

Data Storage and Representation

DATA TYPES

Data today come in different forms including numbers, text, audio, images, and video (Figure 6.1).

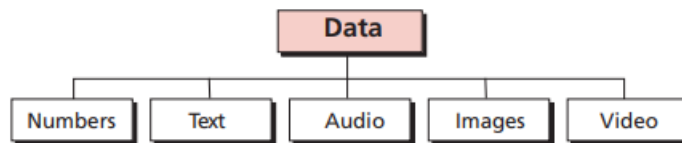


Figure (6.1) Data type

Data inside the computer

All data types are transformed into a uniform representation when they are stored in a computer and transformed back to their original form when retrieved. This universal representation is called a bit pattern.

Bits

A bit (binary digit) is the smallest unit of data that can be stored in a computer and has a value of 0 or 1. A bit represents the state of a device that can take one of two states. For example, a switch can be **on** or **off**.

Bit patterns

To represent different types of data, we use a bit pattern, a sequence, or as it is sometimes called, a string of bits. Figure 6.2 shows a bit pattern made up of sixteen bits. It is a combination of sixteen 0s and 1s. This means that if we need to store a bit pattern made of sixteen bits, we need sixteen electronic switches. If we need to store 1000 bit patterns, each sixteen bits long, we need 16 000 switches, and so on. By tradition a bit pattern with eight bits is called a byte. Sometimes the term word is used to refer to a longer bit pattern.

1 0 0 0 1 0 1 0 1 1 1 1 1 1 0 1

Figure (6.2) Bit patterns: a piece of data belonging to different data types can be stored as the same pattern in the memory.

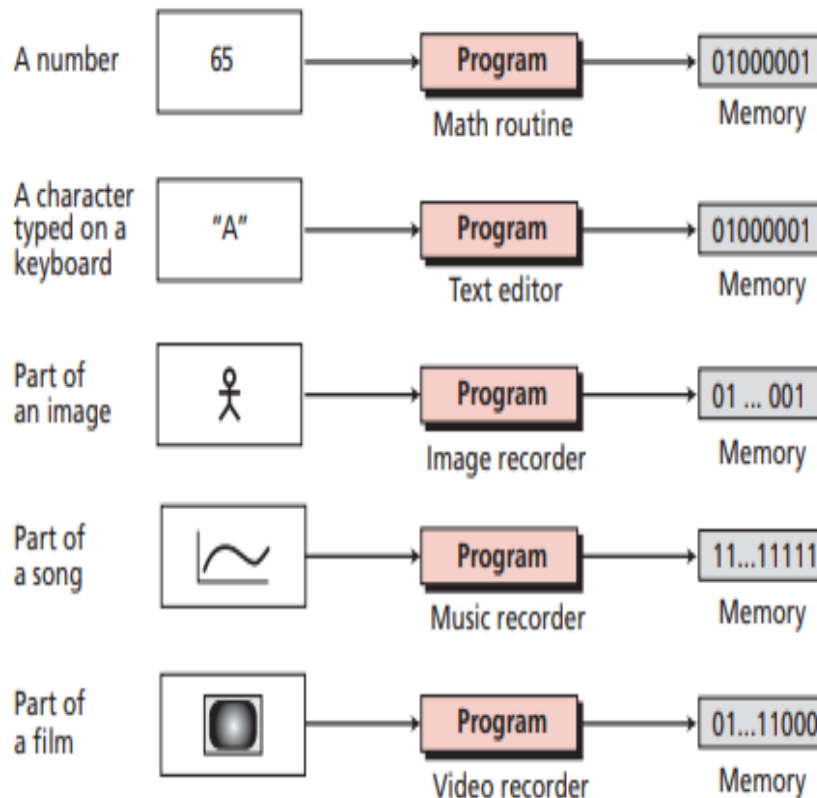


Figure (6.3) Storage of different data types

If we are using a text editor (a word processor), the character A typed on the keyboard can be stored as the 8-bit pattern 01000001. The same 8-bit pattern can represent the number 65 if we are using a mathematical routine. Moreover, the same pattern can represent part of an image, part of a song, or part of a scene in a film. The computer's memory stores all of them without recognizing what type of data they represent figure (6.3).

