

AD-A160 995

DATA PREPARATION USING LOTUS AND OTHER SPREADSHEETS(U)
AIR FORCE LOGISTICS COMMAND WRIGHT-PATTERSON AFB OH
R E COOPER SEP 85

1/1

UNCLASSIFIED

F/G 9/2

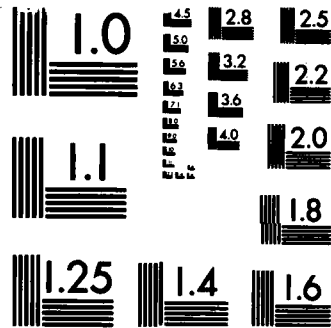
NL



END

FALSB

DTC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A160 995

①

Data Preparation Using LOTUS and Other Spreadsheets

by

**Ronnie E. Cooper
Cost Analyst
AFLC/ACMCE**

**Presented at the
19th Annual Department of Defense
Cost Analysis Symposium
Xerox Training Center
Leesburg, Virginia
September 17 - 20 1985**

DTIC FILE COPY

**DTIC
ELECTE
NOV 6 1985
S A D**

This document has been approved for public release and sale; its distribution is unlimited.

85 11 06 031

-A-

TABLE OF CONTENTS

- Introduction To Spreadsheet Use ;
- Distributed Processing - Someone Preprocesses The Needed Data ;
- Comparisons - Why Use a Spreadsheet or a Program ;
- Spreadsheet Preparation - General Principles ;
- Documentation - Organizing All That Information ;
- Data Types, Potential Errors ;
- Data Entry - What Does It Look Like ;
- How To Compute A Weighted Average ;
- Why And How To Structure Data ;
- Using Learning Curves ;
- Using Comparison Data ;
- Using CERs (Cost Estimating Relationships) ;
- What Is A Macro ;
- Macro Examples and Their Use In Menus ;
- The Future of the Concept ;
- Conclusion .

This material is new. It has not been presented or published elsewhere.

Accession No.	
NTIS CRAD	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Just	
On Form 50	
By	JK
Date	
Agency Codes	
Order Number or Special	
AI	



Introduction To Spreadsheet Use

When I was told this paper had been accepted for presentation, my boss said "You'll like the trip to the pyramid". I thought of the famed Egyptian ones first, then he told me of this beautiful structure. Although this is a great structure, I just hope it isn't just another burial place like my first thought. Our purpose here is to bring life to ideas, not bury them in our daily shuffle or the enjoyment of this wonderful retreat.

Today I've chosen the classic Henry Ford method. Take existing ideas and methods and integrate them into a new package; "Data Preparation using LOTUS and Other Spreadsheets". Most of us have great programs, but developing, reading or changing the data is a time consuming pain. If the changes are very extensive, we may even take computational shortcuts. Finally, when last minute changes occur, these changes can roll through all the data and require extensive rework.

