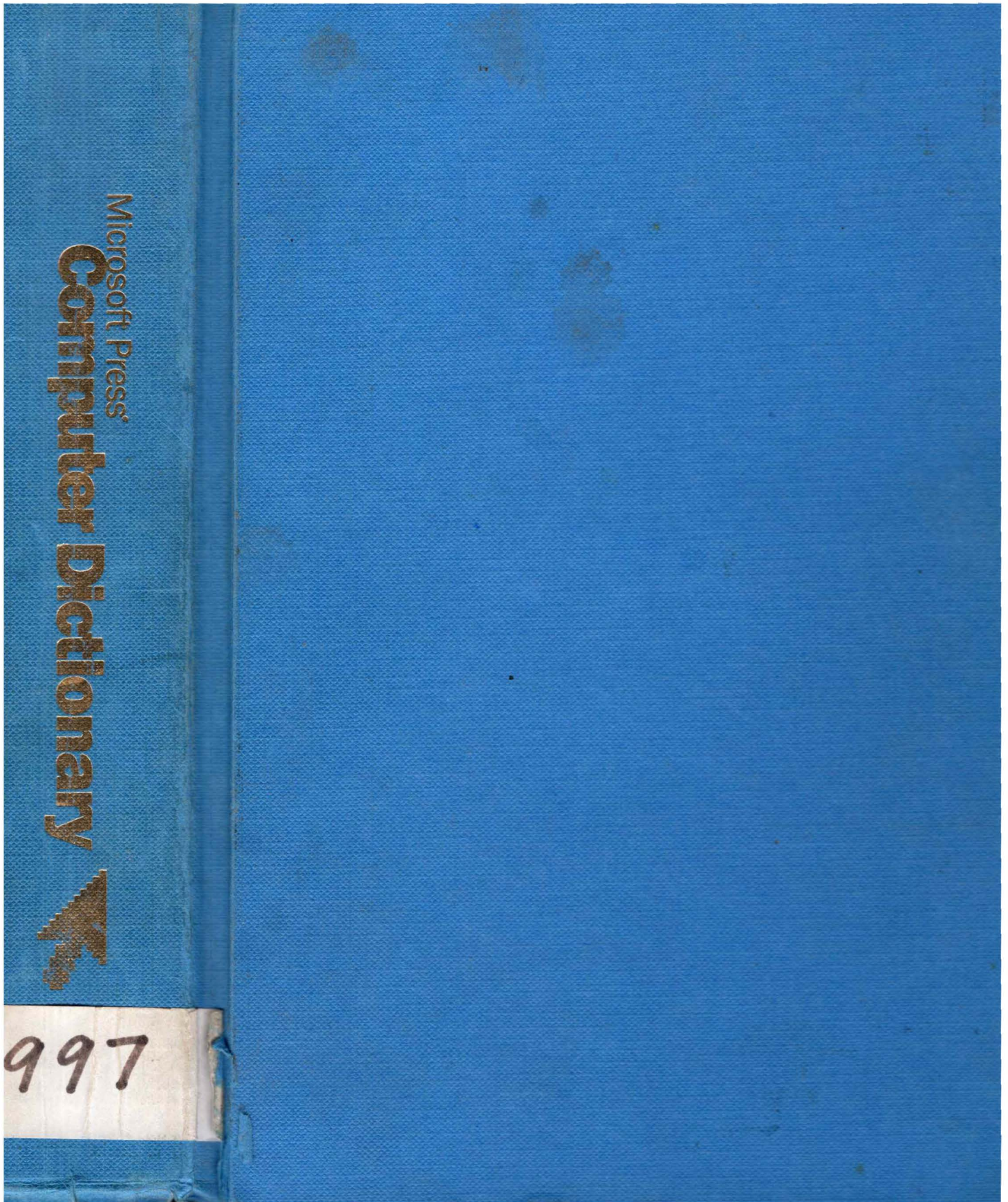




Microsoft Press

# Computer Dictionary

Third Edition

**Microsoft** Press

Petitioner's Exhibit 1009  
Page 0002

**PUBLISHED BY**

Microsoft Press  
A Division of Microsoft Corporation  
One Microsoft Way  
Redmond, Washington 98052-6399

Copyright © 1997 by Microsoft Corporation

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Library of Congress Cataloging-in-Publication Data pending.

