCARD)=(366-BIRTHD)+YEARDA                                    00024800  C  002:0126:4
   61 NDW=(NCARD)=NAD(NCARD)-IADW(NCARD)                                00024900  C  002:0124:1
      WADG(NCARD)=FLOAT(IFYW(NCARD)-IDWT(NCARD))/FLOAT(NADY(NCARD))     00025000  C  002:0129:5
      LGA(NCARD)=IFYW(NCARD)-IAWW(NCARD)                                00025100  C  002:0132:3
      ACG(NCARD)=FLOAT(LGA(NCARD))/FLOAT(NDBW(NCARD))                   00025200  C  002:0136:1
      Y=ACG(NCARD)*165+IWAWST(NCARD)                                    00025300  C  002:0139:4
      IY=Y+.5                                                           00025400  C  002:0130:5
C     SPLIT BY SEX TO GET YEARLING TOTALS                               00025500  C  002:013E:5
      IF(IS(NCARD)-2)5G1,5G2,5G3                                        00025600  C  002:013E:5
  501 IYC=IYC+1                                                         00025700  C  002:0142:5
      IYW(NCARD)=IY                                                     00025800  C  002:0144:1
      NCG=NCG+IY                                                        00025900  C  002:0145:5
      NCGW=NCGW+WS(NCARD)                                               00026000  C  002:0147:2
      KGLG=KGLG+LGA(NCARD)                                              00026100  C  002:0149:3
      GO TO 5                                                           00026200  C  002:014H:4
  502 ITB=ITB+1                                                         00026300  C  002:014C:1
      IYW(NCARD)=IY                                                     00026400  C  002:014D:3
      NGULL=NGULL+IY                                                    00026500  C  002:014F:1
      NGBWT=NGBWT+WS(NCARD)                                             00026600  C  002:0150:4
      KGLG=KGLG+LGA(NCARD)                                              00026700  C  002:0152:5
      GO TO 5                                                           00026800  C  002:0155:0
  503 ITS=ITS+1                                                         00026900  C  002:0155:3
      IYW(NCARD)=IY                                                     00027000  C  002:0156:5
      NSTEER=NSTEER+IY                                                  00027100  C  002:0158:3
      NCS=NCS+WS(NCARD)                                                 00027200  C  002:0154:0
      KSLG=KSLG+LGA(NCARD)                                              00027300  C  002:0150:1
      GO TO 5                                                           00027400  C  002:0156:2
C     CALCULATE AVERAGE FOR AVERAGE WEAN ADJUSTED WEIGHT AND WEAN       00027500  C  002:0156:5
C     CALCULATION SCORES IN SEX GROUPS                                  00027600  C  002:0156:5
    6 IF(ITC.EQ.0)GO TO 41                                              00027700  C  002:0156:5
      SWAW=FLOAT(ICCW)/FLOAT(ITC)                                       00027800  C  002:0160:0
      SWAG=FLOAT(ICC)/FLOAT(ITC)                                        00027900  C  002:0161:2
   41 IF(ITB.EQ.0)GO TO 42                                              00028000  C  002:0162:4
      SWAW=FLOAT(ICBW)/FLOAT(ITB)                                       00028100  C  002:0163:5
      SWAG=FLOAT(ICB)/FLOAT(ITB)                                        00028200  C  002:0164:1
   42 IF(ITS.EQ.0)GO TO 45                                              00028300  C  002:0166:3
      SWAW=FLOAT(ISTEER)/FLOAT(ITS)                                     00028400  C  002:0167:4
      SWAG=FLOAT(ICS)/FLOAT(ITS)                                        00028500  C  002:0169:0
C     WRITE OUT WEANING AVERAGES                                        00028600  C  002:0164:2
   45 WRITE(6,4320)                                                     00028700  C  002:0164:2
      WRITE(6,4330)SWAW,SWAG                                            00028800  C  002:015E:2
```

```
      IF(COPYS .LE. 9)GO TO 72                                          00028900  C  002:0178:2
      DO 73 I=10,COPYS                                                  00029000  C  002:0179:4
      WRITE(I,432)                                                      00029100  C  002:0178:0
73    WRITE(I,433)CWAW,BWAW,SWAW                                        00029200  C  002:0187:2
C     CALCULATE AVERAGE FOR YEAR ADJUSTED WT AVERAGE, YEAR CONFORMATION 00029300  C  002:0198:3
C     SCORE, AND AVERAGE WT GAIN; BY SEX GROUP                          00029400  C  002:0198:3
72    IF(IYC.EQ.0)GO TO 51                                              00029500  C  002:0199:3
      CYAW=FLOAT(NCCW)/FLOAT(IYC)                                       00029600  C  002:019C:4
      CAYS=FLOAT(NCC)/FLOAT(IYC)                                        00029700  C  002:019E:0
      KBLGA=KBLG/IYC                                                    00029800  C  002:019F:2
   51 IF(IYB.EQ.0)GO TO 52                                              00029900  C  002:01A0:5
      BYAW=FLOAT(NBULL)/FLOAT(IYB)                                      00030000  C  002:01A2:0
      BAYS=FLOAT(NCB)/FLOAT(IYB)                                        00030100  C  002:01A3:2
      KBLGA=KBLG/IYB                                                    00030200  C  002:01A4:4
   52 IF(IYS.EQ.0)GO TO 43                                              00030300  C  002:01A6:1
      SYAW=FLOAT(NSTEER)/FLOAT(IYS)                                     00030400  C  002:01A7:2
      SAYS=FLOAT(NCS)/FLOAT(IYS)                                        00030500  C  002:01A9:4
      KBLGA=KBLG/IYS                                                    00030600  C  002:01AA:0
   43 WRITE(6,444)                                                      00030700  C  002:01AD:3
      WRITE(6,445)CYAW,BYAW,SYAW                                        00030800  C  002:01AF:2
      IF(YEARLG .IS. 'NO') WRITE(6,920)                                 00030900  C  002:01B1:2
      IF(COPYS .LE. 9)GO TO 74                                          00031000  C  002:01BF:2
      DO 75 I=10,COPYS                                                  00031100  C  002:01C0:4
      WRITE(I,444)                                                      00031200  C  002:01C2:0
      WRITE(I,445)CYAW,BYAW,SYAW                                        00031300  C  002:01CE:2
75    IF(YEARLG .IS. 'NO') WRITE(I,920)                                 00031400  C  002:01D0:2
74    NC=NCARD-1                                                        00031500  C  002:01F0:3
      DO 700 KK=1,NC                                                    00031600  C  002:01F1:5
C     SPLIT BY SEX TO GET WEANING WEIGHT RATIO,AND WEANING CONFORMATION 00031700  C  002:01F3:0
C     RATIO                                                             00031800  C  002:01F3:0
      IF(IS(KK)-23)701,702,703                                          00031900  C  002:01F3:0
  701 CWW=FLOAT(IWWS(KK))/CWAW                                          00032000  C  002:01F7:0
      IWW(KK)=CWW*100.                                                  00032100  C  002:01F9:0
      AWWPH = AWWHH + FLOAT(IWW(KK))                                    00032200  C  002:01FB:2
      SCR= 0.0                                                          00032300  C  002:01FD:2
      IF (CWAS .NE. 0.0)SCR=FLOAT(IWS(KK))/CAWS                         00032400  C  002:01FE:0
      ICR(KK)=SCR*100.                                                  00032500  C  002:0201:1
      GO TO 700                                                         00032600  C  002:0203:3
  702 BWW=FLOAT(IWWS(KK))/BWAW                                          00032700  C  002:0204:0
      IWW(KK)=BWW*100.                                                  00032800  C  002:0206:0
      AWWHH = AWWHH + FLOAT(IWW(KK))                                    00032900  C  002:0208:2
      SCR= 0.0                                                          00033000  C  002:020A:2
      IF (BAWS .NE. 0.0) SCR=FLOAT(IWS(KK))/BAWS                        00033100  C  002:020B:0
      ICR(KK)=SCR*100.                                                  00033200  C  002:020E:1
      GO TO 700                                                         00033300  C  002:0210:3
  703 SWW=FLOAT(IWWS(KK))/SWAW                                          00033400  C  002:0211:0
      IWW(KK)=SWW*100.                                                  00033500  C  002:0213:0
      AWWPH = AWWPH + FLOAT(IWW(KK))                                    00033600  C  002:0215:2
      SCR=0.0                                                           00033700  C  002:0217:2
      IF(SAWS .NE. 0.0) SCR=FLOAT(IWS(KK))/SAWS                         00033800  C  002:0218:0
      ICR(KK)=SCR*100.                                                  00033900  C  002:021B:1
C     CHECK FOR YEARLING WEIGHT SO AS TO CALCULATE YEAR CONFORMATION    00034000  C  002:021D:5
C     RATIO OF WGT                                                      00034100  C  002:021D:5
  700 IF(IY(KK).EQ.0)GO TO 700                                          00034200  C  002:021E:5
      IF(IS(KK)-23)711,712,713                                          00034300  C  002:0221:2
  711 CYW=FLOAT(IYWS(KK))/CYAW                                          00034400  C  002:0225:2
      IYW(KK)=CYW*100.                                                  00034500  C  002:0225:2
      YCR= 0.0                                                          00034600  C  002:0227:4
      IF (CAYS .NE. 0.0) YCR=FLOAT(YWS(KK))/CAYS                        00034700  C  002:0228:2
      ICYR(KK)=YCR*100.                                                 00034800  C  002:0233:3
      GO TO 700                                                         00034900  C  002:0235:0
```

```
  892 BYW=FLOAT(IYAW(KK))/BYAW                                          00035000  C  002:022E:2
      ITW(KK)=BYW*100.                                                  00035100  C  002:0230:2
      YBR=0.0                                                           00035200  C  002:0232:4
      IF (BAYS .NE. 0.0)YBR=FLOAT(NYS(KK))/BAYS                         00035300  C  002:0233:2
      NBR(KK)=YBR*100.                                                  00035400  C  002:0236:3
      GO TO 795                                                         00035500  C  002:0238:5
  893 SYW=FLOAT(IYAW(KK))/SYAW                                          00035600  C  002:0239:2
      ITW(KK)=SYW*100.                                                  00035700  C  002:023B:2
      YSR=FLOAT(NYS(KK))/SAYS                                           00035800  C  002:023D:4
      NSR(KK)=YSR*100.                                                  00035900  C  002:023F:4
  700 CONTINUE                                                          00036000  C  002:0241:0
      IF(YEAPLS .IS. *YES*) GO TO 95                                    00036100  C  002:0244:1
C     SET ASSOCIATED TAGS AND PROPER RETURN POINT                       00036200  C  002:0246:4
      DO 89 M=1,NC                                                      00036300  C  002:0248:4
      LW(M)=IW(M)                                                       00036400  C  002:024B:0
      LTAG(M)=ITAG(M)                                                   00036500  C  002:024A:2
      GW(M)=IW(M)                                                       00036600  C  002:024C:4
      GTAG(M)=ITAG(M)                                                   00036700  C  002:024F:0
      LSE(M)=IS(M)                                                      00036800  C  002:0251:2
      GSE(M)=IS(M)                                                      00036900  C  002:0253:4
   89 CONTINUE                                                          00037000  C  002:0256:0
C     TRANSFER FOR SORT                                                 00037100  C  002:0258:1
      IRETURN=1                                                         00037200  C  002:025A:3
      GO TO 8889                                                        00037300  C  002:025B:5
8889  PRTSEX=1                                                          00037400  C  002:025D:2
C     SET MEAN HEIFER HEADING AND SET PRTSEX TO GET OTHER HEADINGS      00037500  C  002:025A:0
      SEX(1)=*   MEAN                                                   00037600  C  002:025A:0
      SEX(2)=*YLING *                                                   00037700  C  002:025C:1
      SEX(3)=*HEIFER*                                                   00037800  C  002:025D:1
      SEX(4)=*S     *                                                   00037900  C  002:025E:1
      SEX(5)=*      *                                                   00038000  C  002:025E:1
      SEX(6)=*      *                                                   00038100  C  002:025F:1
      SEX(7)=*      *                                                   00038200  C  002:0261:1
      WRITE (I,8888) NC, AW,HH                                          00038300  C  002:0263:1
97    LINE=1                                                            00038400  C  002:0276:2
      JL=0                                                              00038500  C  002:0276:1
      JU=0                                                              00038600  C  002:0276:5
C     TEST OUT SET OF ANIMAL IF INPUT IS 0 OR BLANK AND WRITE OUT       00038700  C  002:0277:0
C     A PAGE OF DATA                                                    00038800  C  002:0277:1
      WRITE(I,9999)                                                     00038900  C  002:0279:3
      IF(PRTSEX .EQ. PRTSEX) GO TO 90                                   00039000  C  002:027A:1
      JL=JL+1                                                           00039100  C  002:027A:0
      JU=JU+1                                                           00039200  C  002:027C:2
91    IF(LSE(JL) .EQ. PRTSEX) GO TO 92                                  00039300  C  002:027D:4
      JL=JL+1                                                           00039400  C  002:027F:4
      GO TO 93                                                          00039500  C  002:0281:0
92    IF(GSE(JU) .EQ. PRTSEX) GO TO 94                                  00039600  C  002:0281:3
      JU=JU+1                                                           00039700  C  002:0283:3
      GO TO 92                                                          00039800  C  002:0284:5
94    IF(LINE .LT. 28)GO TO 288                                         00039900  C  002:0285:2
      LINE=1                                                            00040000  C  002:0286:4
      WRITE (6,1111)HH,   WORD                                          00040100  C  002:0287:2
      WRITE(6,221) SEX                                                  00040200  C  002:0290:2
      WRITE(6,222)                                                      00040300  C  002:0297:2
      IF(COPYS .LE. 9) GO TO 988                                        00040400  C  002:0298:2
      DO 795 I=10,COPYS                                                 00040500  C  002:029C:4
      WRITE (I,1111)HH,   WORD                                          00040600  C  002:029E:0
      WRITE(I,221) SEX                                                  00040700  C  002:02AE:2
795   WRITE(I,222)                                                      00040800  C  002:02B0:2
  988 WRITE(6,232)ITAG(KK),IBR(KK),IS(KK),IM(KK,1),ID(KK,1),            00040900  C  002:02B3:5
     1ITR(KK,1),IS,I(KK),IM(KK,2),ID(KK,2),ITR(KK,2),IAWM(KK),IWS(KK),  00041000  C  002:02D9:0
```