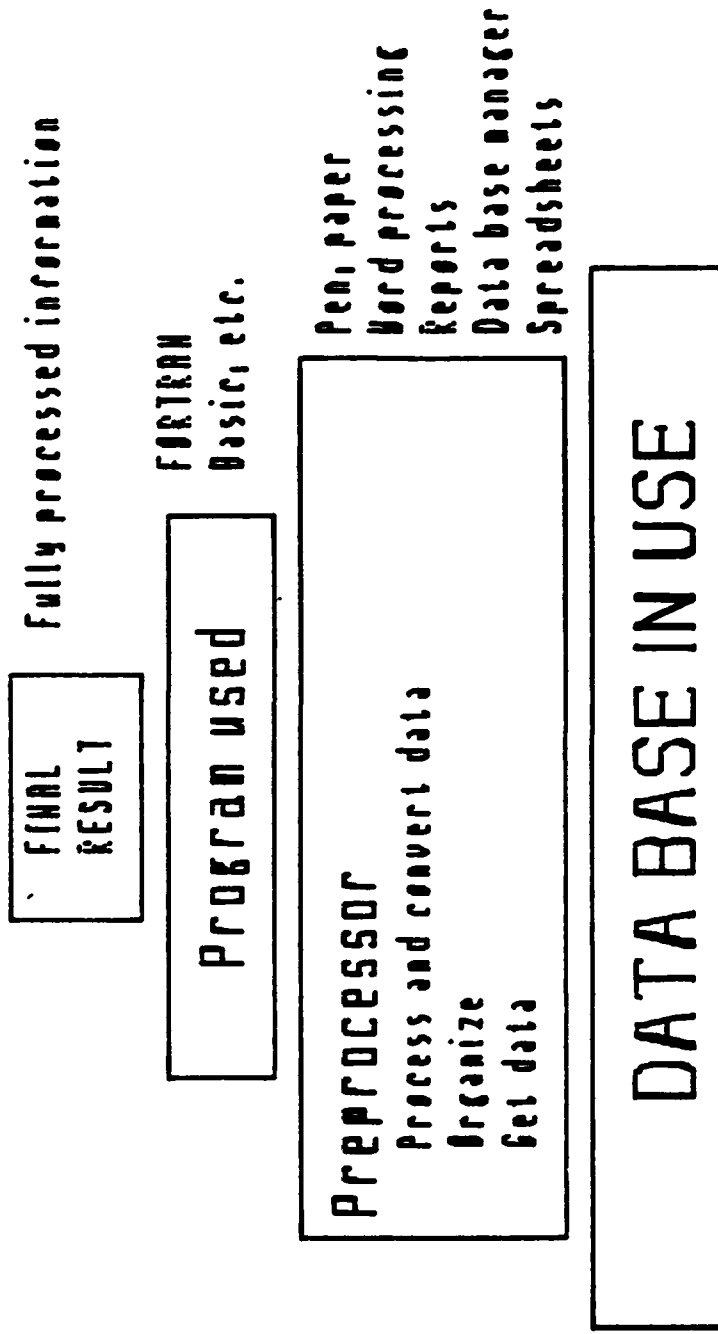
What I'm presenting could save you a lot of time with data preparation and improve your current program usage, but I expect mostly it will boil down to improved estimates. You'll use more time generating analysis instead of reinventing your data every time a delivery schedule changes or another what-if situation occurs. You'll save time, because a spreadsheet template saves you the effort of recalculating those what-if iterations. The copying capabilities and the ability to label your entries are other built-in aids to improved speed and accuracy in building good data files.

First, in this paper, we'll go over some general discussion of the process of preparing data and what to consider in using a spreadsheet. Then, we'll go over some specific spreadsheet examples of methods to solve common problems in computing data. Lastly, we'll go over the use of macros, preprogrammed instructions, and give some macro examples as used in LOTUS menus. All these steps will generally prepare you to perform a cost estimate in the most efficient manner possible.

Distributed Processing; Someone Preprocesses The Needed Data

The dual use of spreadsheets and your own custom program is really distributed processing. The people doing the inputs can produce the first stage results. In fact, they have more than just the results you wanted. They also have the data needed to produce your required inputs. Therefore, they can produce a historical data base using a data base program and a spreadsheet output modified for quote marks around text. Ideally, their existing data base can be sorted and used as direct input to a spreadsheet. The spreadsheet can then convert the data into the form needed by your program, the second stage of the processing.

Distributed Processing



- Pen, paper
- Word processing
- Reports
- Data base manager
- Spreadsheets

Each block is a step toward the final result.

Two stage processing, with spreadsheets for the first stage, not counting constructing the data base, should be more than just an exercise in inputting numbers. Usually, relationships between the data can be established by using smarter structuring of the data. Some easy ones are: organizing for locations which have known rates and averages, using common hardware groupings (frames, engines, radars, etc. which can have similar costs to other similar groups), and in general using consistent methods of being specific about the data you're working with. Simply put - compute the known relationships. Using a spreadsheet as assistance will reduce manual computation and keyboard inputs someone has to do to get the final data input used for a program.

The basic purpose of using a spreadsheet is to reduce the inputting task to raw data into labelled/defined blank cells. Then let the spreadsheet format the data into the proper format type and calculate the other needed results.

Sophisticated stage one processing (the spreadsheet) can use many methods, eg, learning curve formulas, cost estimating relationships (CERs), or other comparative historical data. You may use the max, min, average, etc. of the comparative data base you have to approximate unknown values needed. I'll show some examples of these later on.