ISBN 1-57231-743-4

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 QMQM 2 1 0 9 8 7

Distributed to the book trade in Canada by Macmillan of Canada, a division of Canada Publishing Corporation.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office. Or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at [mspress.microsoft.com](http://mspress.microsoft.com).

Macintosh, Power Macintosh, QuickTime, and TrueType fonts are registered trademarks of Apple Computer, Inc. Intel is a registered trademark of Intel Corporation. DirectInput, DirectX, Microsoft, Microsoft Press, MS-DOS, Visual Basic, Visual C++, Win32, Win32s, Windows, Windows NT, and XENIX are registered trademarks and ActiveMovie, ActiveX, and Visual J++ are trademarks of Microsoft Corporation. Java is a trademark of Sun Microsystems, Inc. Other product and company names mentioned herein may be the trademarks of their respective owners.

**Acquisitions Editor:** Kim Fryer

**Project Editors:** Maureen Williams Zimmerman, Anne Taussig

**Technical Editors:** Dail Magee Jr., Gary Nelson, Jean Ross, Jim Fuchs, John Conrow,  
Kurt Meyer, Robert Lyon, Roslyn Lutsch

# Contents

Introduction

v

## **Dictionary** *of Computer Terms*

■

### **Appendix A**

ASCII Character Set

520

### **Appendix B**

IBM Extended Character Set

522

### **Appendix C**

Apple Macintosh Extended Character Set

524

### **Appendix D**

EBCDIC Character Set

528

### **Appendix E**

Numeric Equivalents

534

information (such as a memory address, a data type, or an actual value). *See also* binding time, dynamic binding, static binding.

**binding time** \bīn'dēng tīm\ *n.* The point in a program's use at which binding of information occurs, usually in reference to program elements being bound to their storage locations and values. The most common binding times are during compilation (compile-time binding), during linking (link-time binding), and during program execution (run-time binding). *See also* bind, compile-time binding, link-time binding, run-time binding.

**BinHex**<sup>1</sup> \bīn'heks\ *n.* **1.** A code for converting binary data files into ASCII text so they can be transmitted via e-mail to another computer or in a newsgroup post. This method can be used when standard ASCII characters are needed for transmission, as they are on the Internet. BinHex is used most frequently by Mac users. *See also* MIME. **2.** An Apple Macintosh program for converting binary data files into ASCII text and vice versa using the BinHex code. *Compare* uuencode<sup>1</sup>, uuencode<sup>1</sup>.

**BinHex**<sup>2</sup> \bīn'heks\ *vb.* To convert a binary file into printable 7-bit ASCII text or to convert the resulting ASCII text file back to binary format using the BinHex program. *Compare* uuencode<sup>2</sup>, uuencode<sup>2</sup>.

**binomial distribution** \bī-nō'mē-əl dis-trā-byōō'shən\ *n.* In statistics, a list or a function that describes the probabilities of the possible values of a random variable chosen by means of a Bernoulli sampling process. A Bernoulli process has three characteristics: each trial has only two possible outcomes—success or failure; each trial is independent of all other trials; and the probability of success for each trial is constant. A binomial distribution can be used to calculate the probability of getting a specified number of successes in a Bernoulli process. For example, the binomial distribution can be used to calculate the probability of getting a 7 three times in 20 rolls of a pair of dice. *Also called* Bernoulli distribution.

**bionics** \bī-on'iks\ *n.* The study of living organisms, their characteristics, and the ways they function, with a view toward creating hardware that can simulate or duplicate the activities of a biological system. *See also* cybernetics.

**BIOS** \bī'ōs, B'I-O-S\ *n.* Acronym for **basic input/output system**. On PC-compatible computers, the set of essential software routines that test hardware at startup, start the operating system, and support the transfer of data among hardware devices. The BIOS is stored in read-only memory (ROM) so that it can be executed when the computer is turned on. Although critical to performance, the BIOS is usually invisible to computer users. *See also* AMI BIOS, CMOS setup, Phoenix BIOS, ROM BIOS. *Compare* Toolbox.