```
     *IDANO(KK),IDA(KK),IS(PE(KK),                                      00041100  C  002:02E9:1
     2                                  IWAMST(KK),IADW(KK),IWM(KK),ICR(KK),  00041200  C  002:02EF:4
     IWPD(KK),IP(KK),GTAG(JG),GWW(JG),LTAG(JL),LWW(JL)                  00041300  C  002:02F8:2
      IF(IS(KK).NE.4) WRITE(3,1400) IDANO(KK),IAW(KK)                   00041400  C  002:0302:2
      IF(COPYS .LE. 9) GO TO 75                                         00041500  C  002:0314:2
      GO TO 77 I=10,COPYS                                               00041600  C  002:0315:4
   77 WRITE(1,232)ITAS(KK),IBP(KK),IS(KK),IM(KK,1),ID(KK,1),            00041700  C  002:0317:3
     1ITP(KK,1),IB*I(KK),IM(KK,2),ID(KK,2),ITP(KK,2),IAWM(KK),IWS(KK),  00041800  C  002:0320:3
     *IDANO(KK),IDA(KK),IS(PE(KK),                                      00041900  C  002:0330:4
     2                                  IWAMST(KK),IADW(KK),IWM(KK),ICR(KK),  00042000  C  002:0343:1
     1*PD(KK),IP(KK),GTAG(JG),GWW(JG),LTAG(JL),LWW(JL)                  00042100  C  002:0345:5
75    LINE=LINE+1                                                      00042200  C  002:0350:3
   96 CONTINUE                                                         00042300  C  002:0358:5
C     SET PRTSEX TO GET PROPER SEX GROUP HEADING                        00042400  C  002:0361:0
      IF(PRTSEX .EQ.0) GO TO 95                                         00042500  C  002:0361:0
      PRTSEX=PRTSEX+1                                                   00042600  C  002:0362:1
      IF(PRTSEX .EQ. 4) GO TO 952                                       00042700  C  002:0363:3
      IF(PRTSEX .EQ. 2) SEX(3)=*  BULL*                                 00042800  C  002:0364:3
      IF(PRTSEX .EQ. 3) SEX(3)=*STEER*                                  00042900  C  002:0368:1
      GO TO 97                                                          00043000  C  002:0369:1
C     ASSIGN INSUFFICIENT DATA HEADING                                  00043100  C  002:0369:4
952   PRTSEX=2                                                          00043200  C  002:0369:4
      SEX(3)=*ASSLI*                                                    00043300  C  002:0369:4
      SEX(2)=*VS LAB*                                                   00043400  C  002:0370:1
      SEX(3)=*OPN  O*                                                   00043500  C  002:0372:1
      SEX(4)=*R INSU*                                                   00043600  C  002:0375:1
      SEX(5)=*FFICIE*                                                   00043700  C  002:0378:1
      SEX(6)=*NT DAT*                                                   00043800  C  002:0379:1
      SEX(7)=*A     *                                                   00043900  C  002:0379:1
      GO TO 97                                                          00044000  C  002:0379:1
   95 LOOP 3                                                            00044100  C  002:0381:4
      IF(ISEX(3). IS .*NO*) GO TO 395                                   00044200  C  002:0384:1
C     SET PRTSEX TO CORRECT TAGS AND PROPER RETURN POINT                00044300  C  002:0385:4
      PRTSEX=PRTSEX+1                                                   00044400  C  002:0385:4
      LTAG(JL)=LTAG(JL)                                                 00044500  C  002:0387:3
      LTAG(JL)=LTAG(JL)                                                 00044600  C  002:0387:3
      GTAG(JG)=GTAG(JG)                                                 00044700  C  002:0387:3
      GTAG(JG)=GTAG(JG)                                                 00044800  C  002:0387:3
      ISEX(3)=ISEX(3)                                                   00044900  C  002:0388:1
395   ISEX(3)=ISEX(3)                                                   00045000  C  002:0392:4
      IRETURN=2                                                         00045100  C  002:0397:1
      GO TO 8888                                                        00045200  C  002:0398:3
C     REPORTING HEIFER HEADING                                          00045300  C  002:0398:3
3888  PRTSEX=3                                                          00045400  C  002:0399:5
      SEX(1)=*   YEA*                                                   00045500  C  002:0399:1
      SEX(2)=*RLING *                                                   00045600  C  002:0399:1
      SEX(3)=*HEIFER*                                                   00045700  C  002:0390:1
      SEX(4)=*S     *                                                   00045800  C  002:03A0:1
      SEX(5)=*      *                                                   00045900  C  002:03A3:1
      SEX(6)=*      *                                                   00046000  C  002:03A5:1
397   LINE=50                                                           00046100  C  002:03A7:1
      JJ=5                                                              00046200  C  002:03A8:0
      JL=0                                                              00046300  C  002:03AA:4
C     BLANK OUT SEX IF D OR BLANK ON INPUT AND WRITE OUT YEARLING DATA  00046400  C  002:03A8:2
      DO 950 JJ=1,NC                                                    00046500  C  002:03A9:2
      IF(ISEX(JJ) .NE. PRTSEX) GO TO 950                                00046600  C  002:03AC:0
      SEX(7)=*      *                                                   00046700  C  002:03AE:0
      JL=JL+1                                                           00046800  C  002:03B0:1
      JG=JG+1                                                           00046900  C  002:03B1:5
393   IF(ISEX(JL) .EQ. PRTSEX) GO TO 392                                00047000  C  002:03B2:5
      JL=JL+1                                                           00047100  C  002:03B4:5
```

```
      GO TO 393                                                         00047200  C  002:0395:1
392   IF(QSEX(JQ) .EQ. PRTSEX) GO TO 394                                00047300  C  002:0396:4
      JQ=JQ+1                                                           00047400  C  002:0398:6
      GO TO 392                                                         00047500  C  002:039A:0
394   IF(LINE .LT. 25)GO TO 951                                         00047600  C  002:039A:3
      LINE=1                                                            00047700  C  002:039B:5
      WRITE (6,1333)KM,    WORD                                         00047800  C  002:039C:3
      WRITE(6,300) SEX                                                  00047900  C  002:039C:2
      WRITE(6,244)                                                      00048000  C  002:039C:2
      IF(COPYS .LE. 9) GO TO 951                                        00048100  C  002:039D:2
       DO 706 I=10,COPYS                                                00048200  C  002:039D:4
      WRITE (I,1333)KM,    WORD                                         00048300  C  002:039D:0
      WRITE(I,300) SEX                                                  00048400  C  002:03E3:2
706   WRITE(I,244)                                                      00048500  C  002:03F2:2
  951 WRITE(6,233) ITAG(JJ), IBR(JJ), IS(JJ),                           00048600  C  002:0400:3
     1                    IM(JJ,3),ID(JJ,3),IYR(JJ,3),IFYWT(JJ),       00048700  C  002:0409:4
     1IYAW(JJ),NADY(JJ),IYW(JJ),NYS(JJ),NCR(JJ),NDBW(JJ),LGA(JJ),ADG(JJ) 00048800  C  002:0413:2
     1,WCA(JJ),YPRCB(JJ),QTAG(JC),QWW(JC),LTAG(JL),LWW(JL)              00048900  C  002:042:0
      IF(COPYS .LE. 9) GO TO 78                                         00049000  C  002:0434:2
      DO 79 I=10,COPYS                                                  00049100  C  002:0-35:4
   79 WRITE(I,233) ITAG(JJ), IBR(JJ), IS(JJ),                           00049200  C  002:0-37:0
     1                    IM(JJ,3),ID(JJ,3),IYR(JJ,3),IFYWT(JJ),       00049300  C  002:0448:1
     1IYAW(JJ),NADY(JJ),IYW(JJ),NYS(JJ),NCR(JJ),NDBW(JJ),LGA(JJ),ADG(JJ) 00049400  C  002:0451:5
     1,WCA(JJ),YPRCB(JJ),QTAG(JC),QWW(JC),LTAG(JL),LWW(JL)              00049500  C  002:0462:3
78    LINE=LINE+1                                                       00049600  C  002:0474:3
  950 CONTINUE                                                          00049700  C  002:0475:5
C     SET PRTSEX TO GET PROPER SEX GROUP HEADING                        00049800  C  002:0478:0
      IF(PRTSEX .EQ. 0) GO TO 395                                       00049900  C  002:0478:0
      PRTSEX=PRTSEX+1                                                   00050000  C  002:0479:1
      IF(PRTSEX .EQ. 4) GO TO 352                                       00050100  C  002:047A:3
      IF(PRTSEX .EQ. 2) SEX(3)='  BULL'                                 00050200  C  002:047B:3
      IF(PRTSEX .EQ. 3) SEX(3)=' STEER'                                 00050300  C  002:047F:1
      GO TO 397                                                         00050400  C  002:0483:1
C     YEARLING INSUFFICIENT DATA HEADING                                00050500  C  002:0483:4
352   WRITE(6                                                           00050600  C  002:0483:4
      SEX(1)='YEARLI'                                                   00050700  C  002:0484:2
      SEX(2)='NG INS'                                                   00050800  C  002:0487:1
      SEX(3)='UFFICI'                                                   00050900  C  002:0489:1
      SEX(4)='ENT DA'                                                   00051000  C  002:048C:1
      SEX(5)='TA    '                                                   00051100  C  002:048F:1
      GO TO 397                                                         00051200  C  002:0492:1
C     STOP  LAMING DATA                                                 00051300  C  002:0492:4
  395 CLOSE 5                                                           00051400  C  002:0494:4
C     READ ANY SPECIAL NOTES IN DATA DECK                               00051500  C  002:0494:0
      READ(4,900,END=905)WORD                                           00051600  C  002:0494:0
      WRITE(6,902)                                                      00051700  C  002:0490:0
      IF(COPYS .LE. 9) GO TO 903                                        00051800  C  002:04A0:2
      DO 904 I=10,COPYS                                                 00051900  C  002:0-A1:4
904   WRITE(I,902)                                                      00052000  C  002:04A3:0
929   WRITE(6,901) WORD                                                 00052100  C  002:0-B1:3
      IF(COPYS .LE. 9) GO TO 903                                        00052200  C  002:04B4:2
      DO 907 I=10,COPYS                                                 00052300  C  002:04B4:4
907   WRITE(I,901) WORD                                                 00052400  C  002:04B8:0
903   READ(4,900,END=905) WORD                                          00052500  C  002:04C2:3
      GO TO 929                                                         00052600  C  002:04D4:0
905   WRITE(6,470)                                                      00052700  C  002:04D4:3
      IF(COPYS .LE. 9) GO TO 707                                        00052800  C  002:04D3:2
      DO 704 I=10,COPYS                                                 00052900  C  002:04D9:4
704   WRITE(I,470)                                                      00053000  C  002:04DB:0
707   STOP                                                              00053100  C  002:04E9:3
C     SORT INTO DECENDING ORDER BY MEAN WT RATIO RANKING AND ACTUAL     00053200  C  002:04EA:2
```

```
C       WEAN WT
 8888 DO 90 I=1,NC
      DO 90 J=I+1,NCARD
      IF(LWW(I).GE.LWW(J))GO TO 91
      L=LWW(I)
      LWW(I)=LWW(J)
      LWW(J)=L
      L=LSEX(I)
      LSEX(I)=LSEX(J)
      LSEX(J)=L
      S=LTAG(I)
      LTAG(I)=LTAG(J)
      LTAG(J)=S
91    IF(QWW(I) .GE. QWW(J)) GO TO 90
      L=QWW(I)
      QWW(I)=QWW(J)
      QWW(J)=L
      L=QSEX(I)
      QSEX(I)=QSEX(J)
      QSEX(J)=L
      S=QTAG(I)
      QTAG(I)=QTAG(J)
      QTAG(J)=S
C     GO BACK TO SET SEX TO PRINTOUT BY SEX GROUP
   90 CONTINUE
      GO TO (8889,8887) IRETRN
      STOP
      END
```

```
00053300  C  002:04EA:2
00053400  C  002:04EA:0
00053500  C  002:04EB:0
00053600  C  002:04ED:0
00053700  C  002:04EF:4
00053800  C  002:04F1:1
00053900  C  002:04F3:3
00054000  C  002:04F5:1
00054100  C  002:04F6:4
00054200  C  002:04F9:0
00054300  C  002:04FA:4
00054400  C  002:04FC:1
00054500  C  002:04FE:3
00054600  C  002:0500:1
00054700  C  002:0502:5
00054800  C  002:0504:2
00054900  C  002:0506:4
00055000  C  002:050B:2
00055100  C  002:0509:5
00055200  C  002:050C:1
00055300  C  002:050D:5
00055400  C  002:050F:2
00055500  C  002:0511:4
00055600  C  002:0513:2
00055700  C  002:0513:2
00055800  C  002:0517:4
00055900  C  002:051C:3
00056000  C  002:051D:2
          SEGMENT 002 IS 0568 LONG
```

