Chapter 2

Data Representation

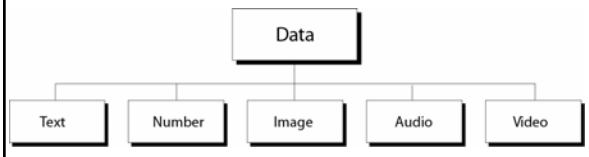
OBJECTIVES

- Define data types.
- Visualize how data are stored inside a computer.
- Understand the differences between text, numbers, images, video, and audio.
- Work with hexadecimal and octal notations.

2.1 DATA TYPES

Figure 2-1

Different types of data



Examples

- MatLab -> numbers
- MS Word -> text
- Photoshop -> images
- iTunes, WinAmp -> audio
- Media Player -> video
- ***Any other examples?***

Note:

The computer industry uses the term "multimedia" to define information that contains numbers, text, images, audio, and video.

2.2 DATA INSIDE THE COMPUTER

Question?

How can we handle different data types?

What is the language that computer can understand?

Figure 2-2

Bit pattern

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

Bits and Bytes

- A bit is the smallest unit of data (0 or 1)
- Byte = 8 bits
- 1 Kilobyte = 1024 Bytes
- 1 Megabyte = 1024 Kilobytes
- 1 Gigabyte = 1024 Megabytes
- 1 Terabyte = 1024 Gigabytes
- 1 Petabyte = 1024 Terabytes

Questions?

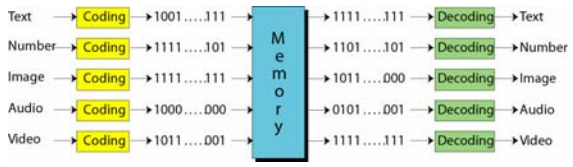
- How many bits are in 1Kilobyte? (*possible type of question on the first midterm*)
- How many bits are in 1 Megabyte?

Questions?

- Your textbook for this course is about 1 Mega Byte in size !!!
- What units do we use to measure the capacity of your hard drive?
- What units do we use to measure the capacity of RAM (main memory)?

Figure 2-3

Examples of bit patterns



2.3 REPRESENTING DATA

Representing Data: Text

- Text in any language – sequence of symbols
- Number of distinct symbols is limited
- We can represent each symbol with a distinct bit pattern
- Thus we can encode text using bit patterns

Figure 2-4

Representing symbols using bit patterns

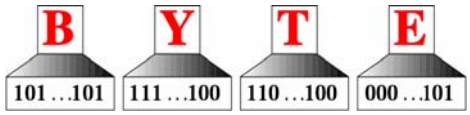


Table 2.1 Number of symbols and bit pattern length

Number of Symbols	Bit Pattern Length
2	1
4	2
8	3
16	4
…	…
128	7
256	8
…	…
65,536	16

Representing Data: Text

- Logarithmic relationship
- Log_2(Number_of_symbols)

Data Representation: Text

- There are many variants of bit patters to represent text symbols
- ASCII (American Standard Code for Information Interchange)
- This code uses 7 bits to encode every symbol
- There are 2^7=128 different symbols that can be represented using ASCII

Figure 2-5

Representation of the word "BYTE" in ASCII code

Representing Data: Text

- Extended ASCII (8 bits)
- Unicode (16 bits)
- ISO (32 bits = 4,294,967,296 symbols)

Representing Data: Numbers

- Numbers are represented using the binary system
- ASCII **is NOT** used to represent numbers
- We'll talk about it in details during the next lecture

Figure 2-6

Image representation methods

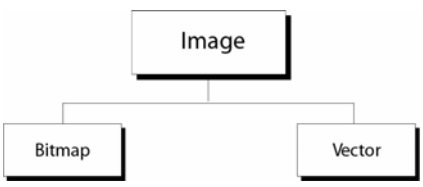


Figure 2-7

Bitmap graphic method of a black-and-white image

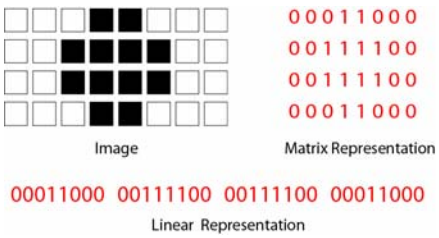
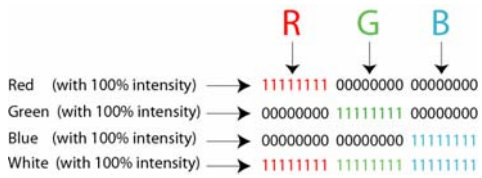