**bipolar** \bī'pō'lār\ *adj.* **1.** Having two opposite states, such as positive and negative. **2.** In information transfer and processing, pertaining to or characteristic of a signal in which opposite voltage polarities represent on and off, true and false, or some other pair of values. *See also* non-return to zero. *Compare* unipolar. **3.** In electronics, pertaining to or characteristic of a transistor having two types of charge carriers. *See also* transistor.

**BIS** \B'I-S\ *n.* *See* business information system.

**bistable** \bī'stā'bl\ *adj.* Of, pertaining to, or characteristic of a system or device that has two possible states, such as on and off. *See also* flip-flop.

**bistable circuit** \bī'stā'bl sər'kət\ *n.* Any circuit that has only two stable states. The transition between them must be initiated from outside the circuit. A bistable circuit is capable of storing 1 bit of information.

**bistable multivibrator** \bī'stā'bl mul-tī-vī'brā-tər, mul-tē-vī'brā-tər\ *n.* *See* flip-flop.

**BISYNC** \bī'sēnk\ *n.* Short for **binary synchronous communications protocol**. A communications standard developed by IBM. BISYNC transmissions are encoded in either ASCII or EBCDIC. Messages can be of any length and are sent in units called frames, optionally preceded by a message header. BISYNC uses synchronous transmission, in which message elements are separated by a specific time interval, so each frame is preceded and followed by special characters that enable the sending and receiving machines to synchronize their clocks. STX and ETX are control characters that mark the beginning and end of the message text; BCC is a set of characters used to verify the accuracy of transmission. *See the illustration on the next page. Also called* BSC.



**data set** \dā'tə set, dat'ə\ *n.* **1.** A collection of related information made up of separate elements that can be treated as a unit in data handling. **2.** In communications, a modem. *See also* modem.

**Data Set Ready** \dā'tə set red'ē, dat'ə\ *n.* *See* DSR.

**data sharing** \dā'tə shâr'ēng, dat'ə\ *n.* The use of a single file by more than one person or computer. Data sharing can be done by physically transferring a file from one computer to another, or, more commonly, by networking and computer-to-computer communications.

**data signal** \dā'tə sig'nəl, dat'ə\ *n.* The information transmitted over a line or circuit. It consists of binary digits and can include actual information or messages and other elements such as control characters or error-checking codes.

**data sink** \dā'tə sēnk, dat'ə\ *n.* **1.** Any recording medium where data can be stored until needed. **2.** In communications, the portion of a Data Terminal Equipment (DTE) device that receives transmitted data.

**data source** \dā'tə sōrs, dat'ə\ *n.* **1.** The originator of computer data, frequently an analog or digital data collection device. **2.** In communications, the portion of a Data Terminal Equipment (DTE) device that sends data.

**data stream** \dā'tə strēm, dat'ə\ *n.* An undifferentiated, byte-by-byte flow of data.

**data structure** \dā'tə struk'chur, dat'ə\ *n.* An organizational scheme, such as a record or array, that can be applied to data to facilitate interpreting the data or performing operations on it.

**data switch** \dā'tə swich, dat'ə\ *n.* A device in a computer system that routes incoming data to various locations.

**Data Terminal Ready** \dā'tə tər-mə-nəl red'ē, dat'ə\ *n.* *See* DTR.

**data traffic** \dā'tə traf'ik, dat'ə\ *n.* The exchange of electronic messages—control and data—across a network. Traffic capacity is measured in bandwidth; traffic speed is measured in bits per unit of time.

**data transfer** \dā'tə trans'fər, dat'ə\ *n.* The movement of information from one location to another, either within a computer (as from a disk drive to memory), between a computer and an external device (as between a file server and a computer on a network), or between separate computers.

**data transfer rate** \dā'tə trans'fər rāt, dat'ə\ *n.* *See* data rate.

**data transmission** \dā'tə tranz-mish'ən, dat'ə\ *n.* The electronic transfer of information from a sending device to a receiving device.

**data type** \dā'tə tīp, dat'ə\ *n.* In programming, a definition of a set of data that specifies the possible range of values of the set, the operations that can be performed on the values, and the way in which the values are stored in memory. Defining the data type allows a computer to manipulate the data appropriately. Data types are most often supported in high-level languages and often include types such as real, integer, floating point, character, Boolean, and pointer. How a language handles data typing is one of its major characteristics. *See also* cast, constant, enumerated data type, strong typing, type checking, user-defined data type, variable, weak typing.

