

BASIC

BASIC (an acronym for Beginner's All-purpose Symbolic Instruction Code) is a family of general-purpose, high-level programming languages whose design philosophy emphasizes ease of use. In 1964, John G. Kemeny and Thomas E. Kurtz designed the original BASIC language at Dartmouth College in New Hampshire, United States. They wanted to enable students in fields other than science and mathematics to use computers. At the time, nearly all use of computers required writing custom software, which was something only scientists and mathematicians tended to learn.

Versions of BASIC became widespread on microcomputers in the mid-1970s and 1980s. Microcomputers usually shipped with BASIC, often in the machine's firmware. Having an easy-to-learn language on these early personal computers allowed small business owners, professionals, hobbyists, and consultants to develop custom software on computers they could afford. In the 2010s, BASIC was popular in many computing dialects and in new languages influenced by BASIC, such as Microsoft's Visual Basic.

History

Before the mid-1960s, the only computers were huge mainframe computers. Users submitted jobs (calculations or other requests) on punched cards or similar media to specialist computer operators. The computer stored these, then used a batch processing system to run this queue of jobs one after another, allowing very high levels of utilization of these expensive machines. As the performance of computing hardware rose through the 1960s, multi-processing was developed. This allowed a mix of batch jobs to be run together, but the real revolution was the development of time-sharing. Time-sharing allowed multiple remote interactive users to share use of the computer, interacting with the computer from computer terminals with keyboards and teletype printers, and later display screens, in much the same way as desktop computers or personal computers would be used later.

Origin

The original BASIC language was released on May 1, 1964 by John G. Kemeny and Thomas E. Kurtz and implemented under their direction by a team of Dartmouth College students. The acronym BASIC comes from the name of an unpublished paper by Thomas Kurtz. BASIC was designed to allow students to write mainframe computer programs for the Dartmouth Time-Sharing System. It was intended specifically for less technical users who did not have or want the mathematical background previously expected. Being able to use a computer to support teaching and research was quite novel at the time.

The language was based on FORTRAN II, with some influences from ALGOL 60 and with additions to make it suitable for timesharing. Initially, BASIC concentrated on supporting straightforward mathematical work, with matrix arithmetic support from its initial implementation as a batch language, and character string functionality being added by 1965. Wanting use of the language to become widespread, its designers made the compiler available free of charge. (In the 1960s, software became a chargeable commodity; until then, it was provided without charge as a service with the very expensive computers, usually available only to lease.) They also made it available to high schools in the Hanover, New Hampshire area and put considerable effort into promoting the language. In the following years, as other dialects of BASIC appeared, Kemeny and Kurtz's original BASIC dialect became known as Dartmouth BASIC

Typical BASIC keywords[edit]

Data manipulation

LET—assigns a value (which may be the result of an expression) to a variable.

DATA—holds a list of values which are assigned sequentially using the READ command.

Program flow control

IF ... THEN ... ELSE—used to perform comparisons or make decisions.

FOR ... TO ... {STEP} ... NEXT—repeat a section of code a given number of times. A variable that acts as a counter is available within the loop.

WHILE ... WEND and REPEAT ... UNTIL—repeat a section of code while the specified condition is true. The condition may be evaluated before each iteration of the loop, or after.

DO ... LOOP {WHILE} or {UNTIL}—repeat a section of code Forever or While/Until the specified condition is true. The condition may be evaluated before each iteration of the loop, or after.

GOTO—jumps to a numbered or labelled line in the program.

GOSUB—jumps to a numbered or labelled line, executes the code it finds there until it reaches a RETURN Command, on which it jumps back to the operator following the GOSUB – either after a colon, or on the next line. This is used to implement subroutines.

ON ... GOTO/GOSUB—chooses where to jump based on the specified conditions. See Switch statement for other forms.

DEF FN—a pair of keywords introduced in the early 1960s to define functions. The original BASIC functions were modeled on FORTRAN single-line functions. BASIC functions were one expression with variable arguments, rather than subroutines, with a syntax on the model of DEF FND(x) = x*x at the beginning of a program. Function names were originally restricted to FN+one letter.

Input and output

LIST—displays all inputted code.

PRINT—displays a message on the screen or other output device.

INPUT—asks the user to enter the value of a variable. The statement may include a prompt message.

TAB or AT: sets the position where the next character will be shown on the screen or printed on paper.

List of functions

ABS—Absolute value

ATN—Arctangent value (result in radians)

COS—Cosine value (argument in radians)

EXP—Exponential value

INT—Integer value

LOG—Natural Logarithmic value

RND—Random value

SIN—Sine value (argument in radians)

SQR—Square root value

TAN—Tangent value (argument in radians)

Miscellaneous

REM—holds a programmer's comment or REMark; often used to give a title to the program and to help identify the purpose of a given section of code.

USR—transfers program control to a machine language subroutine, usually entered as an alphanumeric string or in a list of DATA statements.

TRON—turns on display of each line number as it is run ("TRace ON"). This was useful for debugging or correcting of problems in a program.

TROFF—turns off the display line numbers.

ASM—some compilers such as Freebasic, Purebasic, and Powerbasic, also support inline assembly language, allowing the programmer to intermix high-level and low-level code, typically prefixed with "ASM" or "!" statements.

```
INPUT "What is your name: "; U$
20 PRINT "Hello "; U$
30 INPUT "How many stars do you want: "; N
40 S$ = ""
50 FOR I = 1 TO N
60 S$ = S$ + "*"
70 NEXT I
80 PRINT S$
90 INPUT "Do you want more stars? "; A$
100 IF LEN(A$) = 0 THEN GOTO 90
110 A$ = LEFT$(A$, 1)
```

```
120 IF A$ = "Y" OR A$ = "y" THEN GOTO 30
130 PRINT "Goodbye "; U$
140 END
```