APPLE® COMMANDS

APPLE II, II+ and IIe

D: DOS f: File Name m,n,l,j: Integers
A: Applesoft A$: String x,y,z: Real nos.
I: Integer Basic X: Variable

Consult your Apple manuals for detailed descriptions and instructions.

- A I **ABS(x)** *Absolute (positive) value of x*
- A I **AND** *Logical "and" in an IF statement*
D - - **APPEND f** *Add data to sequential text file*
- A I **ASC("A")** *ASCII value of character*
- A I **ASC(A$)** *ASCII of string's first character*
- A I **AT** *See DRAW, XDRAW, HLIN and VLIN*
- A - **ATN(x)** *Arctangent of x in radians*
- - I **AUTO n, m** *Start auto-line numbering*

D - - **BLOAD f** *Load binary file f*
D - - **BRUN f** *Load & run certain binary files*
D - - **BSAVE f, An, Lm** *Save data at n, length m*

- A I **CALL n** *Branch to machine subroutine at n*
D - - **CATALOG** *Display disk contents*
D - - **CHAIN f** *Run file f; don't clear variables*
- A - **CHR$(n)** *Character whose ASCII value is n*
- A - **CLEAR** *Reset all variables to zero*
D - - **CLOSE** *Stop reading or writing a text file*
- - I **CLR** *Reset all variables to zero*
- A I **COLOR=n** *Set lo-res color to n (0-15)*
- - I **CON** *Continue an Integer program*
- A - **CONT** *Continue an Applesoft program*
- A I **control-C** *Stop a Basic program*
D - - **control-D** *DOS command character*
- A I **control-G** *Beep the speaker*
- A I **control-H** *Backspace (left-arrow)*
- A I **control-I** *Tab (Apple IIe only)*
- A I **control-J** *Line feed (down-arrow on IIe)*
- A I **control-K** *Up-arrow (Apple IIe only)*
- A I **control-M** *Carriage-return*
- A I **control-U** *Right-arrow*
- A I **control-X** *Cancel line being typed*
- A I **control-[** *Escape*
- A - **COS(x)** *Cosine of x in radians*

- A - **DATA A$, x, y, z** *Data to be READ*
- A - **DEF FN A(X)=f(x)** *Define function*
- A I **DEL n, m** *Delete program lines n to m*
D - - **DELETE f** *Delete file f from disk*
- A I **DIM X(n)** *Dimension an array*
- A I **DIM A$(n)** *Dimension a string*
- A - **DRAW n AT I, j** *Draw hi-res shape n*
- - I **DSP X** *Display X's values & line numbers*

- A I **END** *Stop program without message*
- A I **esc-@** *Clear the text screen; no prompt*
- A I **esc-A** *Move cursor one space right*
- A I **esc-B** *Move cursor one space left*
- A I **esc-C** *Move cursor one space down*
- A I **esc-D** *Move cursor one space up*
- A I **esc-I** *Cursor up; recursive*
- A I **esc-J** *Cursor left; recursive*
- A I **esc-K** *Cursor right; recursive*
- A I **esc-M** *Cursor down; recursive*
D - - **EXEC f** *Execute text file f*
- A - **EXP(x)** *e (2.718289) to the xth power*

- A - **FLASH** *Set flashing screen output*
- A - **FN** *See DEF FN*
- A I **FOR X=n TO m** *Let X=n, X=n+1... until X=m*
D - - **FP** *Switch to Applesoft Basic*
- A - **FRE(0)** *Amount of free memory available*

- A - **GET A$** *Wait for one-character input*
- A - **GET X** *Wait for one-number input*

- A I **GOSUB n** *Branch to subroutine at line n*
- A I **GOTO n** *Branch to line n*
- - I **GOTO X or GOSUB X** *Branch to line X*
- A I **GR** *View and clear lo-res page 1*

- A - **HCOLOR=n** *Set hi-res color to n (0-7)*
- A - **HGR** *View and clear upper hi-res page 1*
- A - **HGR2** *View and clear full hi-res page 2*
- A I **HIMEM:n** *Set highest address available*
- A I **HLIN n, m AT j** *Draw horizontal lo-res line*
- A - **HOME** *Clear text screen to black*
- A - **HPLOT I, j** *Plot hi-res point*
- A - **HPLOT I, j TO n, m** *Draw hi-res line*
- A - **HTAB n** *Cursor to horizontal tab n (1-40)*

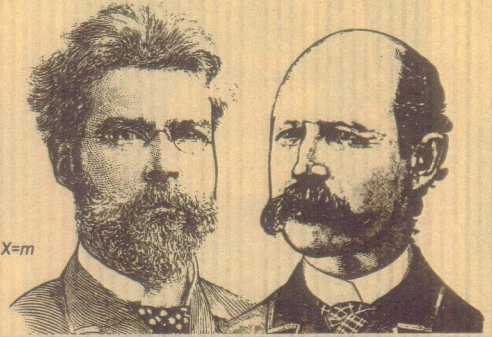
- A I **IF...THEN...** *Logical "if" true, "then" execute*
D A I **IN#n** *Take input from slot n*
D - - **INIT f** *Erase and format a disk*
- A I **INPUT X (or A$)** *Wait for input and return*
- - I **INPUT "ABC", X (or A$)** *Print & get input*
- A - **INPUT "ABC"; X (or A$)** *Print & get input*
D - - **INT** *Switch to Integer Basic*
- A - **INT(RND(1)*n)** *Random integer 0 to n-1*
- A - **INT(x)** *Integer value of x*
- A - **INVERSE** *Set black-on-white text output*