**data validation** \dā'tə val'ə-dā'shən, dat'ə\ *n.* The process of testing the accuracy of data.

**data value** \dā'tə val'yōō, dat'ə\ *n.* The literal or interpreted meaning of a data item, such as an entry in a database, or a type, such as an integer, that can be used for a variable.

**data warehouse** \dā'tə wâr'hous, dat'ə\ *n.* A database, frequently very large, that can access all of a company's information. While the warehouse can be distributed over several computers and may contain several databases and information from numerous sources in a variety of formats, it should be accessible through a server. Thus, access to the warehouse is transparent to the user, who can use simple commands to retrieve and analyze all the information. The data warehouse also contains data about how the warehouse is organized, where the information can be found, and any connections between data. Frequently used for decision support within an organization, the data warehouse also allows the organization to organize its data, coordinate updates, and see relationships between information gathered from different parts of the organization. *See also* database, decision support system, server (definition 1), transparent (definition 1).

**date stamping** \dāt' stam'pēng\ *n.* A software feature that automatically inserts the current date into a document.



programs are necessary for copy-protected applications, which cannot be copied by normal operating-system commands. They typically limit the number of copies that can be installed.

**Installer** \in-stā'lər\ *n.* A program, provided with the Apple Macintosh operating system, that allows the user to install system upgrades and make bootable (system) disks.

**instance** \in'stəns\ *n.* An object, in object-oriented programming, in relation to the class to which it belongs. For example, an object *myList* that belongs to a class *List* is an instance of the class *List*. *See also* class, instance variable, instantiate, object (definition 2).

**instance variable** \in'stəns vār'ē-ə-bl\ *n.* A variable associated with an instance of a class (an object). If a class defines a certain variable, each instance of the class has its own copy of that variable. *See also* class, instance, object (definition 2), object-oriented programming.

**instantiate** \in-stān'shē-āt\ *vb.* To create an instance of a class. *See also* class, instance, object (definition 2).

**instruction** \in-struk'shən\ *n.* An action statement in any computer language, most often in machine or assembly language. Most programs consist of two types of statements: declarations and instructions. *See also* declaration, statement.

**instruction code** \in-struk'shən kōd'\ *n.* *See* operation code.

**instruction counter** \in-struk'shən coun'tər\ *n.* *See* instruction register.

**instruction cycle** \in-struk'shən sī'kl\ *n.* The cycle in which a processor retrieves an instruction from memory, decodes it, and carries it out. The time required for an instruction cycle is the sum of the instruction (fetch) time and the execution (translate and execute) time and is measured by the number of clock ticks (pulses of a processor's internal timer) consumed.

**instruction mix** \in-struk'shən miks'\ *n.* The assortment of types of instructions contained in a program, such as assignment instructions, mathematical instructions (floating-point or integer), control instructions, and indexing instructions. Knowledge of instruction mixes is important to designers of CPUs because it tells them which instructions should be shortened to yield the great-

est speed, and to designers of benchmarks because it enables them to make the benchmarks relevant to real tasks.

**instruction pointer** \in-struk'shən pōin'tər\ *n.* *See* program counter.

**instruction register** \in-struk'shən rej'ə-stər\ *n.* A register in a central processing unit that holds the address of the next instruction to be executed.

**instruction set** \in-struk'shən set'\ *n.* The set of machine instructions that a processor recognizes and can execute. *See also* assembler, microcode.

**instruction time** \in-struk'shən tīm'\ *n.* The number of clock ticks (pulses of a computer's internal timer) required to retrieve an instruction from memory. Instruction time is the first part of an instruction cycle; the second part is the execution (translate and execute) time. *Also called* I-time.

**instruction word** \in-struk'shən wərd'\ *n.* **1.** The length of a machine language instruction. **2.** A machine language instruction containing an operation code identifying the type of instruction, possibly one or more operands specifying data to be affected or its address, and possibly bits used for indexing or other purposes. *See also* assembler, machine code.

**insulator** \in'sə-lā'tər\ *n.* **1.** Any material that is a very poor conductor of electricity, such as rubber, glass,\*or ceramic. *Also called* nonconductor. *Compare* conductor, semiconductor. **2.** A device used to separate elements of electrical circuits and prevent current from taking unwanted paths, such as the stacks of ceramic disks that suspend high-voltage power lines from transmission towers.

