ALGOL-M
CP/M VERSION 1.1
AUGUST 1978

WRITTEN BY
LT MARK MORANVILLE(CODE 52MI)
NAVAL POSTGRADUATE SCHOOL
MONTEREY,CALIF 93940

ON THIS DISK YOU SHOULD FIND VARIOUS TXT FILES, THESE FILES CONTAIN DOCUMENTATION CONCERNING THE ALGOL-M PROGRAMMING LANGUAGE. THEY INCLUDE THE FOLLOWING:-

ALGINTRO.TXT: AN OVERVIEW OF THE ALGOL-M LANGUAGE .

USRMAN.TXT: AN ALGOL-M USER'S MANUAL WITH A COMPLETE DESCRIPTION OF THE VARIOUS ALGOL-M LANGUAGE STRUCTURES.

COMERR.TXT: A DESCRIPTION OF THE ALGOL-M COMPILER ERROR MESSAGES.

RUNERR.TXT: A DESCRIPTION OF THE ALGOL-M INTERPRETER ERROR AND WARNING MESSAGES.

IN ADDITION TO THESE FILES YOU SHOULD FIND VARIOUS SAMPLE PROGRAMS WHICH DEMONSTRATE MOST OF THE FEATURES OF THE LANGUAGE.

INCLUDED IN THIS SECTION IS A DESCRIPTION OF HOW TO CREATE,COMPILE, AND EXECUTE AN ALGOL-M PROGRAM.

ALGOL-M REQUIRES A CP/M SYSTEM WITH AT LEAST 24K OF MEMORY. TO CREATE AND RUN A PROGRAM YOU DO THE FOLLOWING:-

1. CREATE AN "ALG" FILE USING THE CP/M EDITOR. THIS FILE MUST BEGIN WITH A "BEGIN" OR A COMMENT(WHICH CONSISTS OF EITHER A. A STRING ENCLOSED WITH "%" OR B. A STRING BEGINNING WITH THE WORD "COMMENT" AND ENDING WITH A ";"). THIS FILE MUST END WITH EITHER A. THE WORD "END" WITH NO ; OR B. AN "END" FOLLOWED BY AN "EOF". REVIEW THE VARIOUS SAMPLE PROGRAMS FOR THE EXACT FORMAT.

2. COMPILE THE "ALG" PROGRAM BY TYPING "ALGOLM FILENAME" FOLLOWED BY A CARRIAGE RETURN.
"FILENAME" STANDS FOR THE NAME OF YOUR FILE. YOU MAY OBTAIN AN OPTIONAL LISTING OF THE FILE AT THE TERMINAL BY ADDING A "$A" AFTER THE FILE NAME. THE FORMAT FOR THE LISTING IS AS FOLLOWS:

```
1 1 BEGIN
```

THE NUMBER 1 ON THE LEFT REPRESENTS THE LINE NUMBER. THE NUMBER 1 IN THE MIDDLE REPRESENTS THE BLOCK LEVEL. AT THE END OF THE PROGRAM THE LAST LINE MUST CONTAIN A BLOCK LEVEL OF 0 OR YOU HAVE LEFT OFF AN END IN YOUR PROGRAM. THE WORD BEGIN REPRESENTS THE INFORMATION ON LINE 1. THE FORMATS FOR COMPILING ARE AS FOLLOWS:-
(THE INFORMATION CONTAINED IN THE PARENTHESES IS NOT TYPED)

```
ALGOLM LUNAR (NO LISTING,NO TRACE)

ALGOLM LUNAR $A (LISTING,NO TRACE)
```

```
ALGOLM LUNAR $E (NO LISTING,TRACE)

ALGOLM LUNAR $AE (LISTING,TRACE)
```

3. TO ACQUIRE A TRACE OF PROGRAM EXECUTION USE THE $E OPTION. WHEN THE PROGRAM IS EXECUTED THE SOURCE CODE LINE NUMBER WILL BE TYPED AT THE CONSOLE AFTER EXECUTION OF THAT LINE HAS COMPLETED.

4. AFTER A SUCCESSFUL COMPILATION THERE WILL BE AN "AIN" FILE CREATED. TO EXECUTE THE "AIN" FILE TYPE "RUNALG FILENAME" FOLLOWED BY A CARRIAGE RETURN.

APPENDIX C - INTERPRETER ERROR/WARNING MESSAGES

ERROR MESSAGES

AB	Array subscript out of bounds.
CE	Disk file close error.
DB	Input field length is larger then the buffer size.
DW	Disk file write error.
ER	Variable block size write error.
IO	Integer overflow(integer value greater than 16383).
IR	Record number incorrect or random file is not initialized.
ME	Disk file creation error.
NA	No AIN file found on directory.
OV	Decimal register overflow during arithmetic operation/ load.
RE	Attempt to read past end of record on blocked file.
RU	Attempt to random access a non-blocked file.
SK	Stack overflow(no more memory available).

WARNING MESSAGES

AZ	Attempt to allocate null decimal or string, system defaults to 10 digits/characters.
DO	Decimal overflow during store operation. The value of the variable is set to 1.0 and execution continues. The variable's allocation size should be increased in it's declaration statement.
DI	Disk file variable format error.
DZ	Decimal division by zero, result is set to 1.0.

EF End of file on read.

IA Integer addition/subtraction over/under flow result is set to 1.

II Invalid Console Input. Try input again.

IR Record number incorrect or random file is not initialized.

IZ Integer division by zero. Divisor set to 1 and division is completed.

NX Negative exponential. Exponentiation not done.

SO Characters lost during string store.

APPENDIX B - COMPILER ERROR MESSAGES

AS Function/Procedure on left hand side of assignment statement.

BP Incorrect bound pair subtype (must be integer).

DE Disk error; no corrective action can be taken in the program.

DD Doubly declared identifier, label, variable etc.

FP Incorrect file open statement.

IC Invalid special character.

ID Subtypes incompatible (decimal values can not be assigned to integer variables).

IO Integer overflow.

IT Identifier is not declared as a simple variable or function.

NG No ALG file found.

NI Subtype is not integer.

NP No applicable production exists.

NS Subtype is not string.

NT For clause,Step expression,Until clause expressions are not of the same subtype.(must all be integer or decimal).

PC Number of parameters in procedure call does not match the number in the procedure declaration.

PD Undeclared parameter.

PM Parameter type does not match the declared type.

SO Stack overflow.

SI Array subscript is not of subtype integer.

TD Subtype has to be integer or decimal.

TM Subtypes do not match or are incompatible.

TO Symbol table overflow.

TS Undeclared subscripted variable.

UD Undeclared identifier.

UF Undeclared file/function.

UL Undeclared label.

UP Undeclared procedure.

US Undeclared simple variable.

VO Varc table overflow. Possibly caused by too many long identifiers.