```
                                                    FORMAT SEGMENT IS 0012 LONG
                                                       START OF SEGMENT 006
                                                    SEGMENT 006 IS 008C LONG
NO ERRORS DETECTED.  NUMBER OF CARDS = 562.
COMPILATION TIME = 27 SECONDS ELAPSED.  5.60 SECONDS PROCESSING(6021 CPM).
02 STACK SIZE = 60 WORDS.  FILESIZE = 1006 WORDS.  ESTIMATED CORE STORAGE REQUIREMENT = 43902 WORDS.
TOTAL PROGRAM CODE = 1663 WORDS.  ARRAY STORAGE = 41046 WORDS.
NUMBER OF PROGRAM SEGMENTS = 9.  NUMBER OF DISK SEGMENTS = 165.
PROGRAM CODE FILE = (180002)PERFORMANCE/WEAN. COMPILER COMPILED ON 12/18/75(FORTRAN ON PACK)
```

SECTION II

PERFORMANCE/WEAN/TERML

This is the teletype version of the P/WEAN program. At the present time of documentation of the USU cattle performance testing system, this one is just in the developmental and programing phase of implementation.

The scope and major points of performance tests will be maintained. Input will be from remote units with output to remote, line printer, and diskpack. All input is free format except for the control character and heading line, and the special notes section. All output is formated with one (1) copy going to the line printer by default. Program flow will be the same basically as P/WEAN.

The operational dead line is June 1, 1976.

SECTION III

PERFORMANCE/MPPA/IMPPAL

This program performs a vital function in that it makes the initial diskpack file for first run herds, or the capacity to bring a herd up to date without the need to use performance/wean; even if the data is punched differently from that which is in current use.

The input is from cards in two forms. Form one (1) the first card of the deck contains three (3) information parts: (1) metric conversion control (Ø = no, 1 = yes); (2) file control (0 = initial herd, 1 = update run); (3) format of the remaining cards (36 characters maximum). Form two (2) the remaining cards with data punched in as indicated by the format but in this order:

Calf sex	
Birth:	Date
	Month
	Day
	Year
Birth weight	
Wean:	Date
	Month
	Day
	Year
Actual weaning weight	
Dam number	
Dam age	

Output is a diskpack file either of two forms. Form one (1), the initial file which will contain dam number, total calves in record (1), total wean weight ratio, previous MPPA score (0), and production indicat-

tor (used in P/M/MPPA). Form two (2) updating run contains dam number and wean weight ratio. Records for an initial run are sorted by dam number before writing to the file. First record of each file contains number of records and total wean weight ratio for the herd; with initial load also giving number producing calves.

The processing of data is handled in the same manner as that in P/Wean, to get the desired output.

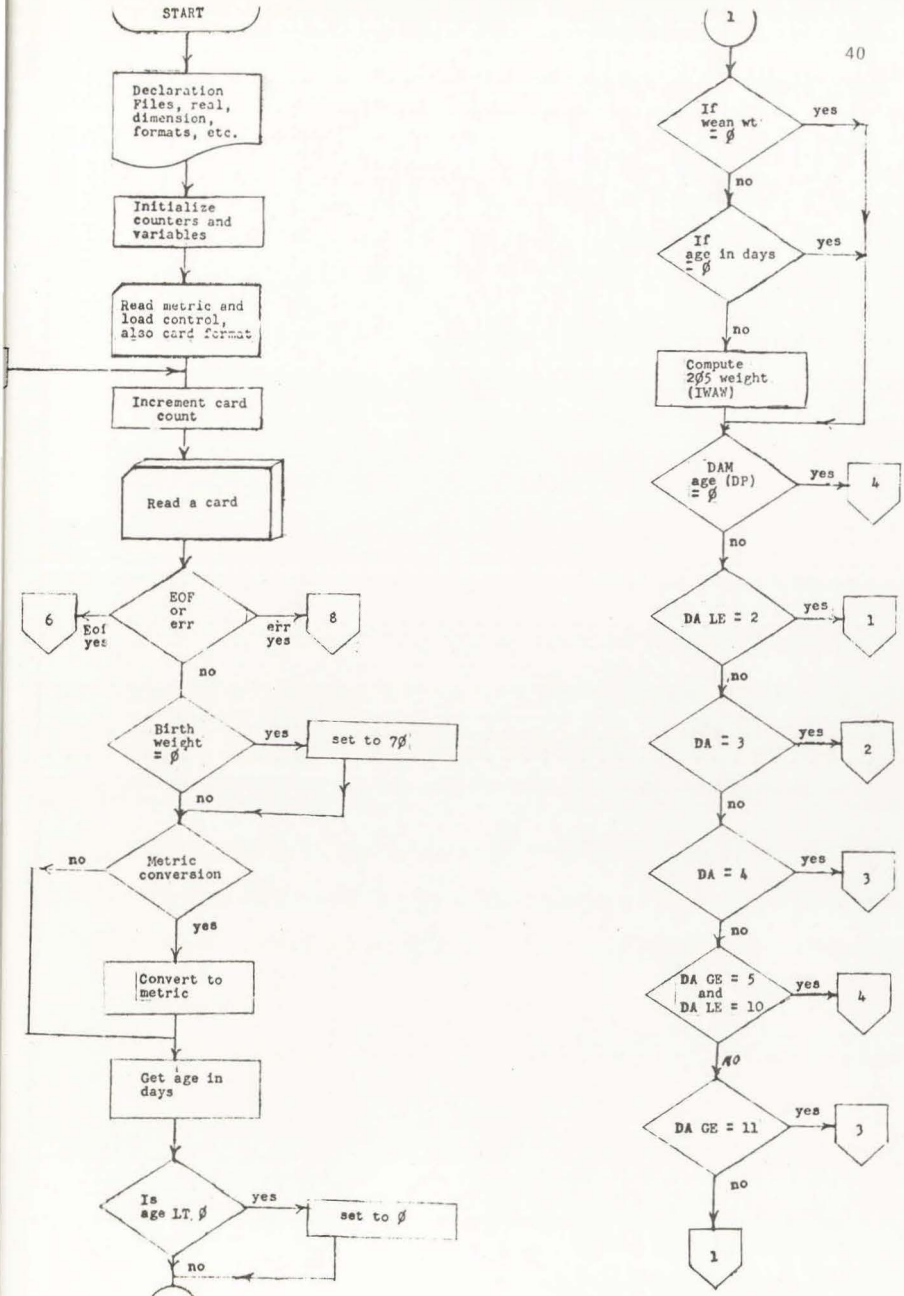
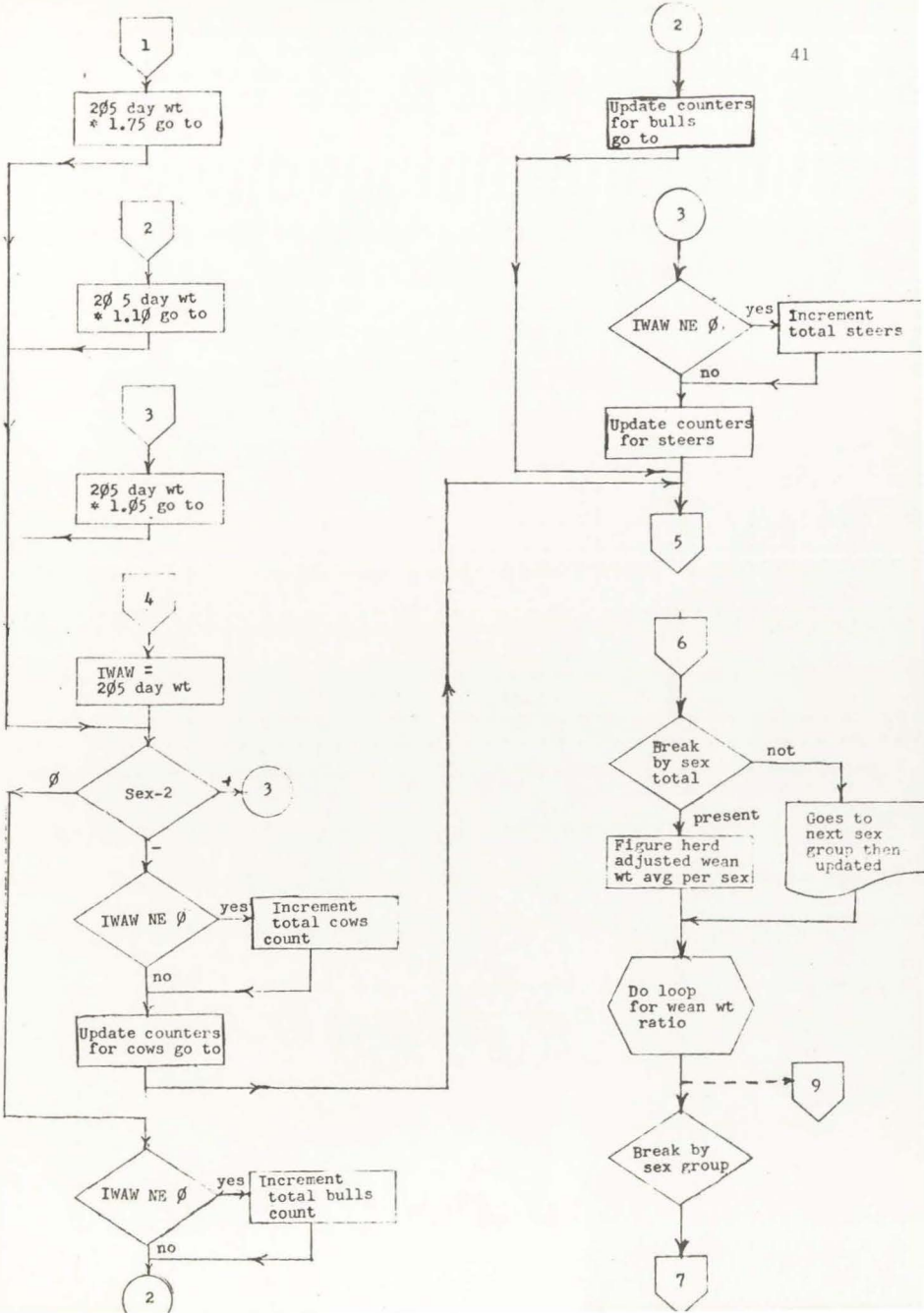
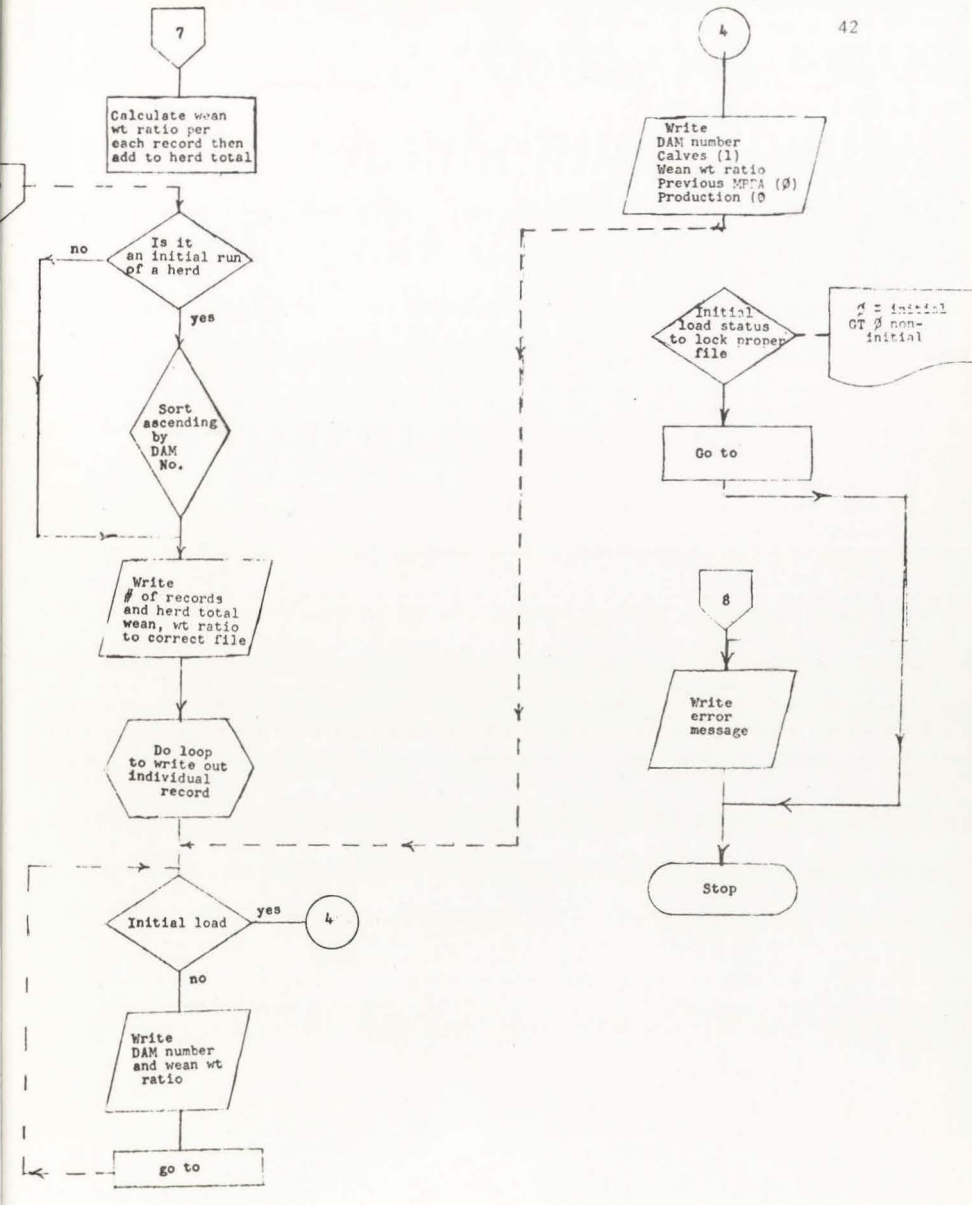


Figure 3. PERFORMANCE/MPPA/IMPPAL PROGRAM FLOW CHART


1
2Ø5 day wt * 1.75 go to
2
2Ø 5 day wt * 1.1Ø go to
3
2Ø5 day wt * 1.Ø5 go to
4
IWAW = 2Ø5 day wt
Sex-2
Ø
+
3
-
IWAW NE Ø
yes
Increment total cows count
no
Update counters for cows go to
IWAW NE Ø
yes
Increment total bulls count
no
2
2
Update counters for bulls go to
3
IWAW NE Ø
yes
Increment total steers
no
Update counters for steers
5
6
Break by sex total
not
present
Figure herd adjusted wean wt avg per sex
Goes to next sex group then updated
Do loop for wean wt ratio
9
Break by sex group
7


7
Calculate wean wt ratio per each record then add to herd total
Is it an initial run of a herd
no
yes
Sort ascending by DAM No.
Write # of records and herd total wean, wt ratio to correct file
Do loop to write out individual record
Initial load
yes
4
no
Write DAM number and wean wt ratio
go to
4
Write DAM number Calves (1) Wean wt ratio Previous MPDA (Ø) Production (Ø
Initial load status to lock proper file
Ø = initial GT Ø non-initial
Go to
8
Write error message
Stop

VARIABLE	I/O STATUS	INFORMATION	FORMAT SPECIFICATION
FMAT	I	Array which holds the run time format for the data	6A6
KM	I	Control metric conversion[3]	I1
ICY	I	Control of run output (update or initial)[3]	I1
NR	O	Number of records processed	I4
IC	O	Number of calves (I) on initial run of herd	I2
PM	O	Previous MPPA score initial run	F6.2
AWWRH	O	Herd total actual wean weight ratio	F10.0
IS	I	Calf sex	I1
IM	I	Calf date Month[2]	I2
ID		Day[2]	I2
IYR		Year[2]	I2
IBWT	I	Birth weight	I3
IAWW	I	Actual wean weight	I3
IDAMNO	I/O	Dam number	C6
IDA	I	Dam age	I2
IWW	O	Wean weight ratio	I3
AWW	O	Wean weight ratio	F6.0
NN	O	Production indicator initial run only	I1

[1] Fmat is an indicator as to where on the card the information is located. The specifications listed are the maximum allowable size. This format applies to the variable in the list IS down to IDA.

[2] These are the calf birth date and weaning date. The wean date follows the IS variable with weaning placed between IBWT and IAWW.

[3] The card with the control characters and the format is constructed with order as follows: (1) metric conversion (0 = no, else yes); output control (0 = initial herd run, else update); and (3) run time format.

Figure 4. CARD LAYOUT PERFORMANCE/MPPA/IMPPAL

```
H6703/07700   F O R T R A N   C O M P I L A T I O N   M A R K   2.7.480   TUESDAY, 03/16/76   11:19 AM

$ SET LINECNT=0                                                                    C  0001000015
$ OPT SEQ SC = 50                                                          00000050  C  0001000015
FILE  5=LOADER/DATAIN                                                      00000100  C
                                                                                        FIO IS 0008 LONG
FILE  7 (KIND=DISKPACK, MAXRECSIZE=5, BLOCKSIZE=150, SAVEFACTOR=2,         00000150  C
     * FLEXIBLE=TRUE, SECURITYTYPE=CLASSA, TITLE="LYMPPA")                  00000200  C
FILE  8 (KIND=DISKPACK, MAXRECSIZE=3, BLOCKSIZE=90, SAVEFACTOR=2,          00000250  C
     *SECURITYTYPE=CLASSA, TITLE="NOPAI")                                   00000300  C
                                                                                        START OF SEGMENT 002
      REAL EDAMNO(1000)                                                    00000350  C  0021000010
      DIMENSION FMAT(5), IM(2), ID(2), IYR(2), IAGE(2), ICUM(11),          00000400  C  0021000010
     * NAME(13), IS(1000), IWA*ST(1000), IWM(1000)                          00000450  C  0021000010
      EQUIVALENCE (IAGE(1),BIRTHD), (IAGE(2),MEANDA)                       00000500  C  0021000010
      DATA ICUM/31,59,90,120,151,181,212,243,273,304,334/,                 00000550  C  0021000010
     * NAME/1,2,3,4,5,6,7,8,9,10,11,12/                                     00000600  C  0021000010
    6 FORMAT(5A,F10.0)                                                     00000650  C  0021000010
    9 FORMAT (5A,12,F6.0,F6.2,I1)                                          00000700  C  0021000010
   19 FORMAT (2I4,F10.0)                                                   00000750  C  0021000010
  115 FORMAT(1X,"FORMAT AND DATA DO NOT AGREE")                            00000800  C  0021000010
  125 FORMAT (2I1,4A6)                                                     00000850  C  0021000010
  836 FORMAT(F6.I3)                                                        00000900  C  0021000010
      NC=0                                                                 00000950  C  0021000010
      IC=1                                                                 00001000  C  0021000014
      A**KH=0                                                              00001050  C  0021000112
      P**0                                                                 00001100  C  0021000210
      READ(5,12^) KH, ITY, FMAT                                            00001150  C  0021000214
    5 NC = NC + 1                                                          00001200  C  0021000312
      READ(5,FMAT,END=6,ERR=7)IS(NC), IM(1), ID(1), IYR(1), IAWT,          00001250  C  0021000314
     * IM(2), ID(2), IYR(2), IAWW, EDAMNO(NC), IDA                          00001300  C  0021001E10
      IF (IS(*). EQ .0) IAWT=70                                            00001350  C  0021003010
      IF (AW. EQ .0) GO TO 10                                              00001400  C  0021003210
      B*T = (FLOAT(IAWT) * 0.4536)                                         00001450  C  0021003311
      IS*T = B*T                                                           00001500  C  0021003514
      A** = (FLOAT(IAWW) * 0.4536)                                         00001550  C  0021003612
      IA** = A**                                                           00001600  C  0021003814
   10 DO 11 NTIME = 1,2                                                    00001650  C  0021003912
      ILEAP =0                                                             00001700  C  0021003410
      IAGE(NTIME)=0                                                        00001750  C  0021003414
      ALEAP = FLOAT(IYR(NTIME)) / 4.                                       00001800  C  0021003C11
      ILEAP = IYR(NTIME) / 4                                               00001850  C  0021003E11
      IF (ALEAP. EQ .ILEAP) ILEAP=1                                        00001900  C  0021004011
      DO 12 I = 1, 12                                                      00001950  C  0021004211
      IF (IM(NTIME). NE .NAME(I)) GO TO 12                                 00002000  C  0021004310
      IF (I. EQ .13) GO TO 98                                              00002050  C  0021004514
      IF (I. NE .3. AND .ILEAP. EQ .1) IAGE(NTIME) = 1                     00002100  C  0021004615
      IAGE(NTIME) = ICUM(I-1) + ID(NTIME) + IAGE(NTIME)                    00002150  C  0021004813
      GO TO 15                                                             00002200  C  0021004E15
   88 IAGE(NTIME) = ID(NTIME)                                              00002250  C  0021004F12
      GO TO 15                                                             00002300  C  0021005114
   12 CONTINUE                                                             00002350  C  0021005211
   11 CONTINUE                                                             00002400  C  0021005412
      IF (IYR(1). EQ .IYR(2)) GO TO 40                                     00002450  C  0021005615
      ILEAP = MOD(IYR(1),4)                                                00002500  C  0021005811
      IF (ILEAP. NE .0). GO TO 42                                          00002550  C  0021005215
      IAGE = (367 - BIRTHD) + MEANDA                                       00002600  C  0021005414
      GO TO 44                                                             00002650  C  0021005511
   42 IAGE = (366 - BIRTHD) + MEANDA                                       00002700  C  0021005014
      GO TO 44                                                             00002750  C  0021006011
   40 IAGE = MEANDA - BIRTHD                                               00002800  C  0021006014
```