**integer** \in'tə-jər\ *n.* **1.** A positive or negative "whole" number, such as 37, -50, or 764. **2.** A data type representing whole numbers. Calculations involving only integers are much faster than calculations involving floating-point numbers, so integers are widely used in programming for counting and numbering purposes. Integers can be signed (positive or negative) or unsigned (positive). They can also be described as long or short, depending on the number of bytes needed to store them. Short integers cover a smaller range of numbers (for example, -32,768 through 32,767) than do long integers (for example, -2,147,483,648 through 2,147,483,647). *Also called* integral number. *See also* floating-point notation.



**millions of instructions per second** \mil'yənz əv in-struk'shənz pər sek'ənd\ *n.* See MIPS.

**millisecond** \mil'ə-sek'ənd\ *n.* Abbreviated ms or msec. One thousandth of a second.

**millivolt** \mil'ə-vōlt\ *n.* Abbreviated mV. One thousandth of a volt.

**MILNET** \mil'net\ *n.* Short for **Military Network**. A wide area network that represents the military side of the original ARPANET. MILNET carries nonclassified U.S. military traffic. *See also* ARPANET. *Compare* NSFnet.

**MIMD** \MI-M-D\ *n.* Acronym for **multiple instruction, multiple data** stream processing. A category of computer architecture engaged in parallel processing in which central processing units independently fetch instructions and operate on data. *See also* architecture (definition 1), central processing unit, instruction, parallel processing. *Compare* SIMD.

**MIME** \mīm, MI-M-E\ *n.* Acronym for **Multipurpose Internet Mail Extensions**. A standard that extends the Simple Mail Transfer Protocol (SMTP) to permit data, such as video, sound, and binary files, to be transmitted by Internet e-mail without having to be translated into ASCII format first. This is accomplished by the use of MIME types, which describe the contents of a document. A MIME-compliant application sending a file, such as some e-mail programs, assigns a MIME type to the file. The receiving application, which must also be MIME-compliant, refers to a standardized list of documents that are organized into MIME types and subtypes to interpret the content of the file. For instance, one MIME type is *text*, and it has a number of subtypes, including *plain* and *html*. A MIME type of *text/html* refers to a file that contains text written in HTML. MIME is part of HTTP, and both Web browsers and HTTP servers use MIME to interpret e-mail files they send and receive. *See also* HTTP, HTTP server, Simple Mail Transfer Protocol, Web browser. *Compare* BinHex<sup>1</sup> (definition 1).

**miniaturization** \min'ə-chər-i-zā'shən\ *n.* In the development of integrated circuits, the process of reducing the size and increasing the density of transistors and other elements on a semiconductor chip. Besides providing the benefits of small size, miniaturization of electronic circuits lowers power

requirements, reduces heat, and shortens delays in the propagation of signals from one circuit element to the next. *See also* integrated circuit, integration (definition 2).

**minicomputer** \min'ē-kəm-pyōō'tər\ *n.* A mid-level computer built to perform complex computations while dealing efficiently with a high level of input and output from users connected via terminals. Minicomputers also frequently connect to other minicomputers on a network and distribute processing among all the attached machines. Minicomputers are used heavily in transaction-processing applications and as interfaces between mainframe computer systems and wide area networks. *See also* computer, mainframe computer, microcomputer, supercomputer, wide area network. *Compare* midrange computer, workstation (definition 2).

**mini-driver architecture** \min'ē-drī-vər ār'kə-tek-chur\ *n.* An architecture in Windows 3.1 and Windows 95 that uses a relatively small and simple driver, containing any additional instructions needed by a specific hardware device, to interface with the universal driver for that class of devices. *See also* driver.

**minifloppy** \min'ē-flop'ē\ *n.* A 5.25-inch floppy disk. *See also* floppy disk.

**minimize** \min'ə-mīz\ *vb.* In a graphical user interface, to hide a window without shutting down the program responsible for the window. Usually an icon, button, or name for the window is placed on the desktop; when the user clicks on the button, icon, or name, the window is restored to its previous size. *See also* graphical user interface, Minimize button, taskbar, window. *Compare* maximize.