- A - **LEFT$(A$, n)** *Left n characters of A$*
- A I **LEN(A$)** *Number of characters in A$*
- A I **LET X=Y** *X=Y ("LET" is optional)*
- A I **LIST** *List program from beginning*
- A - **LIST-n** *List to line n*
- A - **LIST n-** *List from line n*
- A - **LIST n-m** *List lines n through m*
- A I **LIST n, m** *List lines n through m*
- A I **LOAD** *Load program from tape*
D - - **LOAD f** *Load file from disk*
D - - **LOCK f** *Protect file from alteration*
- A - **LOG(x)** *Natural logarithm of x*
- A I **LOMEM:n** *Set lowest memory available*

- - I **MAN** *Cancel AUTO*
D - - **MAXFILES n** *Reserve file buffers (1-16)*
- A - **MID$(A$, n, m)** *m characters, starting at #n*
- - I **A$(n, m)** *Characters n through m of A$*
- - I **m MOD n** *Remainder of m divided by n*
D - - **MON C, I, O** *Display disk functions*

- A I **NEW** *Delete current program*
- A - **NEXT** *Define bottom of FOR-NEXT loop*
- A I **NEXT X** *Define bottom of FOR-NEXT loop*
D - - **NOMON C, I, O** *Cancel MON*
- A - **NORMAL** *Set white-on-black text output*
- A I **NOT** *Logical "not" in an IF statement*
- A I **NOTRACE** *Cancel TRACE*

- A - **ON X GOSUB n, m...** *GOSUB Xth number*
- A - **ON X GOTO n, m** *Branch to Xth number*



- A - **ONERR GOTO n** *Branch to n if error*
D - - **OPEN f** *Begin READ or WRITE of text file*
- A I **OR** *Logical "or" in an IF statement*

- A I **PDL(n)** *Value (0-255) of paddle n (0-3)*
- A I **PEEK(n)** *Memory value at location n*
- A I **PLOT I, j** *Plot lo-res dot*
- A I **POKE n, m** *Set memory at n to value m*
- A I **POP** *Cancel GOSUB*
- A - **POS(0)** *Horizontal cursor position (0-39)*
D - - **POSITION f** *Locate READ or WRITE in file*
D A I **PR#n** *Send output to slot #n*
- A I **PRINT** *Skip a text line*
- A I **PRINT "ABC"** *Print characters in quotes*
- A I **PRINT X** *Print value of variable X*

- A - **READ A$** *Get DATA string*
- A - **READ X** *Get DATA value*
D - - **READ f** *Get input from text file*
- A - **RECALL X** *Retrieve array from tape*
- A I **REM** *Programmer's remark follows*
D - - **RENAME f1, f2** *Rename file on disk*
- A - **RESTORE** *Set pointer to 1st DATA element*
- A - **RESUME** *Continue where error occurred*
- A I **RETURN** *Branch to statement after GOSUB*
- A - **RIGHT$(A$, n)** *Last n characters of A$*
- A - **RND(0)** *Repeat last random number*
- A - **RND(1)** *Random number (0 to 0.999999999)*
- - I **RND(n)+1** *Random integer between 1 & n*
- A - **ROT=n** *Set rotation of shape to n (0-64)*
- A I **RUN** *Execute program from 1st line number*
- A I **RUN n** *Execute program from line n*
D - - **RUN f** *Load and execute program from disk*

- A I **SAVE** *Save program to tape*
D - - **SAVE f** *Save program to disk*
- A - **SCALE=n** *Set size of DRAW or XDRAW*
- A I **SCRN(I, j)** *Lo-res screen color at point*
- A I **SGN(X)** *Sign (+1, -1 or 0) of X*
- A - **SHLOAD** *Load shape table from tape*
- A - **SIN(x)** *Sine of x in radians*
- A - **SPC(n)** *n spaces in print-statement*
- A - **SPEED=n** *Character output rate (0-255)*
- A - **SQR(x)** *Square root of x*
- A I **STEP n** *Increment size in FOR-NEXT loop*
- A - **STOP** *Halt program and print line number*
- A - **STORE X** *Store array on tape*
- A - **STR$(x)** *String equivalent of value x*

- A - **TAB(n)** *Cursor position in print statement*
- - I **TAB n** *Cursor to horizontal position n*
- A - **TAN(x)** *Tangent of x in radians*
- A I **TEXT** *Switch to text mode*
- A I **THEN** *Logical "then" in an IF statement*
- A I **TO** *See FOR and HPLOT*
- A I **TRACE** *Print line numbers being executed*

D - - **UNLOCK f** *Cancel LOCK*
D - - **USR(x)** *Pass x to machine subroutine*

- A - **VAL(A$)** *Numeric value of string A$*
D - - **VERIFY f** *Verify file on disk*
- A I **VLIN n, m AT I** *Vertical lo-res line*
- A I **VTAB n** *Move cursor to vertical position n*

- A - **WAIT I, j, k** *Insert conditional pause*
D - - **WRITE f** *Write to a text file*

- A - **XDRAW n AT I, j** *DRAW in opposite color*
- A - **XPLOT** *Unused reserved word*

- - I **#** *Not equal to*
- A - **?** *PRINT*

Beagle Bros™
Micro Software Inc.

Beagle Bros sells useful and entertaining Utilities, Games and Publications for Apple II, II+ and IIe Computers.
To get on a really good mailing list, write or call:
BEAGLE BROS, 4315 Sierra Vista, San Diego, California 92103 / Phone 619-296-6400

4/83