```
 44 IF (IADW. LT .0)  IADW = 0                                          00002850  C  002:0062:3
    U = 0.0                                                             00002900  C  002:0064:2
    IF (IADW. EQ .0) GO TO 50                                           00002950  C  002:0065:3
    IF (IADW. EQ .0) GO TO 50                                           00003000  C  002:0066:1
    U = (FLOAT(IADW)-IBWT)/FLOAT(IADW)) * 205. * (FLOAT(IBWT))          00003050  C  002:0067:2
 50 IF (IDA. EQ .0) GO TO 75                                            00003100  C  002:0068:1
    IF (IDA. LE .0) GO TO 72                                            00003150  C  002:0069:2
    IF (IDA. EQ .0) GO TO 73                                            00003200  C  002:0070:4
    IF (IDA. EQ .4) GO TO 74                                            00003250  C  002:0071:3
    IF (IDA. LE .5. AND .IDA. LE .10) GO TO 75                          00003300  C  002:0072:2
    IF (IDA. LE .10) GO TO 74                                           00003350  C  002:0073:4
 72 IWW = U * 1.75                                                      00003400  C  002:0074:3
    GO TO 76                                                            00003450  C  002:0075:3
 73 IWW = U * 1.10                                                      00003500  C  002:0076:2
    GO TO 76                                                            00003550  C  002:0077:5
 74 IWW = U * 1.05                                                      00003600  C  002:0078:2
    GO TO 76                                                            00003650  C  002:0079:5
 75 IWW = U                                                             00003700  C  002:0080:2
 76 IF (IS(NC) - 2) 81, 82, 83                                          00003750  C  002:0081:2
 81 IF (IWW. NE .0) ITC = ITC + 1                                       00003800  C  002:0082:2
    IWW=ST(NC) = IWW                                                    00003850  C  002:0083:5
    ICCW = ICCW + IWW                                                   00003900  C  002:0084:5
    GO TO 85                                                            00003950  C  002:0085:3
 82 IF (IWW. NE .0) ITB = ITB + 1                                       00004000  C  002:0086:3
    IWW=ST(NC) = IWW                                                    00004050  C  002:0087:0
    IBCLL = IBCLL + IWW                                                 00004100  C  002:0088:4
    GO TO 85                                                            00004150  C  002:0089:1
 83 IF (IWW. NE .0) ITS = ITS + 1                                       00004200  C  002:0090:4
    IWW=ST(NC) = IWW                                                    00004250  C  002:0091:1
    ISCT = ISCT + IWW                                                   00004300  C  002:0092:5
 85 GO TO 5                                                             00004350  C  002:0093:2
  6 IF (ITC. EQ .0) GO TO 101                                           00004400  C  002:0094:5
    CWAW = FLOAT(ICCW) / FLOAT(ITC)                                     00004450  C  002:0095:0
101 IF (ITB. EQ .0) GO TO 102                                           00004500  C  002:0096:2
    BWAW = FLOAT(IBCLL) / FLOAT(ITB)                                    00004550  C  002:0097:3
102 IF (ITS. EQ .0) GO TO 104                                           00004600  C  002:0098:5
    SWAW = FLOAT(ISCT) / FLOAT(ITS)                                     00004650  C  002:0099:3
104 NK = NC - 1                                                         00004700  C  002:0099:2
    DO 700 KK = 1, NK                                                   00004750  C  002:0099:4
    IF (IS(KK) -2) 701, 702, 703                                        00004800  C  002:0099:0
701 IWW(KK) = (FLOAT(IWWST(KK)) / CWAW) * 100.                          00004850  C  002:00A2:0
    GO TO 704                                                           00004900  C  002:00A5:3
702 IWW(KK) = (FLOAT(IWWST(KK)) / BWAW) * 100.                          00004950  C  002:00A6:0
    GO TO 704                                                           00005000  C  002:00A9:3
703 IWW(KK) = (FLOAT(IWWST(KK)) / SWAW) * 100.                          00005050  C  002:00AA:0
704 AWWH = AWWH + FLOAT(IWW(KK))                                        00005100  C  002:00AD:3
700 CONTINUE                                                            00005150  C  002:00AE:3
    IF ( ITT. NE .0) GO TO 800                                          00005200  C  002:00B1:4
    DO 800 I = 1, NK-1                                                  00005250  C  002:00B2:5
    DO 800 J = I+1, NK                                                  00005300  C  002:00B6:3
    IF (IDAMND(I). GE .IDAMND(J)) GO TO 800                             00005350  C  002:00B8:0
    RTEM = IDAMND(I)                                                    00005400  C  002:00BA:4
    ITEM = IWW(I)                                                       00005450  C  002:00BC:1
    IDAMND(I) = IDAMND(J)                                               00005500  C  002:00BD:4
    IWW(I) = IWW(J)                                                     00005550  C  002:00C0:0
    IDAMND(J) = RTEM                                                    00005600  C  002:00C2:2
    IWW(J) =ITEM                                                        00005650  C  002:00C4:0
800 CONTINUE                                                            00005700  C  002:00C5:4
    IF (ITT. NE .0) WRITE(8,8) NR, AWWRH                                00005750  C  002:00C8:2
    IF (ITT. NE .0) GO TO 820                                           00005800  C  002:00D1:2
    WRITE (7,19) NR, NR, AWWRH                                          00005850  C  002:00D2:3
```

```
820 DO 801 K = 1, NR
    AWW = FLOAT(IWW(K))
    IF (ITY. EQ .0) GO TO 802
    WRITE(6,806) IDAMNO(K), IWW(K)
    GO TO 801
802 WRITE(7,9)IDAMNO(K), IC, AWW, PM, NN
801 CONTINUE
    IF (ITY. NE .0) LOCK 8
    IF (ITY. EQ .0) LOCK 7
    GO TO 810
  7 WRITE (6,110)

810 STOP
    END
```

```
00005900  C  002:0000:7
00005950  C  002:0000:0
00006000  C  002:000E:3
00006050  C  002:000F:4
00006100  C  002:00E9:2
00006150  C  002:00E8:5
00006200  C  002:00F7:2
00006250  C  002:00F9:3
00006300  C  002:00FC:1
00006350  C  002:00FE:5
00006400  C  002:00FF:2
                FIB IS 0006 LONG
00006450  C  002:0103:2
00006500  C  002:0104:1
        SEGMENT 002 IS 0110 LONG
```

```
                                                    FORMAT SEGMENT IS 0020 LONG
                                                       START OF SEGMENT 005
                                                    SEGMENT 006 IS 0026 LONG
NO ERRORS DETECTED.  NUMBER OF CARDS =  131.
COMPILATION TIME = 51 SECONDS ELAPSED.  1.93 SECONDS PROCESSING.
D2 STACK SIZE = 19 WORDS.  FILESIZE = 620 WORDS.  ESTIMATED CORE STORAGE REQUIREMENT = 5096 WORDS.
TOTAL PROGRAM CODE = 376 WORDS.  ARRAY STORAGE = 4037 WORDS.
NUMBER OF PROGRAM SEGMENTS = 6.  NUMBER OF DISK SEGMENTS = 47.
PROGRAM CODE FILE = (180002)PERFORMANCE/MPPA/IMPPAL. COMPILER COMPILED ON 12/18/75(FORTRAN ON PACK)
```

SECTION IV

PERFORMANCE/MPPA/LPMPPA

This program loads previous year's most probably producing ability (LPMPPA) and removes all the dams dropped from the herd since the last processing of the data.

The program input consists of two (2) card decks: one of data records and the other of numbers of those that will be removed from the active file. Record file is titled PMPPA with two (2) records per card. The removal file is titled REMOVE with each number being placed one per card in the first six (6) columns coded exactly as it has previously appeared in past processing. Numbers to be removed will need to be placed in a sorted order (ascending).

The removal process takes place during writing of the data record to the diskpack file (titled LYMPPA); also an indicator as to production is added to those records which are written to the file provided there are no records in the remove file. The removal variable is set to an alphanumeric "999999". During the removal phase, a printed copy of those dams dropped along with the reason is transferred to the line printer.

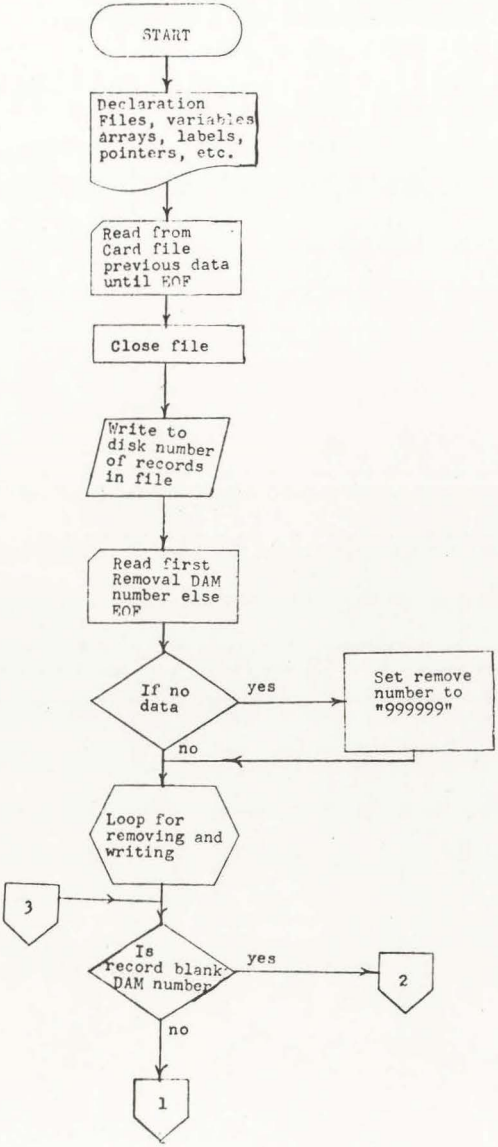
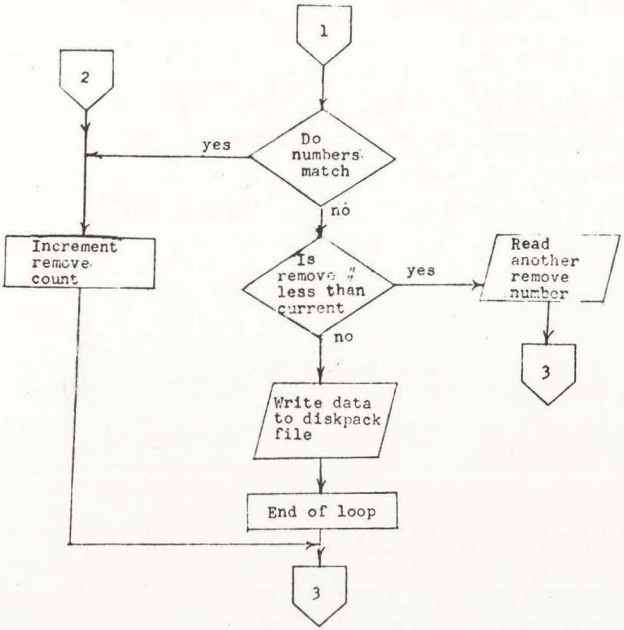
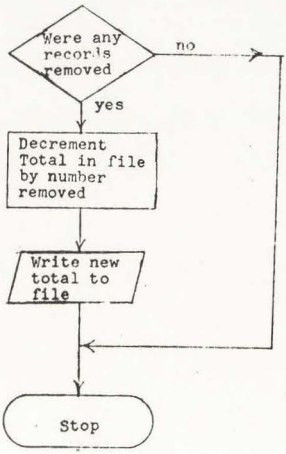


Figure 5. PERFORMANCE/MPPA/LPMPPA PROGRAM FLOW CHART


1
2
Do numbers match
yes
no
Increment remove count
Is remove # less than current
yes
Read another remove number
3
no
Write data to diskpack file
End of loop
3


Were any records removed
no
yes
Decrement Total in file by number removed
Write new total to file
Stop

VARIABLE	I/D STATUS	INFORMATION	FORMAT SPECIFICATION
A	I/O	Card image in (array, 14 words)	Machine relative
B	O	Array (2 dimensional) into which the two (2) records in A are split	Machine relative
I	O	Number of records in the file after the first is written	I4

This program uses two (2) card files: first file contains the previous record, and the second the number to be removed.

The previous record file is first to be opened with the title of PMPPA which is read in machine relative form. This provides a rapid method for I/O, but for later use of the data it will need to be in the form listed below. Cards from the punch program P/M/CARDS are in this form already.

COLUMNS	DATA
1-6 and 40-45	Dam number (alphanumeric, A6, maximum per number)
7-8 and 46-47	Total calves produced (integer, I2)
9-13 and 48-52	Total actual adjusted wean weight ratio (real, F5.0)
14-18 and 53-57	Previous MPPA score (real, F6.2)

(Two [2] different records per card)

The second file opened (or tested for data) is the removal file which is titled REMOVE. This is one dam number per card punched in columns 1-6 in the same way as the number to be removed (the way it appears in the other input data, it must be exact or it will not be

removed).

When none are to be removed program default is an alphanumeric "999999". Columns 7-36 reason for dropping from herd.

Output is to a diskpack file (title LYMPPA). The file contains one dam record per data file record. The records are written in machine relative form. The first record in the file indicates the number of data records to follow in the file. There is a production indicator placed in the record at the time of writing. All records with blank dam numbers are removed automatically by the program as it is assumed to be non-valid information. A printer copy of dams dropped is also produced.

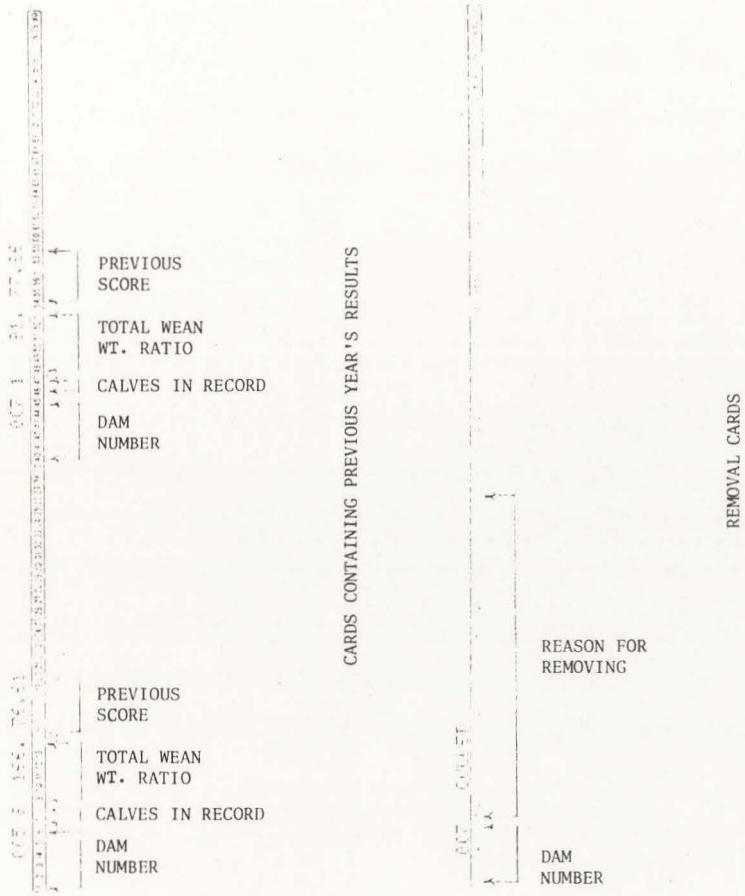


Figure 6. CARD LAYOUT PERFORMANCE/MPPA/LPMPPA

DAMS REMOVED

005	CULLED
049	DIED
106	AB[illegible]RED
304	BREEDING FAILURE
817	DUPLICATE RECORD
819	DUPLICATE

P E R F O R M A N C E / M P P A / L P M E P A