**Minimize button** \min'ə-mīz but'ən\ *n.* In Windows 3.x, Windows 95, and Windows NT, a button in the upper right-hand corner of a window that when clicked hides the window. In Windows 3.x and Windows NT 3.5 and lower, an icon appears on the desktop that represents the window; in Windows 95 and Windows NT 4.0, the name of the window appears on the taskbar at the bottom of the desktop screen. When the icon or the name is clicked, the window is restored to its previous size. *See also* graphical user interface, taskbar, window.



**streaming tape** \strē'mēng tāp\ *n.* See tape (definition 1).

**stream-oriented file** \strēm'ōr'ē-ent-əd fīl\ *n.* A file used to store a fairly continuous series of bits, bytes, or other small, structurally uniform units.

**street price** \strēt' prīs\ *n.* The actual retail or mail-order price of a consumer hardware or software product. In most cases, the street price is somewhat lower than the "suggested retail price."

**stress test** \stres'test\ *n.* A test of a software or hardware system's functional limits, performed by subjecting the system to extreme conditions, such as peak volumes of data or extremes in temperature.

**strikethrough** \strīk'thrō\ *n.* One or more lines drawn through a selected range of text, usually to show deletion or the intent to delete, as in strikethrough.

**string** \strēng\ *n.* A data structure composed of a sequence of characters usually representing human-readable text.

**string variable** \strēng'vâr'ē-ə-bl\ *n.* An arbitrary name assigned by the programmer to a string of alphanumeric characters and used to reference that entire string. See also string.

**strobe** \strōb\ *n.* A timing signal that initiates and coordinates the passage of data, typically through an input/output (I/O) device interface, such as a keyboard or printer.

**stroke** \strōk\ *n.* **1.** In data entry, a keystroke—a signal to the computer that a key has been pressed. **2.** In typography, a line representing part of a letter. **3.** In paint programs, a "swipe" of the brush made with the mouse or keyboard in creating a graphic. **4.** In display technology, a line created as a vector (a path between two coordinates) on a vector graphics display (as opposed to a line of pixels drawn dot by dot on a raster graphics display).

**stroke font** \strōk'font\ *n.* A font printed by drawing a combination of lines rather than by filling a shape, as with an outline font. Compare outline font.

**stroke weight** \strōk'wāt\ *n.* The width, or thickness, of the lines (strokes) that make up a character. See also font.

**stroke writer** \strōk'rītər\ *n.* In video, a display unit that draws characters and graphic images as sets of strokes—lines or curves connecting

points—rather than as sets of dots, as on a typical raster-scan monitor. See also vector graphics.

**strong typing** \strong'tī'pēng\ *n.* A characteristic of a programming language that does not allow the program to change the data type of a variable during program execution. See also data type, variable. Compare weak typing.

**structure** \struk'chur\ *n.* **1.** The design and composition of a program, including program flow, hierarchy, and modularity. **2.** A collection of data elements. See also data structure.

**structured graphics** \struk'churd graf'iks\ *n.* See object-oriented graphics.

**structured programming** \struk'churd prō'gram-ēng\ *n.* Programming that produces programs with clean flow, clear design, and a degree of modularity or hierarchical structure. See also modular programming, object-oriented programming. Compare spaghetti code.

**structured query language** \struk'churd kwēr'ē lang-wəj, kwâr'ē\ *n.* A database sublanguage used in querying, updating, and managing relational databases—the de facto standard for database products. Acronym: SQL (sē'kwəl, S'Q-L).

**structured walkthrough** \struk'churd wāk'thrō\ *n.* **1.** A meeting of programmers working on different aspects of a software development project, in which the programmers attempt to coordinate the various segments of the overall project. The goals, requirements, and components of the project are systematically reviewed in order to minimize the error rate of the software under development. **2.** A method for examining a computer system, including its design and implementation, in a systematic fashion.

**STT** \S'T-T\ *n.* See Secure Transaction Technology.

**stub** \stub\ *n.* A routine that contains no executable code and that generally consists of comments describing what will eventually be there; it is used as a placeholder for a routine to be written later. Also called dummy routine. See also top-down programming.

**StuffIt** \stuf'it\ *n.* A file compression program originally written for the Apple Macintosh, used for storing a file on one or more disks. Originally shareware, StuffIt is now a commercial product for Macs and PCs that supports multiple compression techniques and allows file viewing. StuffIt files can



rate is measured in units of information per unit of time—for example, bits per second or characters per second—and can be measured either as a raw rate, which is the maximum transfer speed, or as an average rate, which includes gaps between blocks of data as part of the transmission time.