1- STORING NUMBERS

A number is changed to the binary system before being stored in the computer memory, For the decimal point, computers use two different representations: fixed-

point and floatingpoint. The first is used to store a number as an integer—without a fractional part, the second is used to store a number as a real—with a fractional part.

Storing integers

Integers are whole numbers (numbers without a fractional part). For example, 134 and -125 are integers, whereas 134.23 and -0.235 are not. An integer can be thought of as a number in which the position of the decimal point is fixed: the decimal point is to the right of the least significant (rightmost) bit.

Note ((An integer is normally stored in memory using fixed-point representation))

Unsigned representation

An unsigned integer is an integer that can never be negative and can take only 0 or positive values. Its range is between 0 and positive infinity. However, since no computer can possibly represent all the integers in this range, most computers define a constant called the maximum unsigned integer, which has the value of $(2^n - 1)$ where n is the number of bits allocated to represent an unsigned integer.

Storing unsigned

integers An input device stores an unsigned integer using the following steps:

1. The integer is changed to binary.
2. If the number of bits is less than n , 0s are added to the left of the binary integer so that there is a total of n bits. If the number of bits is greater than n , the integer cannot be stored. A condition referred to as overflow will occur, which we discuss later.

Example 1

Store 7 in an 8-bit memory location using unsigned representation.

Solution

First change the integer to binary, $(111)_2$. Add five 0s to make a total of eight bits, $(0000111)_2$. The integer is stored in the memory location. Note that the subscript 2 is used to emphasize that the integer is binary, but the subscript is not stored in the computer:

Change 7 to binary → 1 1 1
 Add five bits at the left → 0 0 0 0 0 1 1 1

Example 2

Store 258 in a 16-bit memory location.

Solution

First change the integer to binary $(100000010)_2$. Add seven 0s to make a total of sixteen bits $(0000000100000010)_2$. The integer is stored in the memory location:

Change 258 to binary → 1 0 0 0 0 0 0 1 0
 Add seven bits at the left → 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0

Storing reals

A real is a number with an integral part and a fractional part. For example, 23.7 is a real number—the integral part is 23 and the fractional part is $7/10$. Although a fixed-point representation can be used to represent a real number, the result may not be accurate or it may not have the required precision. The solution for maintaining accuracy or precision is to use floating-point representation.

2- STORING TEXT

A section of text in any language is a sequence of symbols used to represent an idea in that language. For example, the English language uses 26 symbols (A, B, C, ..., Z) to represent uppercase letters, 26 symbols (a, b, c, ..., z) to represent lowercase letters, ten symbols (0, 1, 2, ..., 9) to represent numeric characters (not actual numbers—numbers are treated separately, and symbols (., ?, :, ; , ..., !) to represent punctuation. Other symbols such as blank, newline, and tab are used for text alignment and readability. We can represent each symbol with a bit pattern. In other words, text such as 'CATS', which is made up from four symbols, can be represented as four n-bit patterns, each pattern defining a single symbol (Figure 6.4).

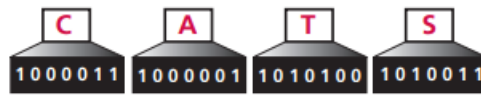


Figure 6.4 Representing symbols using bit patterns

3-STORING AUDIO

Audio is a representation of sound or music. Audio, by nature, is different from the numbers or text we have discussed so far. Text is composed of countable entities (characters): we can count the number of characters in text. Text is an example of digital data. In contrast, audio is not countable. Audio is an entity that changes with time—we can only measure the intensity of the sound at each moment. When we discuss storing audio in computer memory, we mean storing the intensity of an audio signal, such as the signal from a microphone, over a period of time: one second, one hour.

4-STORING IMAGES

Images are stored in computers using two different techniques: raster graphics and vector graphics

4-STORING VIDEO

Video is a representation of images (called frames) over time. A movie consists of a series of frames shown one after another to create the illusion of motion. In other words, video is the representation of information that changes in space (single image) and in time (a series of images). So, if we know how to store an image inside a computer, we also know how to store video: each image or frame is transformed into a set of bit patterns and stored. The combination of the images then represents the video. Today video is normally compressed.