```
 BEGIN                                                                  00000050  0000000F0
                                                                        9.0000 IS SEGMENT 00003
  FILE REMOVE (KIND=READER);                                          1 00000100  0000000B1
                                                                          DATA IS 0005 LONG
  FILE PMPPA (KIND=READER);                                             00000150  0000000B1
                                                                          DATA IS 0005 LONG
  FILE LP (KIND=PRINTER);                                               00000200  0000000B1
                                                                          DATA IS 0004 LONG
  FILE LYMPPA(KIND=DISKPACK,MAXRECSIZE=5,BLOCKSIZE=150,SAVEFACTOR=2,    00000250  0000000B1
     FLEXIBLEFAT,F,SECURITYTYPE=CLASS4);                                00000300  0000000B1
                                                                          DATA IS 0007 LONG
  ALPHA ARRAY A[0:13], B[0:1499,0:07];                                  00000350  0000000B1
  INTEGER I, M, J;                                                      00000400  0000000B2
  LABEL D, EOF, EOF1, REM, RET, MATCH, LESS, PRINT, EOF2, IPF;          00000450  0000000B2
  POINTER PA;                                                           00000500  0000000B2
  PA := POINTER (A);                                                    00000550  0000000B2
%                                                                       00000600  0000000B5
%         READ IN PREVIOUS MPPA DATA                                    00000650  0000000B5
%                                                                       00000700  0000000B5
D:  WRITE(LYMPPA,<14>,A) [EOF];                                         00000750  0000000B5
    REPLACE POINTER(B[I,0]) BY PA FOR 39;                               00000800  0000000B5
    REPLACE POINTER(B[I+1,0]) BY PA+59 FOR 40;                          00000850  0000000B4
    I := **2;                                                           00000900  0000000B1
    GO TO D;                                                            00000950  0000000B1
EOF: CLOSE (PMPPA);                                                     00001000  0000000B4
%                                                                       00001050  0000000B1
%         CHECK TO SEE IF ANY RECORDS ARE TO BE REMOVED                 00001100  0000000B1
%         START WRITING RECORDS TO DISK FILE                            00001150  0000000B3
%                                                                       00001200  0000000B3
    WRITE (LYMPPA,<I4>,I);                                              00001250  0000000B3
    READ (REMOVE,6,A) [EOF1];                                           00001300  0000000B3
    WRITE(LP,SPACE 1,<"DAMS REMOVED">);                                 00001350  0000000B2
    GO TO RET;                                                          00001400  0000000B1
EOF1:  REPLACE PA BY "9" FOR 6;                                         00001450  0000000B2
REM: FOR J := 0 STEP 1 UNTIL I-1 DO                                     00001500  0000000B5
     BEGIN                                                              00001550  0000000B5
%                                                                       00001600  0000000B5
%         SERIES OF CHECKS FIRST FOR BLANK DAM NUMBER                   00001650  0000000B2
%         SECOND FOR A MATCH WITH ONE TO BE REMOVED                     00001700  0000000B2
%         THIRD TO SEE IF NUMBER IS LARGER THAN REMOVE TAG ENTERED      00001750  0000000B2
%                                                                       00001800  0000000B2
      IF POINTER(B[J,0]) = " " FOR 6 THEN BEGIN                         00001850  0000000B2
         M := + 1;                                                      00001900  0000000B2
         GO TO EOF2;                                                    00001950  0000000B2
        END;                                                            00002000  0000000B2
RET:  IF PA = POINTER(B[J,0]) FOR 6 THEN GO TO MATCH;                 3 00002050  0000000B1
      IF PA <= POINTER(B[J,0]) FOR 6 THEN GO TO LESS;                 3 00002100  0000000B1
      GO TO PRINT;                                                      00002150  0000000B1
MATCH: M := **1;                                                        00002200  0000000B2
       WRITE(LP,6,A);                                                   00002250  0000000B5
       READ(REMOVE,6,A) [EOF2];                                         00002300  0000000B5
       GO TO EOF2;                                                      00002350  0000004B1
LESS: READ(REMOVE,6,A) [PRINT];                                         00002400  0000004B4
```

```
      GO TO REI;                                                          00002450  003:0057:1
%                                                                         00002500  003:0057:4
%        WRITE RECORD TO DISK AFTER PUTTING PRODUCTION INDICATOR IN       00002550  003:0057:4
%                                                                         00002600  003:0057:4
PRINT: REPLACE POINTER(B(J,3))+2 BY "1" FOR 1;                            00002650  003:0057:4
      WRITE(LYMPPA,05,H(J,*));                                            00002700  003:0058:2
EOF2: END;                                                                00002750  003:0062:2
%                                                                       2 00002800  003:0062:5
%        IF ANY RECORDS ARE REMOVED CHANGE TOTAL AND WRITE TO FILE        00002850  003:0062:5
%                                                                         00002900  003:0062:5
    IF M = 0 THEN GO TO IPF;                                              00002950  003:0062:5
    I := * - M;                                                           00003000  003:0064:0
    WRITE (LYMPPA(0),<I4>, I);                                            00003050  003:0065:3
IPF: LOCK (LYMPPA);                                                       00003100  003:0066:2
 END.                                                                     00003150  003:0066:5
                                                                        B.C0004003) IS 0042 LONG
                                                                                DATA IS 0008 LONG
```

```
NUMBER OF ERRORS DETECTED = 0.
NUMBER OF SEGMENTS = 3.  TOTAL SEGMENT SIZE = 162 WORDS.  CORE ESTIMATE = 3623 WORDS.  STACK ESTIMATE = 15
PROGRAM SIZE = 63 CARDS, 398 SYNTACTIC ITEMS, 21 DISK SEGMENTS.
PROGRAM FILE NAME:  PERFORMANCE/MPPA/LPMPPA.
COMPILATION TIME = 9.141 SECONDS ELAPSED;  1.023 SECONDS PROCESSING;  1.691 SECONDS I/O.
```

SECTION V

PERFORMANCE/MPPA/MERGE

Most probable producing ability file merge (P/M/MERGE). This program merges two (2) diskpack files coming from P/M/LPMPPA (title-LYMPPA) and P/Wean (or P/W/terminl--title-MPPA1).

The merge is done by means of a binary scheme of lookup. New records to be added to active files are done by holding the unfound number in an array which will be expanded to meet the size of current file then placed at the end of the file upon which a sort is performed after adding all new records. When a record is located the proper accumulators are updated to reflect a change in status.

File structure for the two (2) input files is very different in terms of record size and function. LYMPPA provides data of previous years that will be needed for processing. This file has four (4) parts (dam number, number of calves in record, total actual wean weight ratio, and last computed MPPA record). MPPA1 contains two (2) data items which are the dam number and new actual wean weight ratio; when the program finishes execution this file is purged from the system.

A test is made to determine if any new records are to be placed in the file; upon indication there are no new records the results are returned to the file. New records up to a maximum of five hundred (500) are handled by first expanding the individual record in MPPA1 to the size and arrangement and placed at the end of LYMPPA, with a sort by dam number next performed before writing back to the file.

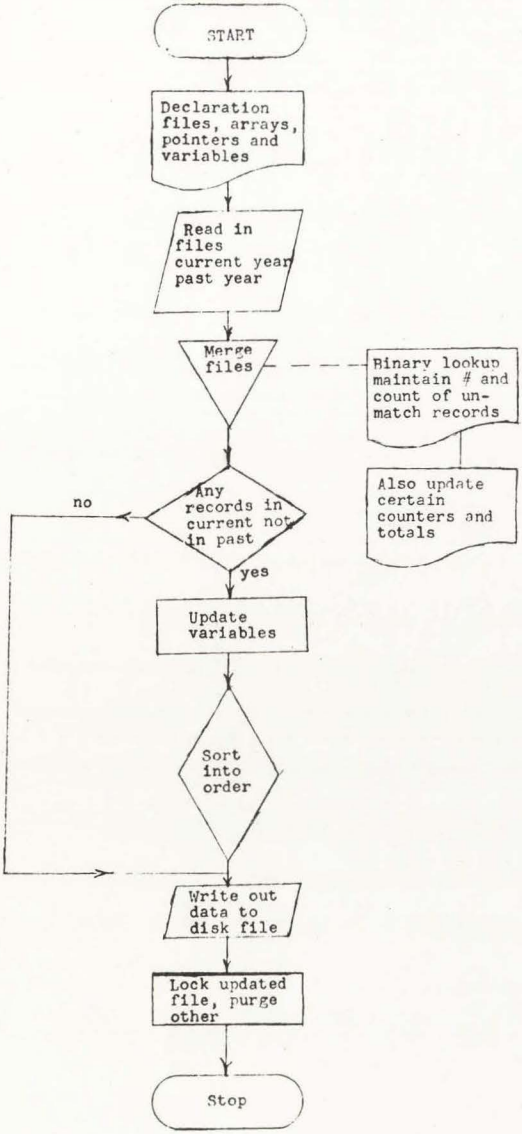


Figure 7. PERFORMANCE/MPPA/MERGE PROGRAM FLOW CHART

VARIABLE	I/O STATUS	INFORMATION	FORMAT SPECIFICATION
A	I/O	Array (2 dimensional) containing data from previous year	Machine relative
B	I/O	Array (2 dimensional) containing this years MPPA data	Machine relative
I	I/O	Number of previous records	I4
J	I/O	Number of records in this years record	I4
F	I/O	Herd total wean weight ratio for this year	F10.0

P E R F O R M A N C E / M P P A / M E R G E

```
$ SET LINEINFO
$ SET SEQ 50 * 50
BEGIN

 FILE LYMPA (KIND=DISKPACK, FILETYPE=7);

 FILE MPPA1 (KIND=DISKPACK, FILETYPE=7);

 DEFINE P=POINTER#;
 ARRAY A[0:1499,0:4], B[0:999,0:2], D[0:4];
 INTEGER ARRAY C[0:499];
 INTEGER I, J, K, L, IMIN, IMAX, MM, N;
 INTEGER ###;    REAL F;
 LABEL NEW, UPD, UPD1, UPD2, LESS, GRT, TRY, EL1, EL3, PRINT;
 POINTER PO;
 TO := POINTER(0);
%
%       READ IN THE FILES TO BE MATCHED
%
   READ (LYMPA,*[0:*]);
   READ (MPPA1,*[*,0:10-5], J, F);
   FOR K := 1 STEP 1 UNTIL I DO READ (LYMPA, D-AIN-1,*);
   FOR K := 1 STEP 1 UNTIL J DO READ (MPPA1, 5, B[K-1,*]);
%
%       MATCH ROUTINE STARTS HERE
%
   FOR L := 0 STEP 1 UNTIL J-1 DO
    BEGIN
     IF POINTER(B[L,0]) = " " FOR 6 THEN GO TO EL1;
     IMIN := 0;
     IMAX := I - 1;
NEW: K := (IMIN + IMAX) / 2;
%
%       TEST SECTION TO GET PROPER RECORD
%
     IF POINTER (A[K,0]) = POINTER (B[L,0]) FOR 6 THEN GO TO UPD;
     IF POINTER (A[K,0]) <= POINTER (B[L,0]) FOR 6 THEN GO TO LESS;
     IF POINTER (A[K,0]) >= POINTER (B[L,0]) FOR 6 THEN GO TO GRT;
LESS: IF (IMAX - IMIN) <= 1 THEN GO TO TRY;
     IMIN := K;
     GO TO NEW;
GRT: IF (IMAX - IMIN) <= 1 THEN GO TO TRY;
     IMAX := K;
     GO TO NEW;
%
%       TRY  CHECKS TO SEE IF IT IS A NEW RECORD IF YES THEN
%       SET INDICATOR AND PLACE THE NUMBER IN A HOLDING AREA
%
TRY: IF POINTER(A[IMIN,0]) = POINTER(B[L,0]) FOR 6 THEN GO TO UPD1;
     IF POINTER(A[IMAX,0]) = POINTER(B[L,0]) FOR 6 THEN GO TO UPD2;
     C[MM] := L;
     MM := MM+1;
     GO TO EL1;
```

```
I
I       UPD AREAS ARE THE LOCATIONS WHERE COUNTERS ARE UPDATED
I
UPD: REPLACE POINTER(A(K,1)) BY (INTEGER(POINTER(A(K,1))+2)+1)
        FOR 2 DIGITS;
     SCAN P(A(K,1))+2 FOR M:6 UNTIL IS ".";
     IF M = 0 THEN M := 6 ELSE M:= 6 - M;
     WWW := INTEGER(POINTER(A(K,1))+2,M);
     WWW := * * INTEGER(POINTER(HIL,1)),3);
     REPLACE POINTER(A(K,1))+2 BY WWW FOR 6 DIGITS;
     REPLACE POINTER(A(K,3))+2 BY "0" FOR 1;
     GO TO EL1;
UPD1: REPLACE POINTER(A(IMIN,1)) BY (INTEGER(POINTER(A(IMIN,1))+2)+1)
        FOR 2 DIGITS;
     SCAN P(A(IMIN,1))+2 FOR M:6 UNTIL IS ".";
     IF M = 0 THEN M := 6 ELSE M := 6 - M;
     WWW := INTEGER(POINTER(A(IMIN,1))+2,M);
     WWW := * * INTEGER(POINTER(HIL,1)),3);
     REPLACE POINTER(A(IMIN,1))+2 BY WWW FOR 6 DIGITS;
     REPLACE POINTER(A(IMIN,3))+2 BY "0" FOR 1;
     GO TO EL1;
UPD2: REPLACE POINTER(A(IMAX,1)) BY (INTEGER(POINTER(A(IMAX,1))+2)+1)
        FOR 2 DIGITS;
     SCAN P(A(IMAX,1))+2 FOR M:6 UNTIL IS ".";
     IF M = 0 THEN M := 6 ELSE M := 6 - M;
     WWW := INTEGER(POINTER(A(IMAX,1))+2,M);
     WWW := * * INTEGER(POINTER(HIL,1)),3);
     REPLACE POINTER(A(IMAX,1))+2 BY WWW FOR 6 DIGITS;
     REPLACE POINTER(A(IMAX,3))+2 BY "0" FOR 1;
EL1: END;
I
I       IF ANY NEW RECORDS ARE TO BE ADDED THEN EXPAND AND
        PLACE AT END OF FILE PROCEED THEN TO SORT
I
   IF MM = 0 THEN GO TO PRINT;
   M := I;
   FOR K := 0 STEP 1 UNTIL MM DO
    BEGIN
     L := GET;
     REPLACE POINTER(A(M,0)) BY " " FOR 5 WORDS;
     REPLACE POINTER(A(M,0)) BY POINTER(HIL,0)) FOR 6;
     REPLACE POINTER(A(M,1))+1 BY "1" FOR 1;
     REPLACE POINTER(A(M,1))+2 BY POINTER(HIL,1)) FOR 3;
     REPLACE POINTER(A(M,3))+2 BY "0" FOR 1;
     M := M + 1;
    END;
   I := I + MM;
   FOR K := 0 STEP 1 UNTIL I-2 DO
    FOR L := K+1 STEP 1 UNTIL I-1 DO
     BEGIN
      IF POINTER(A(K,0)) <= POINTER(A(L,0)) FOR 6 THEN GO TO EL3;
      REPLACE PD BY POINTER(A(K,0)) FOR 25;
      REPLACE POINTER(A(K,0)) BY POINTER(A(L,0)) FOR 25;
      REPLACE POINTER(A(L,0)) BY PD FOR 25;
EL3:  END;
I
I       WRITE OUT THE FILE BACK TO DISK AND REMOVE THIS YEARS FILE
I
PRINT: WRITE (LYNPRA(0),<2I4,F10.0>, I, J, F);
   FOR J := 0 STEP 1 UNTIL I-1 DO
    BEGIN
```

```
00002500  003:005F:F2
00002550  003:005F:F2
00002600  003:005F:F2
00002650  003:005F:F2
00002700  003:006D:F0
00002750  003:006E:F2
00002800  003:006E:F2
00002850  003:006E:F2
00002900  003:006F:F2
00002950  003:007E:F2
00003000  003:007E:F4
00003050  003:007F:F2
00003100  003:007F:F5
00003150  003:007G:F3
00003200  003:007G:F5
00003250  003:007H:F5
00003300  003:007H:F5
00003350  003:007H:F5
00003400  003:007H:F5
00003450  003:007H:F1
00003500  003:007H:F2
00003550  003:007H:F5
00003600  003:007H:F3
00003650  003:007H:F5
00003700  003:007H:F1
00003750  003:007H:F3
00003800  003:007H:F5
00003850  003:007H:F5
00003900  003:007H:F1
00003950  003:007H:F2
2 00004000  003:007H:E5
00004050  003:007H:E5
00004100  003:007H:E5
00004150  003:007H:E5
00004200  003:007H:E5
00004250  003:007H:F3
00004300  003:007H:E3
00004350  003:007H:E3
2 00004400  003:007H:E3
00004450  003:007H:E5
00004500  003:007H:E3
00004550  003:007H:F0
00004600  003:007H:F2
00004650  003:007H:F2
00004700  003:007H:E2
00004750  003:007H:F4
2 00004800  003:007H:F1
00004850  003:007H:F4
00004900  003:007H:F4
00004950  003:007H:F2
2 00005000  003:007H:F2
00005050  003:007H:F5
00005100  003:007E:F0
00005150  003:007E:F3
00005200  003:007E:F0
2 00005250  003:007E:F0
00005300  003:007E:F0
00005350  003:007E:F0
00005400  003:007E:F0
00005450  003:007E:F2
00005500  003:007F:F1
```