Comparisons - Why Use a Spreadsheet or a Program

A balanced usage of these two data processing stages can be obtained using the advantages of each stage. The following is a comparison of advantages/disadvantages of these stages:

- | Stage 1:
Spreadsheet | vs. | Stage 2:
Programs |
|--|-----|---|
| 1) One time use ease. | | 1) Its single purpose use controls results better. |
| 2) Loop calculations are difficult/slow. | | 2) Loop calculations are faster. |
| 3) Template use for data entry, in common and frequent problems. | | 3) Can handle extensive input/output requirements automatically or interactively. |
| 4) User control is very flexible. | | 4) Compiled, debugged program gives known results. |
| 5) Ease of changing. | | 5) Experts needed to change. |
| 6) Relative ease to understand operation. | | 6) Harder to understand operation. |
| 7) One program can do many functions. | | 7) Each program does only one task. |
| 8) Data entry into labeled blank spaces. | | 8) Data entry involved and requires manual to define entries, format, etc. |

- | | |
|--|---|
| 9) Formats can be easily set or reset. | 9) Format changes require reprogramming. |
| 10) Speed of construction. | 10) Programming requires a lot of specialized time. |

Spreadsheet Preparation - General Principles

There are several ways to make spreadsheet usage easier and faster, especially if you are building the spreadsheet from scratch. This list covers some very important principles that will save time throughout the process of using a spreadsheet.

A Dozen Dooezes:

- 1) Put related information in the same row.
- 2) Put constants at top for ease of reference and input.
- 3) Block related groups of information.
- 4) Don't put in blank rows or columns during construction, put them in, if needed, only after your spreadsheet is complete. Generally, blank data rows are input errors to programs.
- 5) Adjust column width for optimum value and computational speed.
- 6) Use 20 or less character width in column A to label rows.
- 7) Generally use 2 or 3 rows for column titles.
- 8) Plan for macros.
- 9) Generally a column is a time frame, use several rows to divide that time frame, eg, a year with 12 months.
- 10) Put constants for rows at far left and right.
- 11) Learn how to sort rows and use absolute references to speed worksheet construction.
- 12) Don't use over one column for text.

Documentation - Organizing All That Information

Using spreadsheets to create inputs has several self documentation advantages. The formulas can be printed or viewed within the spreadsheet. With range naming, the formulas can look exactly like the logical words used to create them. The inputs can be described or listed by their source or in their original form. Good structuring of the spreadsheet will show what is included in results as general variables, eg, GandA, profit, risk, etc. Seeing a label at the top and/or side of the data is invaluable in preventing or reducing transposed or missing entries.

Data Types, Potential Errors

Three basic data types exist for input: alpha-numeric (A/N), real numbers, and integers. These words and numbers may look very similar, but they are not the same to your program. Inputting A/Ns usually involves enclosing the 'text' in quote marks, eg,

'123'. The number 123 is not text to your computer. Nor is 123. the same as 123 . The presence of the decimal makes the integer a real number. The problem during data entry is forgetting the decimal or a quote mark. Using the formatting capability of a spreadsheet within each column solves this data error entry problem.

Data Entry, What Does It Look Like

Here is a visual example of a spreadsheet and a typical word processing entry of the same data:

1) Word Processing entries;

19,9,20,2.369,0.0,9,99,1,5,3,20

2) Spreadsheet Entries (with labels);

N	NSYS	LY	PSC	PSO	NERRC	RFILL	OOPFAC	NECO	ISYR	NDRA
19	9	20	2.369	0.000	9	99	1	5	3	20

I believe the ease of reading and understanding is readily apparent.

How To Compute A Weighted Average.

Let's further look at one of these entries - PSC (Packing and Shipping cost Conus). The following is data from AFLCP 173-10, 1983:

	GBL	LOGAIR
Pk Fac .	1.941 Air \$s.	\$0.532 \$0.391
Pack Lb.	2.369 Surface.	\$0.098

Now let's include entries for weighted ratios of the locations which can then be used to compute PSC.