**transfer statement** \trans'fər stāt'mənt\ *n.* A statement in a programming language that transfers the flow of execution to another location in the program. *See also* branch instruction, CALL statement, GOTO statement, jump instruction.

**transfer time** \trans'fər tīm\ *n.* The time elapsed between the start of a data transfer operation and its completion.

**transform** \trans'fōrm\ *vb.* **1.** To change the appearance or format of data without altering its content; that is, to encode information according to predefined rules. **2.** In mathematics and computer graphics, to alter the position, size, or nature of an object by moving it to another location (translation), making it larger or smaller (scaling), turning it (rotation), changing its description from one type of coordinate system to another, and so on.

**transformer** \trans'fōr'mər\ *n.* A device used to change the voltage of an alternating current signal or to change the impedance of an alternating current circuit. *See the illustration.*



*Transformer.*

**transient** \tran'zhənt, tran'zē-ənt\ *adj.* **1.** Fleeting, temporary, or unpredictable. **2.** Of or pertaining to the region of memory used for programs, such as applications, that are read from disk storage and that reside in memory temporarily until they are replaced by other programs. In this context, *transient* can also refer to the programs themselves. **3.** In electronics, of or pertaining to a short-lived, abnormal, and unpredictable increase in power supply, such as a voltage spike or surge. *Transient time* is the interval during which a change in current or voltage is building up or decaying.

**transient suppressor** \tran'zhənt su-pres'ər, tran'zē-ənt\ *n.* A circuit designed to reduce or eliminate unwanted electrical signals or voltages.

**transistor** \tran'zi'stər\ *n.* Short for **transfer resistor**. A solid-state circuit component, usually with three leads, in which a voltage or a current controls the flow of another current. The transistor can serve many functions, including those of amplifier, switch, and oscillator, and is a fundamental component of almost all modern electronics. *See the illustration on the next page. See also* base (definition 3), FET, NPN transistor, PNP transistor.

**transistor-transistor logic** \tran-zī'stər-tran-zī'stər loj-ik\ *n.* A type of bipolar circuit design that utilizes transistors connected to each other either directly or through resistors. Transistor-transistor logic offers high speed and good noise immunity and is used in many digital circuits. A large number of transistor-transistor logic gates can be fabricated on a single integrated circuit. *Acronym:* TTL (T-T-L).

**translate** \tranz'lāt\ *vb.* **1.** In programming, to convert a program from one language to another. Translation is performed by special programs such as compilers, assemblers, and interpreters. **2.** In computer graphics, to move an image in the "space" represented on the display, without turning (rotating) the image.

**translated file** \tranz'lā-təd fīl\ *n.* A file containing data that has been changed from binary (8-bit) format to ASCII (7-bit) format. BinHex and uuencode both translate binary files into ASCII. Such translation is necessary to transmit data through systems (such as e-mail) that may not preserve the eighth bit of each byte. A translated file must be decoded to its binary form before being used. *See also* BinHex, uuencode.

**translator** \tranz'lā-tər\ *n.* A program that translates one language or data format into another.

**transmission channel** \tranz-mish'an chan'əl\ *n.* *See* channel.

**Transmission Control Protocol/Internet Protocol** \tranz'mish'an kən-trōl' prō-tə-kol-in'tər-net prō-tə-kol\ *n.* *See* TCP/IP.

**transmit** \tranz-mit\ *vb.* To send information over a communications line or a circuit. Computer transmissions can take place in the ways listed on the next page.

owned or maintained by individual users of the computer system. These subdirectories can contain files and additional subdirectories. Typically, /usr directories are used in UNIX systems and can be found on many FTP sites. *See also* FTP site.

**USRT** \U`S-R-T` \ *n.* Acronym for **universal synchronous receiver-transmitter**. A module, usually composed of a single integrated circuit, that contains both the receiving and transmitting circuits required for synchronous serial communication. *Compare* UART.

**UTC** \U`T-C` \ *n.* *See* Universal Time Coordinate.

**utility** \yōō-til`ə-tē \ *n.* A program designed to perform a particular function; the term usually refers to software that solves narrowly focused problems or those related to computer system management. *See also* application.