```
    WRITE(LYMPPA[J+1],5,A[J+1])                                  2   00005550  003102FA11
    END;                                                             00005600  003102FC12
  LOCK (LYMPPA);                                                 2   00005650  003102FC15
  CLOSE (MPPA1, PURGE);                                              00005700  003102FE12
END.                                                                 00005750  003102FF15
                                                                  A.00000003) IS 0114 LONG
                                                                        DATA IS 0008 LONG
*****************************************************************************************
NUMBER OF ERRORS DETECTED = 0.
NUMBER OF SEGMENTS = 4.  TOTAL SEGMENT SIZE = 297 WORDS.  CORE ESTIMATE = 3142 WORDS.  STACK ESTIMATE = 23
PROGRAM SIZE = 116 CARDS, 1092 SYNTACTIC ITEMS, 29 DISK SEGMENTS.
PROGRAM FILE NAME: PERFORMANCE/MPPA/MERGE.
COMPILATION TIME = 10.178 SECONDS ELAPSED; 1.550 SECONDS PROCESSING; 2.779 SECONDS I/O.
*****************************************************************************************
```

SECTION VI

PERFORMANCE/MPPA/MPPA

MPPA is the most probable producing ability. This program performs the calculation of the actual MPPA score of the individual animal that has produced a calf during the past year.

Input by default is from diskpack file created by either P/M/MERGE or P/M/IMPPAL. Each file contains a first record with total records in the file, number of cows producing a calf that year, and the herd total wean weight ratio, the remaining records contain dam number, calves in each individual record, individual total wean weight ratio, previous MPPA score, and a variable indicating calf produced (0=had, 1=not). Card input is also needed to give the rancher's name for the print-out along with the number of print-outs desired (0=1, with 9 being the maximum).

The print-out contains a heading (rancher's name, etc.), followed by column headings (dam number, number of calves, total wean weight ratio, MPPA score, previous MPPA, QDAM number, and QMPPA), followed by data results double spaced. The right side of the print-out is an area used for ranking the animals best to worst.

The data is rewritten to the disk with those producing calves having the previous MPPA score replaced by this year's MPPA score.

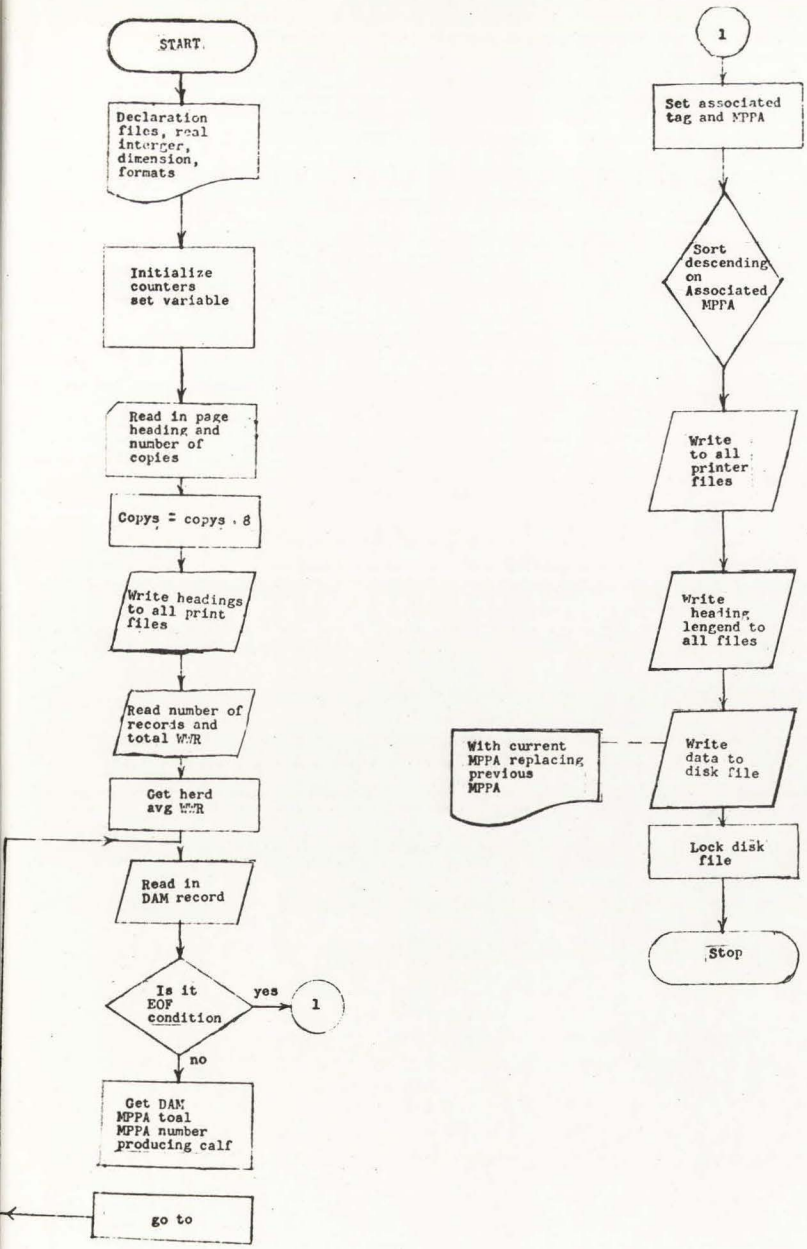


Figure 8. PERFORMANCE/MPPA/MPPA PROGRAM FLOW CHART

VARIABLE	I/O	INFORMATION	FORMAT SPECIFICATION
COPYS	I	Number of copys of printout desired	I1
HEAD	I/O	Page heading entered at run time	12A6
CALF	I	Number of calves or Dam's in the test	I4
HTWWR	I	Herd total wean weight ratio	F10.0
ITAG	I/O	Dam number	A6
ICALF	I/O	Number of calves in Dam Record	I2
AWWR	I/O	Dam total wean weight ratio	F5.0
PMPPA	I/O	Dam's previous MPPA score	F6.2
IPC	I/O	Production indicator (0=produced[A] calf, 1=did not produce a calf this year)	I1
TMPPA	O	Herd average MPPA for those producing calves	F6.2
NP	O	Number of dams producing calves	J4
WORD	O	Production status heading	6A6
MPPA	O	MPPA score for Dam	F6.2
DPC	I	number cows producing calves	I4

Figure 9. CARD LAYOUT PERFORMANCE/MPPA/MPPA

RANCHER X 1976

AVERAGE MPPA OF THOSE DAMS PRODUCING CALVES IS 48.34
WITH 35 DAMS PRODUCING CALVES.

THE RIGHT SIDE OF THE PRINTOUT IS SEPERATE SECTION BEARING AN ASSOCIATED TAG AND MPPA BEING RANKED BY MPPA FROM BEST TO WORST.

RANCHER X 1975

CALVES PRODUCED THIS YEAR					I	RANK BY MPPA	
DAMNO	NCALF	AWAR	MPWA	PMPPA	I	QDAMNO	QMPPA
337	1	145.	59.06	0.00	I	3	41.90
336	1	162.	65.86	0.00	I	28	39.06
340	1	138.	56.26	0.00	I	110	38.25
347	1	128.	52.26	0.00	I	330	37.68
301	2	178.	51.62	32.37	I	150	37.05
813	2	158.	45.90	35.17	I	84	35.46
851	2	156.	45.35	33.37	I	198	35.06
852	2	173.	50.78	39.97	I	228	29.46

HANGMER X 1976

DAMS NOT PRODUCING A CALF | RANK BY MPPA

DAMNO	NCALF	AWWR	MPPA	PMPPA	I	QDAMNO	QMPPA
5	1	72.	0.00	33.97	1	220	0.00
20	1	74.	0.00	33.17	1	224	0.00
34	1	126.	0.00	53.97	1	25	0.00
25	1	135.	0.00	56.77	1	230	0.00
213	1	115.	0.00	48.77	1	233	0.00
220	1	141.	0.00	59.17	1	437	0.00
224	1	127.	0.00	53.57	1	5	0.00
230	1	198.	0.00	45.97	1	20	0.00
233	1	126.	0.00	53.97	1	213	0.00
437	1	87.	0.00	37.57	1	24	0.00
921	1	72.	0.00	31.57	1	921	0.00
M-26	1	78.	0.00	33.97	1	M-26	0.00
M-28	1	87.	0.00	37.57	1	M-28	0.00

HEADING LEGEND:

DAMNO=DAM NUMBER

NCALF=NUMBER OF CALVES IN DAM RECORD

AWWR=TOTAL ADJUSTED WEAN WEIGHT RATIO FOR DAM

MPPA=MOST PROBABLE PRODUCING ABILITY

PMPPA=PREVIOUS YEAR MPPA

ODAMNO=ASSOCIATED DAM NUMBER

OMPPA=ASSOCIATED MPPA OF DAM

```
      H6700/H7700   F O R T R A N   C O M P I L A T I O N   M A R K   2.7.490  THURSDAY, 04/08/76    08:27 PM

 $ SET LINEINFO                                                                  C  0001000015
 $ SET SEG '0 + 50                                                               C  0001000025
FILE  5=PPPA2/DATAIN                                                             C0000100  C
                                                                                            FIB IS 0008 LONG
FILE  7(KIND=DISKPACK,FILETYPE=7,TITLE="LYMPPA")                                 C0000150  C
FILE  10=FILE10, UNIT=PRINTER                                                    C0000200  C
                                                                                            FIB IS 0007 LONG
FILE  11=FILE11, UNIT=PRINTER                                                    C0000250  C
                                                                                            FIB IS 0007 LONG
FILE  12=FILE12, UNIT=PRINTER                                                    C0000300  C
                                                                                            FIB IS 0007 LONG
FILE  13=FILE13, UNIT=PRINTER                                                    C0000350  C
                                                                                            FIB IS 0007 LONG
FILE  14=FILE14, UNIT=PRINTER                                                    C0000400  C
                                                                                            FIB IS 0007 LONG
FILE  15=FILE15, UNIT=PRINTER                                                    C0000450  C
                                                                                            FIB IS 0007 LONG
FILE  16=FILE16, UNIT=PRINTER                                                    C0000500  C
                                                                                            FIB IS 0007 LONG
FILE  17=FILE17, UNIT=PRINTER                                                    C0000550  C
                                                                                            FIB IS 0007 LONG
                                                                                          START OF SEGMENT 002
      INTEGER COPYS, TCALF(1500), PROD, CALF                                     C0000600  C  0021000010
      REAL ITAS(1500), MPPA(1500)                                                C0000650  C  0021000010
      DIMENSION STAG(1500), CMPPA(1500), AWWR(1500),                             C0000700  C  0021000010
     *REAL(2), PMPPA(1500), WORD(6), IPC(1500), LIPC(1500)                        C0000750  C  0021000010
    1 FORMAT(T2,12A6)                                                            C0000800  C  0021000010
    5 FORMAT(T1,4A,12A6)                                                         C0000850  C  0021000010
    7 FORMAT (6I4,F10.0)                                                         C0000900  C  0021000010
   10 FORMAT(T5,"  DAMNO  NCALF     AWWR     MPPA  PMPPA   I  COAMNO    OM C0000950  C  0021000010
     *PPA")                                                                      C0001000  C  0021000010
   12 FORMAT(A6,I2,X,F5.0,X,F5.2,I1)                                             C0001050  C  0021000010
   14 FORMAT("0",X,A6,A3,I2,X,F7.0,X,F7.2,F7.2,3X,"I",2X,A6,2X,F7.2)             C0001100  C  0021000010
   15 FORMAT("0",1X,6A5,2X,"I",2X,"RANK BY MPPA")                                C0001150  C  0021000010
   16 FORMAT(A6,I2,F6.0,F6.2,I1)                                                 C0001200  C  0021000010
   20 FORMAT("1",5X,"HEADING LEGEND",/,10X,                                      C0001250  C  0021000010
     *"DAM NUMBER",/,10X,"NCALF=NUMBER OF CALVES IN DAM RECORD",/,               C0001300  C  0021000010
     *10X,"AWWR---TOTAL ADJUSTED WEAN WEIGHT RATIO FOR DAM",/,10X,               C0001350  C  0021000010
     *"PMPPA=PRODUCTIVE PRODUCING ABILITY",/,10X,                                C0001400  C  0021000010
     *"PMPPA=PREVIOUS YEAR MPPA",/,10X,"COAMNO=ASSOCIATED DAM NUMBER",/,         C0001450  C  0021000010
     *10X,"CMPPA=ASSOCIATED MPPA OF DAM")                                        C0001500  C  0021000010
   21 FORMAT("0",1X,"AVERAGE MPPA OF THOSE DAMS PRODUCING CALVES IS ",           C0001550  C  0021000010
     * F6.2,/,/,"0IN ",J4," DAMS PRODUCING CALVES.",/,/,"0",1X,                   C0001600  C  0021000010
     * "THE RIGHT SIDE OF THE PRINTOUT IS SEPERATE SECTION BEARING AN AS          C0001650  C  0021000010
     *SOCIATED TAG AND MPPA BEING",/,"0",1X,"RANKED BY MPPA FROM BEST TO          C0001700  C  0021000010
     * WORST.")                                                                   C0001750  C  0021000010
      IMPPA = 0                                                                  C0001800  C  0021000010
      NP = 0                                                                     C0001850  C  0021000014
C                                                                                C0001900  C  0021000112
C     READ IN NUMBER OF PRINTOUT COPYS AND HEADER FOR PAGE                       C0001950  C  0021000112
C     PLUS INFORMATION TO GET HERD AVERAGE ADJUSTED WEIGHT TOTAL                 C0002000  C  0021000112
C     RECORDS.  THEN START READING IN RECORDS AND FIGURING MPPA                  C0002050  C  0021000112
C     SCORE ON THOSE PRODUCING A CALF.                                           C0002100  C  0021000112
C                                                                                C0002150  C  0021000112
      READ(5,1) COPYS, HEAD                                                      C0002200  C  0021000112
      COPYS = COPYS + 5                                                          C0002250  C  0021000112
      READ(7,7) CALF, OPC, HTWWR                                                 C0002300  C  0021000115
      HAWWR = HTWWR / OPC                                                        C0002350  C  0021001112
      DO 100 MR = 1, CALF                                                        C0002400  C  0021001114
```

```
      READ(7,12,END=200) ITAG(NR), TCALF(NR), AWWR(NR),PMPPA(NR),
     * IPC(NR)
      IF (IPC(NR). EQ .1) GO TO 100
      CA = AWWR(NR) / TCALF(NR)
      A = (TCALF(NR) - .4) / (1 + ((TCALF(NR) - 1) * .4))
      MPPA(NR) = HAWWR + (A * (CA - HAWWR))
      TMPPA = TMPPA + MPPA(NR)
      NR = NR + 1
  100 CONTINUE
  200 NR = CALF
C
C     SET ASSOCIATED INFORMATION AND SORT DATA FOR RANKING AREA
C
      DO 202 I = 1, NR
      LIPC(I) = IPC(I)
      ITAG(I) = ITAG(I)
  202 SMPPA(I) = MPPA(I)
      DO 210 I = 1, NR-1
      DO 210 J = I+1, NR
      IF(SMPPA(I). GE .SMPPA(J)) GO TO 210
      ITP = LIPC(J)
      TT = ITAG(J)
      TV = SMPPA(J)
      LIPC(J) = LIPC(I)
      ITAG(J) = ITAG(I)
      SMPPA(J) = SMPPA(I)
      LIPC(I) = ITP
      ITAG(I) = TT
      SMPPA(I) = TV
  210 CONTINUE
C
C     SET FIRST PRODUCTION GROUP HEADING
C
      *PRO(1) = ' CALF'
      *PRO(2) = 'FES PROD'
      *PRO(3) = 'UCED '
      *PRO(4) = 'THIS Y'
      *PRO(5) = 'EAR '
      *PRO(6) = '      '
      *PRO(7) = '      '
      TMPPA = TMPPA / FLOAT(NR)
C
C     PRINT THE AVERAGE OF THOSE PRODUCING A CALF
C
      WRITE(6,1) HEAD
      WRITE(6,21) TMPPA, NR
      IF(NPRT. EQ .0) GO TO 220
      DO 221 I = 1, COPYS
  221 WRITE(6,21) TMPPA, NR
  220 LINE = 15
      JL = 0
C
C     GENERATE PRINTOUT OF INDIVIDUAL RECORDS FOR THE NO. OF COPYS
C
      DO 300 I = 1, NR
      IF (NPRT. NE .IPC(I)) GO TO 300
      JL = JL + 1
   70 IF ( LIPC(JL). EQ .PROD) GO TO 75
      JL = JL + 1
      GO TO 70
```