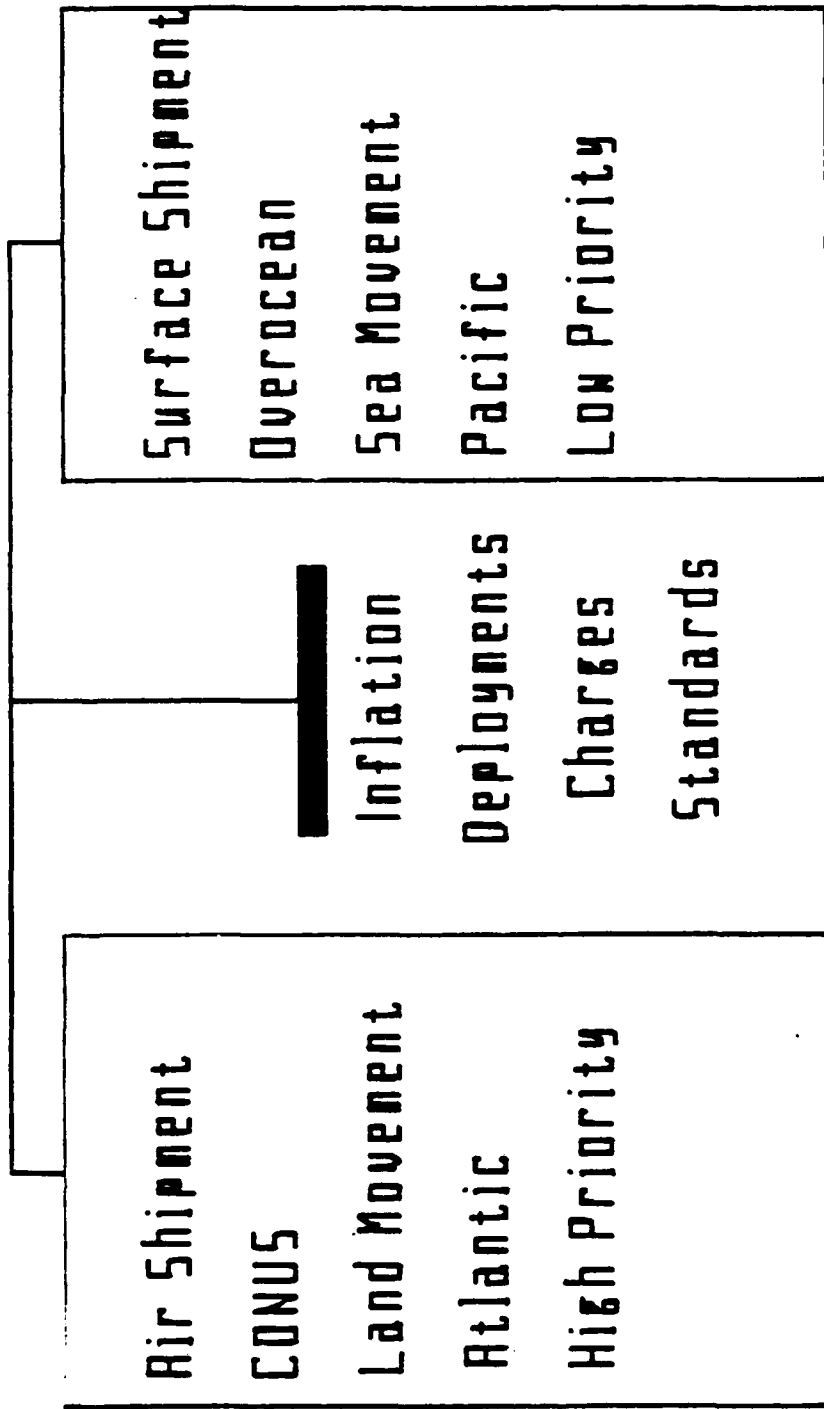
PSC	PkInfB82	1.2337	%Air=\$s.	GBL	LOGAIR
4.870	Pack Lb.	2.369	Surface.	\$0.532	\$0.391
	Pk Fac .	1.941	RatAIR .	100%	0%
			Rat/Loc.	90%	10%

Below is the formula to compute the value of PSC shown.

$$(((0.9*0.532+0.1*0.391)*(0.1))+(0.098*0.9))+2.369*1.941=4.86994$$

Now a more complicated example of the same type problem. This was prepared using LOTUS and is a working example prepared on disk. This is a complete example in compact form to get a weighted value for both PSC and PSO. It is weighting PCTOS (PerCent OverSeas) as a harmonic mean over 20 years to get an accurate split for CONUS and OVEROCEAN. Rates are properly inflated to 84\$'s from the given numbers in AFLCP 173-10. Air shipments are ratioed from the RFILL (Required FILL rate) plus three percent. Finally the weights for GBL thru AAC are

Weighted Average Perspective



How much of each goes where ?

