Prize in Physics (with Bardeen and Brattain), 1956. Emeritus professor, 1979–89. Died 12 August 1989 in Palo Alto, California.

SELECTED WRITINGS

Bardeen, John, Walter Brattain, and William Shockley. *Nobel Lectures—Physics, 1942–62.* New York: Elsevier, 1964.

Shockley, William. *Electrons and Holes in Semiconductors, with Applications to Transistor Electronics.* New York: Van Nostrand, 1950.

FURTHER READING

Caddes, Carolyn. *Portraits of Success: Impressions of Silicon Valley Pioneers.* Palo Alto, Calif.: Tioga, 1986.

Riordan, Michael, and Lillian Hoddeson. *Crystal Fire: The Invention of the Transistor and the Birth of the Information Age.* New York: Norton, 1998.

Wagner, Peter. "How Microchips Shook the World." *Byte*, Dec. 1996, pp. 69.

—Chris Woodford

BASIC

BASIC was the first programming language for an entire generation of **microcomputer** programmers: The first product sold by **Microsoft** was an interpreter of BASIC; the legendary **Apple II** computer came with a BASIC interpreter written single-handedly by **Steve Wozniak** (1950–); and the first **IBM** personal computers (PCs) included a BASIC interpreter bundled with the **operating system**. In short, it is difficult to overstate the influence of BASIC on the programming community of the 1970s and 1980s.

BASIC (Beginner's All-purpose Symbolic Instruction Code) was developed by John Kemeny (1926–92) and Thomas Kurtz (1928–) at Dartmouth College in the mid-1960s. It was used as an interpreted language for time-sharing systems: The students could sit at computer terminals, type in several lines of code, and get immediate results. This was very different from the way that computers were programmed at the time. A **Fortran** programmer, for example, had to punch a stack of cards, deliver them to the computing room, and wait for the results. The timing of BASIC's creation was fortuitous: Interactive computing was just appearing, and BASIC arrived on the scene at the same time that the first time-sharing systems were being created.

As the name implies, BASIC is extremely simple. The lines of the program are numbered and it is possible to write them out of order (line 20 before line 10) since the interpreter reads them in the correct order. It is also possible to eliminate a line simply by writing its number and a return. The interpreter of the language thus provides a very simple editor to extend or modify a program.

The example below shows a BASIC program in which the user is prompted to type in a number of years, and the computer prints the equivalent number of months, by multiplying the number entered by 12.

```
10 PRINT 'Enter number of years'
20 INPUT YEARS
30 PRINT 'The number of months is ', YEARS*12
```

Notice that variables do not have to be declared before using them, as in other programming languages. Typically, it is possible to write nontrivial BASIC programs after only a few hours of practice.

There is an **American National Standards Institute** standard for minimal BASIC (X3.60-1978) and another for full BASIC. The language has continued to evolve over the years. Visual BASIC from Microsoft looks more like **structured programming** languages of the **Pascal** type. It is also used as a macrolanguage for applications such as spreadsheets or word processing programs.

FURTHER READING

Ford, Neville J. *Computer Programming Languages Made Simple.* Upper Saddle River, N.J.: Prentice Hall, 1990.

Kemeny, John G., and Thomas E. Kurtz. *BASIC Programming.* New York: Wiley, 1980.

———. *Back to BASIC: The History, Corruption and Future of the Language.* Reading, Mass.: Addison-Wesley, 1985.

—Raúl Rojas

Batch Processing

Before the advent of interactive computing, programmers, who were the only direct users of computers, had to prepare a set of punched cards containing their programs. The cards were given to the computer operator, who would put them in a card reader. The