```
C0002450  C  0021001A10
C0002500  C  0021002B14
C0002550  C  0021002F10
C0002600  C  0021003015
C0002650  C  0021003313
C0002700  C  0021003410
C0002750  C  0021003711
C0002800  C  0021003F11
C0002850  C  0021004033
C0002900  C  0021004214
C0002950  C  0021004433
C0003000  C  0021004533
C0003050  C  0021004733
C0003100  C  0021004833
C0003150  C  0021004910
C0003200  C  0021004712
C0003250  C  0021004814
C0003300  C  0021004E11
C0003350  C  0021005033
C0003400  C  0021005413
C0003450  C  0021005514
C0003500  C  0021005611
C0003550  C  0021005814
C0003600  C  0021005811
C0003650  C  0021005833
C0003700  C  0021005F15
C0003750  C  0021006011
C0003800  C  0021006315
C0003850  C  0021006333
C0003900  C  0021006511
C0003950  C  0021006611
C0004000  C  0021006711
C0004050  C  0021006811
C0004100  C  0021006915
C0004150  C  0021006E11
C0004200  C  0021006E11
C0004250  C  0021007011
C0004300  C  0021007411
C0004350  C  0021007511
C0004400  C  0021007611
C0004450  C  0021007711
C0004500  C  0021007711
C0004550  C  0021007711
C0004600  C  0021007711
C0004650  C  0021007711
                 THIS IS 0006 LONG
C0004700  C  0021007712
C0004750  C  0021007812
C0004800  C  0021007814
C0004850  C  0021007E11
C0004900  C  0021008011
C0004950  C  0021008112
C0005000  C  0021008210
C0005050  C  0021008210
C0005100  C  0021008310
C0005150  C  0021008310
C0005200  C  0021008310
C0005250  C  0021008510
C0005300  C  0021008612
C0005350  C  0021008812
C0005400  C  0021008914
```

```
   75 IF (LINE. LT .25) GO TO 310
      LINE = 1
      WRITE(6,3) HEAD
      WRITE(6,15) WORD
      WRITE(6,10)
      IF(COPYS. LE .9) GO TO 310
      DO 316 J = 10, COPYS
      WRITE(J,3) HEAD
      WRITE(J,15) WORD
  316 WRITE(J,10)
  310 WRITE(6,14) ITAG(I), TCALF(I), AWWR(I), MPPA(I), PMPPA(I),
     * QTAC(JL), QMPPA(JL)
      IF(COPYS. LE .9) GO TO 320
      DO 325 K = 10, COPYS
  325 WRITE(K,14) ITAG(I), TCALF(I), AWWR(I), MPPA(I), PMPPA(I),
     * QTAC(JL), QMPPA(JL)
  320 LINE = LINE + 1
  300 CONTINUE
C
C      CHANGE PRODUCTION TO AND GROUP HEADING REPEAT PRINTOUT AREA
C
      PROD = PROD + 1
      IF ( PROD. EQ .2) GO TO 340
      WPRO(1) = 'SAMS AV'
      WPRO(2) = 'ION PRO'
      WPRO(3) = 'DUCING'
      WPRO(4) = ' A CAL'
      WPRO(5) = 'F      '
      GO TO 290
C
C      WRITE HEADING LEGENDING DECODE PAGE
C
  340 WRITE(6,20)
      IF(COPYS. LE .9) GO TO 350
      DO 351 KK = 10, COPYS
  351 WRITE(KK,20)
C
C      RETURN INFORMATION TO DISK UPDATING PMPPA SCORE ON THOSE
C      THAT PRODUCED A CALF
C
  350 DO 360 M = 1, NR
      IF(IPC(M). EQ .0) GO TO 365
      WRITE(7+M+1,16) ITAG(M), TCALF(M), AWWR(M), PMPPA(M), IPC(M)
      GO TO 360
  365 WRITE(7+M+1,16) ITAG(M), TCALF(M), AWWR(M), MPPA(M), IPC(M)
  360 CONTINUE
      LOCK 7
      STOP
      END
```

```
00005450  C  002:004431
00005500  C  002:004433
00005550  C  002:004431
00005600  C  002:004432
00005650  C  002:004432
00005700  C  002:004432
00005750  C  002:004434
00005800  C  002:004410
00005850  C  002:004002
00005900  C  002:004032
00005950  C  002:003E03
00006000  C  002:005810
00006050  C  002:010234
00006100  C  002:010234
00006150  C  002:010520
00006200  C  002:011433
00006250  C  002:012333
00006300  C  002:012435
00006350  C  002:012710
00006400  C  002:012710
00006450  C  002:012710
00006500  C  002:012710
00006550  C  002:012712
00006600  C  002:012712
00006650  C  002:012E31
00006700  C  002:012E31
00006750  C  002:013131
00006800  C  002:013431
00006850  C  002:013731
00006900  C  002:013734
00006950  C  002:013734
00007000  C  002:013734
00007050  C  002:013734
00007100  C  002:013732
00007150  C  002:013734
00007200  C  002:013E10
00007250  C  002:014013
00007300  C  002:014013
00007350  C  002:014013
00007400  C  002:014013
00007450  C  002:014013
00007500  C  002:014E10
00007550  C  002:014F15
00007600  C  002:016112
00007650  C  002:016115
00007700  C  002:017312
00007750  C  002:017513
00007800  C  002:017710
00007850  C  002:017715
        SEGMENT 002 IS 019D LONG
```

```
                                                    FORMAT SEGMENT IS 0009 LONG
                                                         START OF SEGMENT 006
                                                    SEGMENT 006 IS 0040 LONG

NO ERRORS DETECTED.  NUMBER OF CARDS = 159.
COMPILATION TIME = 14 SECONDS ELAPSED.  2.29 SECONDS PROCESSING.
02 STACK SIZE = 27 WORDS.  FILESIZE = 764 WORDS.  ESTIMATED CORE STORAGE REQUIREMENT = 16930 WORDS.
TOTAL PROGRAM CODE = 582 WORDS.  ARRAY STORAGE = 13518 WORDS.
NUMBER OF PROGRAM SEGMENTS = 6.  NUMBER OF DISK SEGMENTS = 60.
PROGRAM CODE FILE = (8091005173)PERFORMANCE/MPPA/MPPA. COMPILER COMPILED ON 12/18/75(FORTRAN ON PACK)
```

SECTION VII

PERFORMANCE/MPPA/CARDP

This program punches to cards the updated date from the P/M/MPPA, segment of the MPPA system. Also, it punches a card deck if a file has been created by the P/W/Term1 (same as P/WEAN in scope, but runs from teletype, currently is not operational).

Input is from diskpack file(s) which is specific in its use and scope. The MPPA file contains dam number, total calves produced, total wean weight ratio, and the updated previous MPPA score. If present the wean file holds the information which when punched to cards will be able to become input to the P/WEAN program. Control character is by card input (title is control/datain).

Output is to cards in a form required of its use. MPPA data is two (2) records per card; the wean file puts one (1) record per card in a format needed as card input to the weaning program. Upon finishing the punching, the files are purged from the computer system network. The files are named LYMPPA for te MPPA data and HERD for the weaning date.

The complete cattle record system was programmed and placed in operation, limited by the physical size of the computer system at the time of programming. It is currently in operation on the Burroughs 6700 computer at the Utah State University campus. This record system was made possible by first modifying the calf testing programs to supply needed information, followed then by the actual programming of the MPPA testing routines.

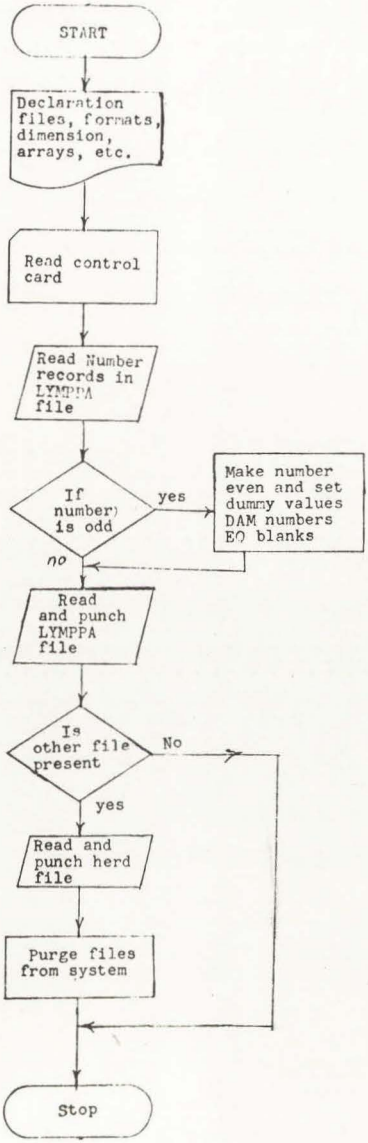


Figure 10. PERFORMANCE/MPPA/CARDP PROGRAM FLOW CHART

VARIABLE	I/O STATUS	INFORMATION	FORMAT SPECIFICATION
A	I/O	Array (2 dimensional) to contain the information[1]	Input A6, I2, F5.0, F6.2 Output A6, I2, F6.0, F6.2
B	I/O	Array (1 dimension- to hold information from P/W/terml	A5, 3I2, I3, I4, I3, A6, I2
I	I	Control indicator of file presence	I1
J	I	Total records of MPPA data	I4

[1]Tab to column forty (40) and repeat thus obtaining two (2) records per card, on output.

Figure 11. CARD LAYOUT PERFORMANCE/MPPA/CARDP

```
B6700/P7700   F O R T R A N   C O M P I L A T I O N   M A R K  2.7.480  THURSDAY, 04/08/76    08:12 PM

 $ SET LINEINFO                                                          C  000:0000:5
 $ SET SEQ 50 + 50                                                 C0000050  C  000:0000:5
FILE  5=CONTROL/DATAIN                                             C0000100  C
                                                                             FIB IS 0008 LONG
FILE  7(KIND=DISKPACK,FILETYPE=7,TITLE="LYMPPA")                   C0000150  C
FILE  8(KIND=DISKPACK,FILETYPE=7,TITLE="HERO")                     C0000200  C
FILE  11=FILE11, UNIT=PUNCH                                        C0000250  C
                                                                             FIB IS 0007 LONG
                                                                           START OF SEGMENT 002
      DIMENSION A(1500,4), B(13)                                   C0000300  C
    1 FORMAT(T4)                                                   C0000350  C
    5 FORMAT (A5,I2,X,F5.0,X,F5.2)                                 C0000400  C
   21 FORMAT(A6,I2,F6.0,F6.2,T40,A6,I2,F6.0,F6.2)                  C0000450  C
   22 FORMAT (I1)                                                  C0000500  C
   31 FORMAT(A5,2X,I1,3I2,I3,3I2,I3,I3,A6,I2)                      C0000550  C
C                                                                  C0000600  C
C       GET  CONTROL INDICATOR  CHECK TO SEE IF MPPA CONTAINS AN   C0000650  C
C       EVEN NUMBER OF RECORDS AFTER FIRST READING HOW MANY IN FILE C0000700  C
C                                                                  C0000750  C
      READ(5,22) I                                                 C0000800  C
      READ (7,1) J                                                 C0000850  C
      K = MOD(J,2)                                                 C0000900  C
      DO 10 K = 1, J                                               C0000950  C
   10 READ (7,5) (A(K,L), L=1,4)                                   C0001000  C
      IF (K. EQ .0) GO TO 15                                       C0001050  C
      J = J + 1                                                    C0001100  C
      A(J,1) = *       *                                           C0001150  C
      DO 35 K = 2, 4                                               C0001200  C
   35 A(J,K) = 0                                                   C0001250  C
C                                                                  C0001300  C
C       PUNCH MPPA DATA TO CARDS                                   C0001350  C
C                                                                  C0001400  C
   15 DO 20 K = 1, J, 2                                            C0001450  C
   20 WRITE (11,21) (A(K,L), L=1,4), (A(K+1,M), M=1,4)             C0001500  C
C                                                                  C0001550  C
C       IF HERO FILE PRESENT THEN PUNCH ELSE GO REMOVE FILES AND   C0001600  C
C       END THE EXECUTION OF THE PROGRAM                           C0001650  C
C                                                                  C0001700  C
      IF (I .LT. 1) GO TO 50                                       C0001750  C
   30 READ(8,31,END=50) (B(M), M=1,13)                             C0001800  C
      WRITE(11,31) (B(K), K=1,13)                                  C0001850  C
      GO TO 30                                                     C0001900  C
   50 PURGE 7                                                      C0001950  C
      PURGE 8                                                      C0002000  C
      STOP                                                         C0002050  C
      END                                                          C0002100  C
                                                                           SEGMENT 002 IS 006C LONG
```