Figure 2-8

Representation of color pixels

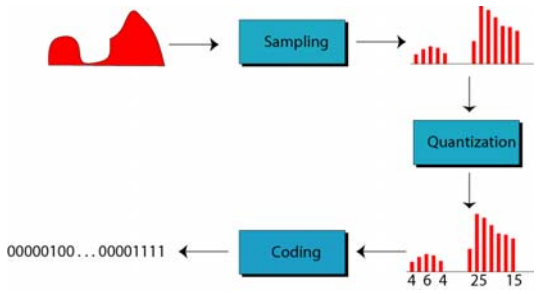


Vector Graphic

- Bitmaps -> exact bit patterns are saved
- Bitmaps -> resize -> grainy look
- Vector Graphic -> mathematical formulas -> lines and curves
- Vector Graphic -> better quality of pictures after rescaling

Figure 2-9

Audio representation



2.4 HEXADECIMAL NOTATION

Hexadecimal Notation

- In hexadecimal notation there are 16 symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
- It helps writing a long streams of 0's and 1's
- It's very important in Computer Science

Note:

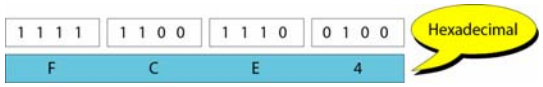
A 4-bit pattern can be represented by a hexadecimal digit, and vice versa.

Table 2.2 Hexadecimal digits

Bit Pattern	Hex Digit	Bit Pattern	Hex Digit
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

Figure 2-10

Binary to hexadecimal and hexadecimal to binary transformation



Example 1

Show the hexadecimal equivalent of the bit pattern 1100 1110 0010.

Solution

Each group of 4 bits is translated to one hexadecimal digit. The equivalent is xCE2.

Example 2

Show the hexadecimal equivalent of the bit pattern 0011100010.

Solution

Divide the bit pattern into 4-bit groups (from the right). In this case, add two extra 0s at the left to make the number of bits divisible by 4. So you have 000011100010, which is translated to x0E2.

Example 3

What is the bit pattern for x24C?

Solution

Write each hexadecimal digit as its equivalent bit pattern to get 001001001100.

2.5 OCTAL NOTATION

Note:

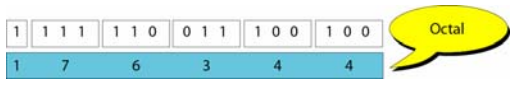
A 3-bit pattern can be represented by an octal digit, and vice versa.

Table 2.3 Octal digits

Bit Pattern	Oct Digit	Bit Pattern	Oct Digit
000	0	100	4
001	1	101	5
010	2	110	6
011	3	111	7

Figure 2-11

Binary to octal and octal to binary transformation



Example 4

Show the octal equivalent of the bit pattern 101110010.

Solution

Each group of 3 bits is translated to one octal digit. The equivalent is 0562, o562, or 562_8.

Example 5

Show the octal equivalent of the bit pattern 1100010.

Solution

Divide the bit pattern into 3-bit groups (from the right). In this case, add two extra 0s at the left to make the number of bits divisible by 3. So you have 001100010, which is translated to 142_8.

Example 6

What is the bit pattern for 24_8?

Solution

Write each octal digit as its equivalent bit pattern to get 010100.

Summary

- Types of data: text, numbers, images, audio, video
- All data types are converted into bit patterns
- Bit is the smallest unit of data
- Byte = 8 Bits
- ASCII (8 bits) is a popular code for symbols
- Unicode (16 bits) and ISO (32 bits)

Summary

- Images use bitmap and vector representation
- Audio is transformed using sampling, quantization, coding
- Video data are a set of sequential images

Next Lecture

We will concentrate on Number Representation in the Computer