EERC	Days/Mo				30.4166			
	DRCT	BRCT	OSTCON	OSTOS	DRCT	BRCT	OSTCON	OSTOS
1,XD2-SRU	50	7	29.11	77.97	1.64	0.23	0.96	2.56
2,XD3-LRU	42	5	29.11	77.97	1.38	0.16	0.96	2.56
3,CONTRCT	63	12	29.11	77.97	2.07	0.39	0.96	2.56
4,AGMC	50	18	29.11	77.97	1.64	0.59	0.96	2.56
5,00/ALC	60	18	29.11	77.97	1.97	0.59	0.96	2.56
6,WR/ALC	60	18	29.11	77.97	1.97	0.59	0.96	2.56
7,SA/ALC	120	18	29.11	77.97	3.95	0.59	0.96	2.56
8,SM/ALC	120	18	29.11	77.97	3.95	0.59	0.96	2.56
9,OC/ALC	63	15	29.11	77.97	2.07	0.49	0.96	2.56
10,GEN	50	12	31	84	1.64	0.39	1.02	2.76

The codes for this data (1-10) are useful for other uses. They can refer to other tables like this one with different purposes, eg, labor rates, workloads, manning levels, etc.

Using Learning Curves

Learning curves have many applications, so their use within a spreadsheet or a program is a common occurrence. The formula and data complete the requirements for a well recognized cost spreading technique. Consider the example below, the #A/C show the deliveries each year. The row '#Items' is the cumulative quantities delivered. The rows 'Guide' and 'Control' contain the computed dollar spread using LC techniques. On these same rows you can note 'T1 Rst' and 'OOPF' as data. T1 Rst is the restart year used after the item is Out of Production and OOPF is the Out of Production Factor.

#A/C	100	400	500	600	200	0						
XFLU	1	2	3	4	5	6	7	T1	Spread			
#Items	100	500	1000	1600	1800	1800		Rst	Slope	b	T1	OOPF
Guide	100	68	65	63	62	62	200	1	95%	-0.074	100	2
Control	200	136	129	125	124	124	223	5	95%	-0.074	200	1.8

Below is the LC formula, in words, used (cumulative average) for years one thru five. Year seven is the OOP year.

$$T1 * ((((.5) \cdot (1+b) - (\#Items * QPA + .5) \cdot (1+b)) / ((1+b) * (1 - \#Items * QPA - .99999))) \cdot (1/b) \cdot b$$

In this example, year seven used the six prior years computed as a table to select a restart T1, then multiplies that value by the OOPF. The 123 formula, in words, is shown below.

$$@HLOOKUP(T1 Reset, Computed Values LC for 5 yrs, offset) * OOPF$$

Using Comparison Data

Within a spreadsheet it is often useful to be able to

compare source data from several places. The statistical relationships can be computed for the numbers or just the visual documentation of the source of the data is very useful to the analysis. Here is an example:

Source Select	ICA Team	Long Lead	Reverse \$ PRICE	ORLA	lot Aver	\$/ Lb	Aver	Std Dev	Max Value
100	105	135	115	118	125	130	118.2	11.85	135
222	220	240	231.3	225	238	218	227.6	8.048	240

Using CERs (Cost Estimating Relationships)

CERs are parametric formulas based on statistically significant relationships between correlated data. The calculation of the factors is a field in itself, but the use of them is useful in spreadsheets. This example is for a quantity of 10,000 using both CER and LC relationships. The formulas used are shown under the spread.

Number items = 10000	Cum Aver	Parameters----	Inflat Yr	CER or Anal/Eng\$
Gu and Cntrl Cost	11545	T1 Slope	ApDia Res/mil Adjust Qty	
Guidance	4430	\$38,391	85.00%	1 2.4 87.00% 1 \$33,400
Cntrl/Atplt	7115	\$28,853	90.00%	30 12 68.30% 1000 \$6,896

CER Guide = 13600 + 7800 * Weight² + 5000 * Force
 CER Cntrl/Atplt = 48 * ApDia + 881 * Res/mil - 5116
 T1 = CER value / Adjust
 T1 = CER value / (Qty^{1/2} * (@LN(Slope) / @LN(2)) * Adjust)
 Cum Aver = T1 * 10000^{1/2} * (@LN(Slope) / @LN(2))

What Is A Macro

A macro has one nice quality for learning to write it - it is the same instructions used when you normally use the spreadsheet commands. Just add a name to the written instructions you did. The macro then can be executed by calling its name when you want to use it. A special tip for writing macros is to capitalize the range names so they stand out and leave the letter commands in small letters. The program doesn't care either way, so do as you choose.