```
                                                    FORMAT SEGMENT IS 0025 LONG
                                                        START OF SEGMENT 009
                                                    SEGMENT 009 IS 0014 LONG
NO ERRORS DETECTED.  NUMBER OF CARDS = 44.
COMPILATION TIME = 9 SECONDS ELAPSED.  1.72 SECONDS PROCESSING.
D2 STACK SIZE = 10 WORDS.  FILESIZE = 124 WORDS.  ESTIMATED CORE STORAGE REQUIREMENT = 6330 WORDS.
TOTAL PROGRAM CODE = 167 WORDS.  ARRAY STORAGE = 6013 WORDS.
NUMBER OF PROGRAM SEGMENTS = 9.  NUMBER OF DISK SEGMENTS = 24.
PROGRAM CODE FILE = (8091005 17)PERFORMANCE/MPPA/CARDP, COMPILER COMPILED ON 12/18/75(FORTRAN ON PACK)
```

APPENDIX

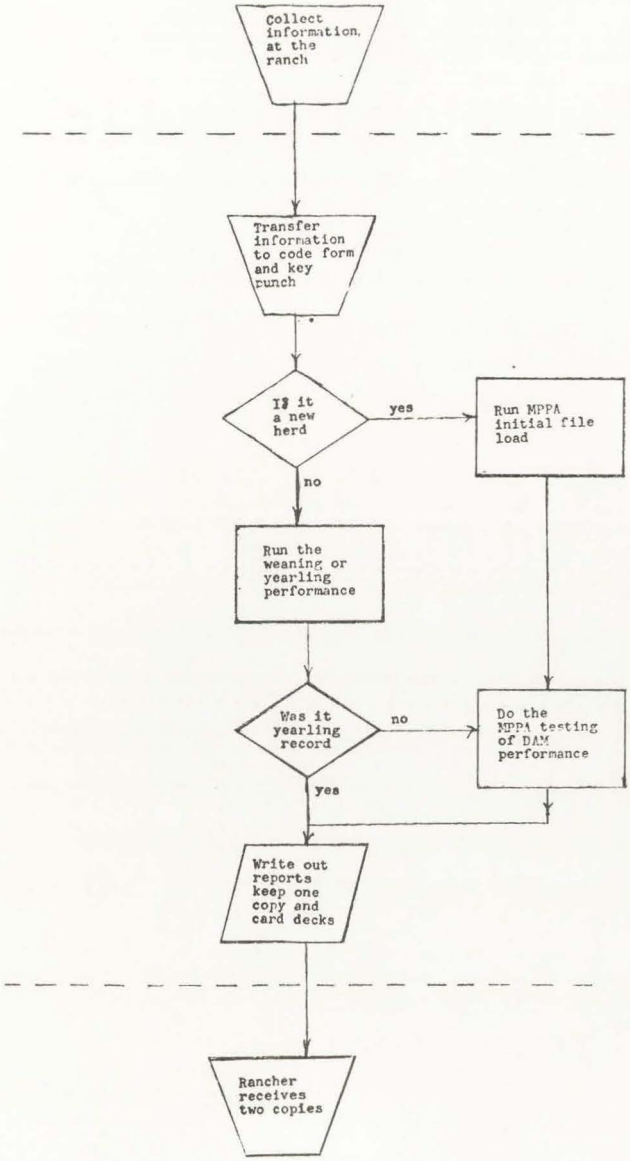


Figure 12. SYSTEM DATA FLOW

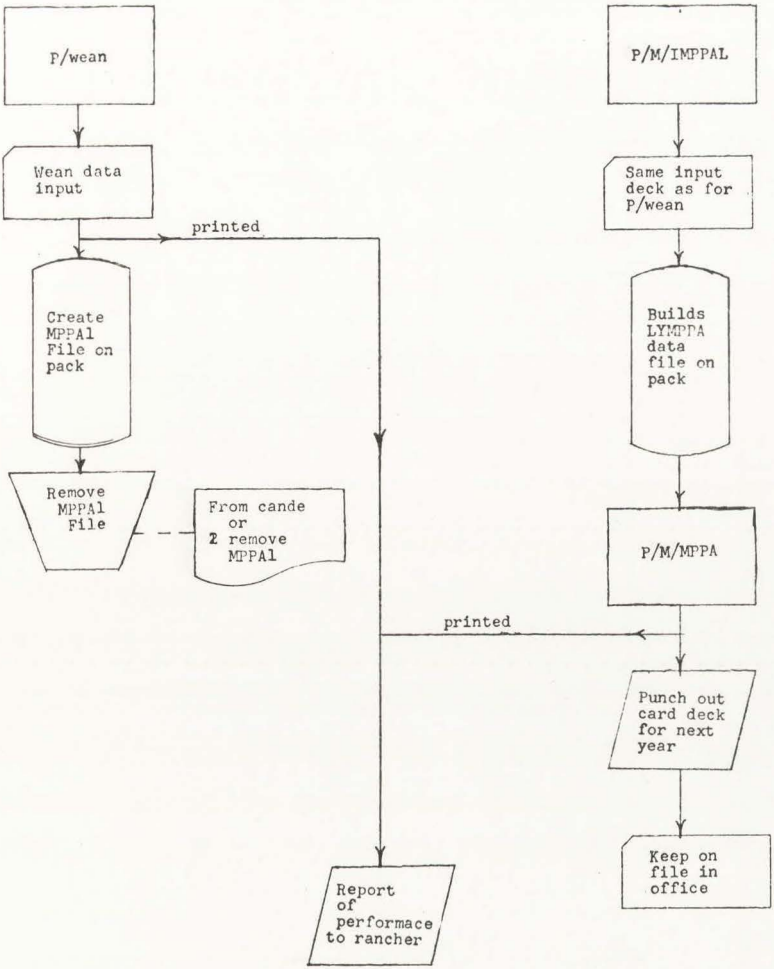


Figure 13. INITIAL HERD RUN

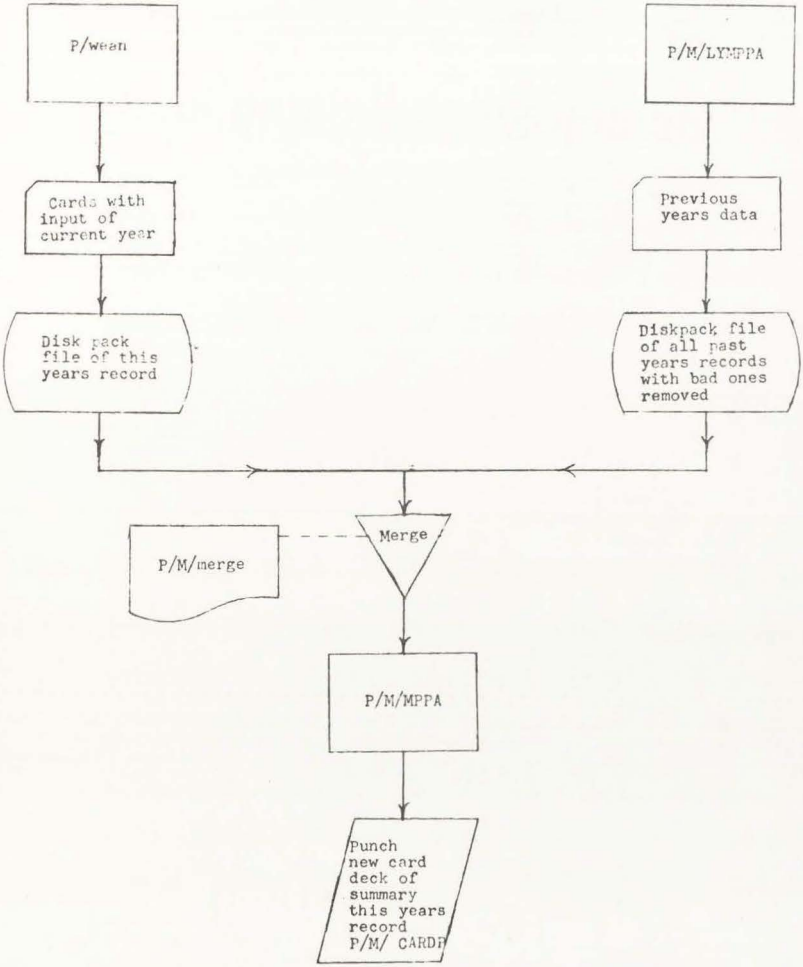


Figure 14. UPDATE A HERD

	WEANING DATA								YEARLING DATA			PROBLEMS, REMARKS
TAG	B.DATE	BWT	W.DATE	AWWT	CONFS	DAM NO.	DAGE	SIBENO	DATE	WT	CONF S	DAMS REMOVED

Figure 15. DATA COLLECTION FORM

PROGRAM	CARD	DISKPACK	PRINTER	PUNCH	REMOTE	PROGRAM LANGUAGE
P/WEAN	5=PERFORMANCE1 DATAIN 4=NOTES[1]	3=MPPA1[3,6]	6,&10-17			FORTRAN
P/WEAN/TERML[2]		3=MPPA[1] 8=HERD[4]	6		4,5,8 10-17[7]	FORTRAN
P/M/IMPPAL	5=LOADERI DATAIN	7=LYMPPA[5,6] 8=MPPA[1]				FORTRAN
P/M/LMPPA	PMPPA REMOVE	LYMPPA	LP			ALGOL
P/M/MERGE		LYMPPA MPPA1				ALGOL
P/M/MPPA	5=MPPA1 DATAIN	7=LYMPPA	6, 10-17			FORTRAN
P/M/CARDP	5=CONTROL/ DATAIN	7=LYMPPA 8=HERD		11=file 11		FORTRAN

1 All files are system default specifications unless noted otherwise.
2 This program is just in the planning stage.
3 Specifications are MAXRECSIZE=3, BLOCKSIZE=90, SECURITY TYPE=CLASS A.
4 MAXRECSIZE=14, BLOCKSIZE=420, SECURITY TYPE=CLASS A.
5 MAXRESIZE=5, BLOCKSIZE=150, FLIXIBLE=TRUE, SECURITY TYPE=CLASS A.
6 Specifications the same as previously listed, and has the same meaning in the programs where this title is listed.
7 myuse=OUT for files 10-17, MYUSE=I/O for files 4 and 5.
8 teletype communication.

```
$ END
4=47
3=CHL547
KEY TO SIRE CODE
$ DATA NOTES
```

NOTES DATA DECK

```
502  1 62875    111175385132    819 7 3
501  1 30875    111175385132    970 6 4
500  2 5.475    111175470142    624 9 3
03  RANCHER W WEAN 1976
$ DATA PERFORMANCE/DATAIN
$ RUN PERFORMANCE/WEAN
$ USER 0000000/PASSWORD
```

HERD DATA DECK

// METRIC , COPIES, AND PAGE HEADING

```
00''000'''''''''00'''0000000000000000000000000000000000000000000000000
1111111111111111111111111111111111111111111111111111111111111111111111
22'22'2222222222222'22'222222222222222222222222222222222222222222222222
3333333333333333333333333333333333333333333333333333333333333333333333
44'4444444444444444'444444444444444444444444444444444444444444444444444
5555'55555555555555555555555555555555555555555555555555555555555555555
66666666666666666666'''66666666666666666666666666666666666666666666666
7777777777777777'7777777777777777777777777777777777777777777777777777
88888888888888888888888888888888888888888888888888888888888888888888888
99999'999999999999999'999999999999999999999999999999999999999999999999
```

CLOINE 342 STANDARD FORM 5087

WEAN RUN WITH SPECIAL NOTES

```
$ END
$ DATA NOTES
502  1 63875   111175385132   819 7 3   310761050143
501  1 38875   111175385132   970 6 4   31076 900151   } DATA
500  2 51475   111175470142   624 9 3   31076 900152
03  RANCHER W YEARLING 1976          // CONTROL AND HEADING CARD
$ DATA PERFORMANCE/I4TAIN
$ RUN PERFORMANCE/MEAN
$ USER 0000000/PASSWORD
```

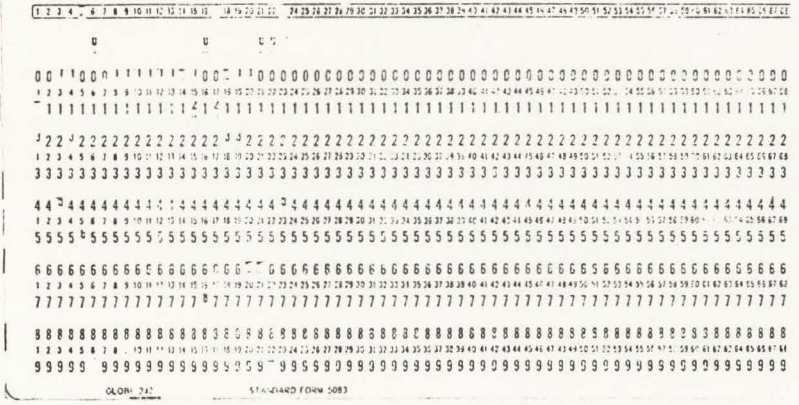
GLOBE 342 STANDARD FORM 5083

YEARLING RUN WITHOUT SPECIAL NOTES

```
$ END
  502  1 62875   111175325132.  819 7 3
  501  1 30875.  111175385132   970 6 4
  500  2 51475  .111175470142   624 9 3
00(T8,I1,2(3I3,I3),T31,A6,I2)
$ DATA LOADER/DATAIN
$ RUN PERFORMANCE/MPPA/IMPPAL
$ USER 0000080/PASSWORD
```

DATA DECK

// CONTROL AND FORMAT CARD

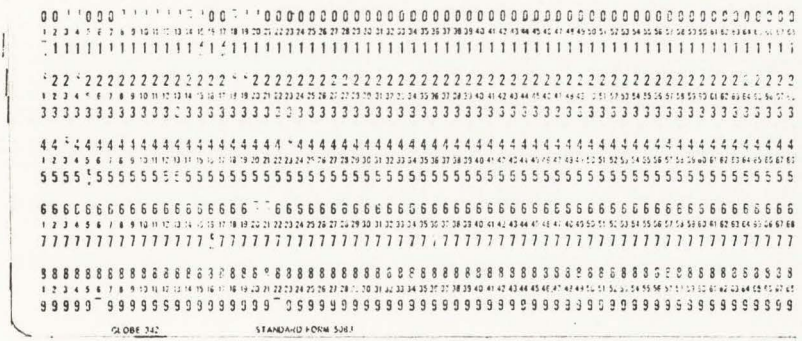


GLORI 242 STANDARD FORM 5083

RUN OF INITIAL LOAD INITIAL FILE

```
$ END
  502  1 62875    111175325132    819 7 3
  501  1 30875    111175385172    970 6 4
  500  2 51475    111175470142    624 9 3
01(T8,I1,2(3I2,I3),T31,A6,I2)          // control and format card
$ DATA LOADER/DATAIN
$ RUN PERFORMANCE/MPPA/IMPPAL
$ USER 0000000/PASSWORD
```

data deck



GLOBE 342 STANDARD FORM 5083

RUN OF INITIAL LOAD UPDATE PHASE

```
$ END
  817
  8.4                    REMOVE DATA DECK
  007
$ DATA REMOVE

                         PREVIOUS YEARS DATA DECK
  009 2  186. 82.33                    013 1   96. 94.13
  005 2  168. 76.81                    007 1   91. 77.26
$ DATA RMPPA
$ RUN PERFORMANCE/MPPA/LRMPPA
$ USER 0000000 PASSWORD
```

LOAD AND REMOVE PREVIOUS YEARS DATA

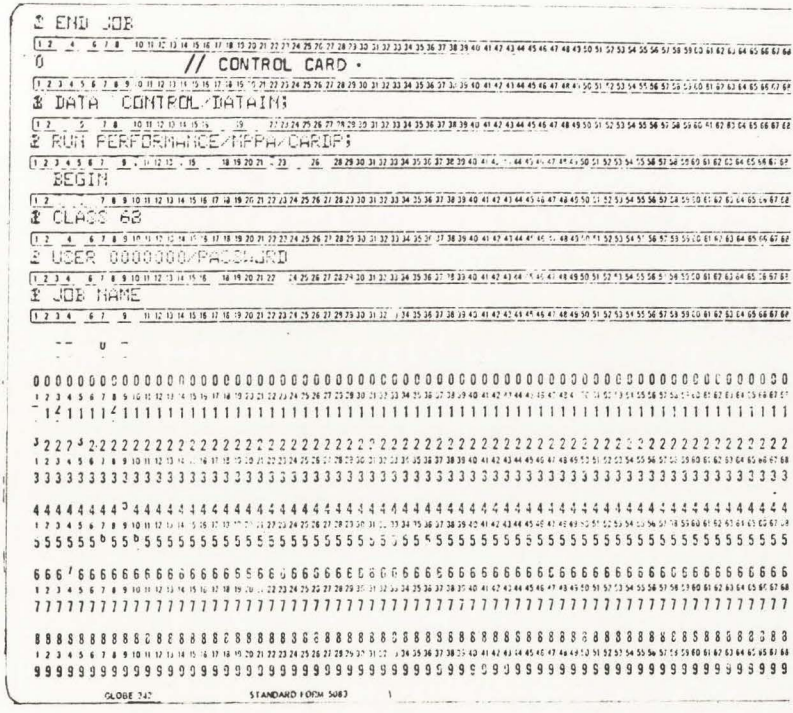
```
3 END
3 RUN PERFORMANCE/MPPA/MERGE
3 USER 0000000/PASSWORD
```

GLOBE 242 STANDARD FORM 5002

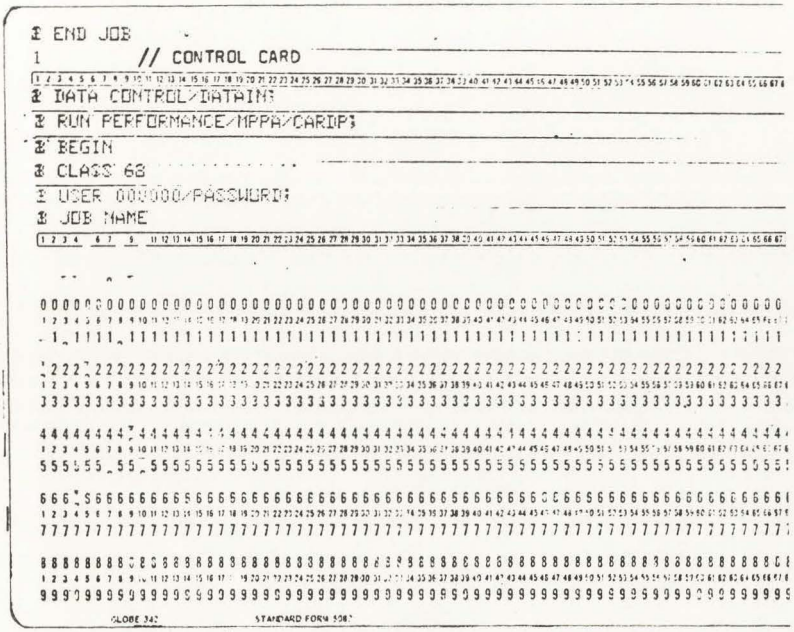
RUN OF MERGE ROUTINE

```
@ END
3 RANCHER X   MPPA   1976                    // COPIES AND PAGE HEADING CARD
@ DATA MPPA/DATAIN
@ RUN PERFORMANCE/MPPA/MPPA
@ USER 0000000/PASSWORD
```

RUN OF MPPA PROGRAM



CARD PUNCH WITH MPPA DATA ONLY



CARD PUNCH WITH BOTH FILES PRESENT