**utility program** \yōō-til`ə-tē pro`gram \ *n.* A program designed to perform maintenance work on the system or on system components (e.g., a storage backup program, disk and file recovery program, or resource editor).

**UTP** \U`T-P` \ *n.* Acronym for **unshielded twisted pair**. A cable containing one or more twisted pairs of wires without additional shielding. UTP is more flexible and takes up less space than shielded twisted-pair (STP) cable but has less bandwidth. *See the illustration.* *See also* twisted-pair cable. *Compare* STP.



**UTP.**

**.uu** \dot`U-U` \ *n.* The file extension for a binary file that has been translated into ASCII format

using uuencode. *Also called* .uud. *See also* ASCII, binary file, uuencode. *Compare* .uue.

**UUCP** \U`U-C-P` \ *n.* Acronym for **UNIX-to-UNIX Copy**. A set of software programs that facilitate transmission of information between UNIX systems using serial data connections, primarily the public switched telephone network. *See also* uupc.

**.uud** \dot`U-U-D` \ *n.* *See* .uu.

**uudecode**<sup>1</sup> \U-U`dē-kōd` \ *n.* A UNIX program that converts a uuencoded file back into its original binary format. This program (along with uuencode) allows binary data, such as images or executable code, to be disseminated through e-mail or newsgroups. *Compare* uuencode<sup>1</sup>.

**uudecode**<sup>2</sup> \U-U`dē-kōd` \ *vb.* To transform a uuencoded file back into its binary original using the uudecode program. *Compare* uuencode<sup>2</sup>.

**.uue** \dot`U-U-E` \ *n.* The file extension for a file that has been decoded from ASCII format back into binary format using uudecode. *See also* ASCII, binary file, uudecode<sup>1</sup>.

**uuencode**<sup>1</sup> \U-U`en-kōd` \ *n.* A UNIX program that converts a binary file, in which all 8 bits of every byte are significant, into printable 7-bit ASCII characters without loss of information. This program (along with uudecode) allows binary data, such as images or executable code, to be disseminated through e-mail or newsgroups. A file thus encoded is one-third again as long as the original. *Compare* uudecode<sup>1</sup>.

**uuencode**<sup>2</sup> \U-U`en-kōd` \ *vb.* To transform a binary file into printable 7-bit ASCII text using the uuencode program. *Compare* uudecode<sup>2</sup>.

**uupc** \U`U-P-C` \ *n.* The version of UUCP for IBM PCs and PC-compatibles running DOS, Windows, or OS/2. This version is a collection of programs for copying files to, logging in to, and running programs on remote networked computers. *See also* UUCP.

**.uy** \dot`U-Y` \ *n.* On the Internet, the major geographic domain specifying that an address is located in Uruguay.

**.uz** \dot`U-Z` \ *n.* On the Internet, the major geographic domain specifying that an address is located in Uzbekistan.



The manuscript for this book was prepared and submitted to Microsoft Press in electronic form. Text files were originally prepared using Microsoft Access 95. Text composition by Publication Services in Garamond with display type in Garamond Bold, using FrameMaker 5.0 for Windows 95. Composed pages were delivered to the printer in electronic prepress files.

*Lexicographer*  
Robert Costello

### **Microsoft Press**

*Cover Designer*  
Gregory Erickson

*Interior Graphic Designer*  
Kim Eggleston

*Interior Illustrator*  
Travis Beaven

*Principal Proofreader/Copy Editor*  
Paula Thurman

### **Publication Services**

#### **Development**

*Text Consultants*  
Thomas P. Magliery  
Terrence M. McLaren  
Larry S. Jackson  
Judson D. Weeks  
Annette B. Jackson  
Thomas A. Jackson  
John Ross

*Pronunciation Guide Consultants*  
Gerard Dalgish  
Sharon Goldstein

*Pronunciations*  
Julia S. Dalgish  
Joel Gordon

#### *Writers*

Jerome Colburn  
Robert Howecton  
Ilana Kingsley  
Thomas A. Long  
David Mason

#### **Production**

*Copy Editor*  
David Mason

*Senior Project Coordinator*  
Rhonda Zachmeyer

*Customer Service Representative*  
Karla Wright