Macro Examples and Their Use In Menus

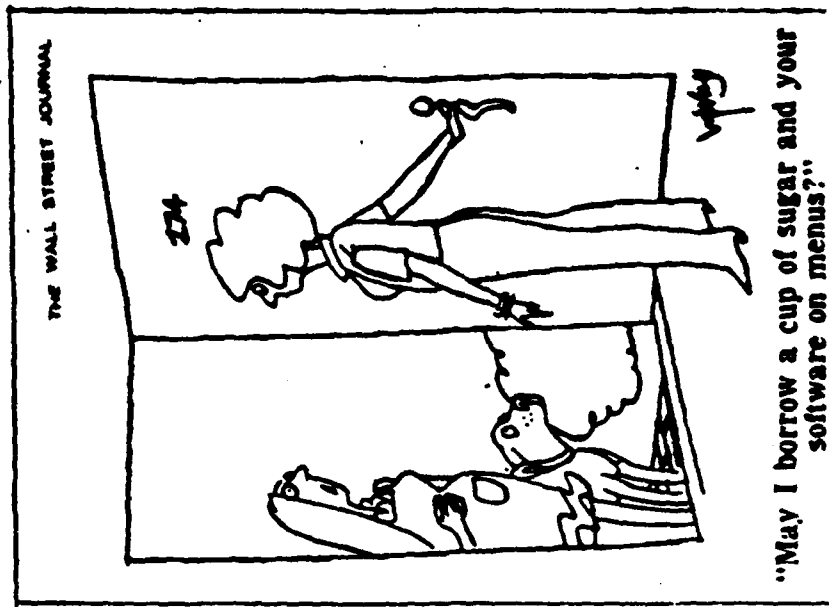
The macro is the starting place for user directed automatic actions in a spreadsheet. I'm highlighting LOTUS 123 because it has sufficient power to function well in a cost estimating environment and I know it well. Other integrated spreadsheets and Supercalc3, version 2 can perform these tasks with some modification.

Below are three macros. Note the letters I capitalize are the letter instructions in the macros as I word them logically. The '/' is the command character. The '½' is the macro identification character. A macro name is the single letter A thru Z. The first, ½I, is to Print on a Printer the Range named INPUTS Cancelling Borders and using Other Other option Unformatted then Quitting other and Going to print before final Quit. The second, ½M, is a macro to go (x is special macro only command) to a Menu named MENU. The third, ½O, is the autoexec macro.

```
½I      /pprINPUTS#cboouqgq
½M      /xmMENU#
½O      /xmMENU#
```

If macros are used in a menu system like LOTUS, other neat things can be done. Putting macros into a menu allows fast use of their power. I'll illustrate. During use of a spreadsheet, a typical need is to change some numbers, get the new calculation, print the new results, and save an ASCII file, the data, for input into another program. Later the summary documentation of inputs, methodology, assumptions, standard factors used, and the like will also be updated and printed. Each of these separate actions can be a selection in a user generated menu. Choosing each of them in turn from the menu will perform the required actions with no further action required, except loading a disk or readying a printer.

Before continuing, I will briefly give the rules to construct LOTUS menus. Refer to the first menu example below. The upper left corner cell of the menu must be named (like this, /rncNAME#). Line one contains the user defined names of what is to be done. These should each begin with a different letter in each menu and ideally be a single word. The maximum is eight, the last should be a RETURN or QUIT, a blank block must be in the right most cell of line one. The second line is further instructions for the single key word in line one. These are placed in the cell directly below the respective key word. The third line, again directly below the respective key words, contains the macros. There are some special instructions that can only be used within macros (the x commands). Since they are useful, I suggest studying them. My examples use them without special comment.



"May I borrow a cup of sugar and your software on menus?"

SIMPLE MENU CONSTRUCTION RULES

NAME the upper left corner.

Line1	One word instruction keys having different first letters. BLANK
Line2	Further instructions for each key word.
Line3	The macros used to do your bidding.

1 2 3 4 5 6 7 8 Max Width

BLANK LINE UNDER EACH OF THE MACROS.

Sample Keys

INPUT OUTPUT PRINT START ENTER HELP MENUS OUT

Menus can interlock.

Using range names consistently allows common usage of menus.

A program of great value is easily possible.

Sample Ranges

ALL	DELIVERIES	SUMMARY	MENU	LIST
SET	DAGES	LABOR	PRINT	COSTS
STOP	TABLE	HELP	SCHEDULE	FIRST

HINT: Always capitalize RANGE NAMES, easier to see in macro.

A special note is this paper was written using LOTUS 123, all examples are thus real working ones. A demonstration of an interlocking set of menus look like those following the master menu. The first menu is the master that calls all the rest in this example. You can't read all the instructions that are there. Therefore, I'm illustrating MENU twice. Both methods work, but the second shows instructions for reading the best. The second line wording I will list, but it is a non-functional listing.

Method One:

```
MENU INPUT PRINT WORK RANGE SAVE CHOOSE HELP QUIT
Input mPrint mWork MeRange MSave MeChoose Help MeLeave
/xmINPU/xmPRIN/xmWORK/xmRANG/xmSAVE/xmCHOO/xmHELP/xq
```

A blank line should be left under each menu. These other menus are sets of instructions to perform the specific changes that may be located anywhere in the spreadsheet. This is how it looks on the screen with a column width of seven. Much of each of the instructions are hidden from view unless the cursor is on the cell in question.

Method Two:

```
MENU INPUT PRINT WORK RANGE SAVE CHOOSE HELP QUIT
Input mPrint mWork MeRange MSave MeChoose Help MeLeave
/xgI /xP /xW /xR /xS /xC /xH /xq
```

```
I /xmINPUT Input menu.
P /xmPRINT Print menu.
W /xmWORK Work Menu.
R /xmRANGE Range Menu.
S /xmSAVE Save Menu.
C /xmCHOOSE Choose Menu.
H /xmHELP Help Menu.
```

I am including representative menus for all mentioned above in this interlocking set. They would be customized for specify needs.

```
INPUT PROGRAMLABOR SUPPORTCOMMON MAINTENDEPOT OTHER RETURN
Input pInput mInput sInput cInput MInput DInput mReturn to
goto P goto M goto S goto C goto M goto D goto M/xmMENU h
```

```

PRINT  INPUTS SUMMARYLONG  MAJOR  APPROPRIATE PRINT-CRETURN
Print iPrint tPrint tPrint tPrint tPrint CReturn to main m
/pprINP#calc#/#calc#/#calc#/#calc#/#pprCOS/xmMENU#LUMNS#r

WORK   TEXT    COMBINEEND  START  RETURN
Line 2 instructions.      Return to main menu.
Line 3 macros.            /xmMENU#

RANGE  INPUTS SUMMARYLONG  MAJOR  RETURN
Line 2 instructions.      Return to main menu.
Line 3 macros.            /xmMENU#

SAVE   PRN-FILFIRST  SPREADSSUMMARYRETURN
Save upSave INITIAL file as Return to main menu.USE ONLY 0
/xiSET=/xiSET=1#/dsdSORTCOST/xmMENU#g/pfB:COST#rCOST#cboou
/xiSET,.1#/dsdSORTCOST#pWUC#a#g/pfB:COST#rCOST#cboo

CHOOSE SCHEDULETABLE  FIRST-CBLANK  RETURN
Set value to select scheduleReturn to main menu.ERNATE.
/xnEnter 1=NORMAL SCHEDULE, /xmMENU#SET#
/xiSET=1#goto#SETN#NORM Sd#calc#/xmWORK#
/xiSET,.1#goto#SETN#ALTR Sd#calc#/xmWORK#

HELP   LOTUS  MENUS  FUNC-KESAVING RETURN
Line 2 instructions.      Return to main menu.
Line 3 macros.            /xmMENU#

```

I believe this material is enough to show the setup and potential for using menus and macros. Please refer to source material for details not mentioned here.

The Future of This Concept

A short ditty, versification, synopsis, or summary:
 Some programs will never be needed because of a spreadsheet.
 Some programs will be entirely replaced by a spreadsheet.
 Some programs will be enhanced by a spreadsheet.
 Some programs will be improved/changed because of a spreadsheet.
 Some programs will never use a spreadsheet.
 Some programs aren't used anyway, even with a spreadsheet.

Conclusion

It is a rare office that can't use a spreadsheet effectively. Using it for program inputs is one of many potential improvements.

END

FILMED

12-85

DTIC