# INTRODUCTION TO COMPUTER SCIENCE

Dr. Yasmine El-Glaly

Fall 2013

# Review

- Representing Info
- Integers:

Base 10 — $234 = 2 \cdot 10^2 + 3 \cdot 10^1 + 4 \cdot 10^0 = \sum_{i=0}^{2} d_i \cdot 10^i$
Generally $d_{k-1} \ldots d_1 d_0 = \sum_{i=0}^{k-1} d_i \cdot 10^i$.

Base 2 — $11101100 =$
$1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = \sum_{i=1}^{7} b_i \cdot 2^i$
Generally $b_{k-1} \ldots b_1 b_0 = \sum_{i=0}^{k-1} b_i \cdot 2^i$.

## Storing Integers

Data Compression

Communication Errors

- The most popular system for representing integers within today’s computers is **two’s complement** notation

- Uses a fixed number of bits to represent each of the values in the system

# Storing Integers

Data Compression
Communication Errors

**a. Using patterns of length three**

| Bit pattern | Value represented |
|---|---|
| 011 | 3 |
| 010 | 2 |
| 001 | 1 |
| 000 | 0 |
| 111 | −1 |
| 110 | −2 |
| 101 | −3 |
| 100 | −4 |

- Starting with a string of 0s and then counting in binary until the pattern consisting of a single 0 followed by 1s is reached
- Negative values are obtained by starting with a string of 1s and then counting backward in binary until the pattern consisting of a single 1 followed by 0s is reached

# Storing Integers

Data Compression

Communication Errors

**a. Using patterns of length three**

| Bit pattern | Value represented |
|---|---|
| 011 | 3 |
| 010 | 2 |
| 001 | 1 |
| 000 | 0 |
| 111 | −1 |
| 110 | −2 |
| 101 | −3 |
| 100 | −4 |

**b. Using patterns of length four**

| Bit pattern | Value represented |
|---|---|
| 0111 | 7 |
| 0110 | 6 |
| 0101 | 5 |
| 0100 | 4 |
| 0011 | 3 |
| 0010 | 2 |
| 0001 | 1 |
| 0000 | 0 |
| 1111 | −1 |
| 1110 | −2 |
| 1101 | −3 |
| 1100 | −4 |
| 1011 | −5 |
| 1010 | −6 |
| 1001 | −7 |
| 1000 | −8 |

## Storing Integers

Data Compression

Communication Errors

**a. Using patterns of length three**

| Bit pattern | Value represented |
|---|---|
| 011 | 3 |
| 010 | 2 |
| 001 | 1 |
| 000 | 0 |
| 111 | −1 |
| 110 | −2 |
| 101 | −3 |
| 100 | −4 |

- In a two's complement system, the leftmost bit of a bit pattern indicates the sign of the value represented. Thus, the leftmost bit is often called the **sign bit**

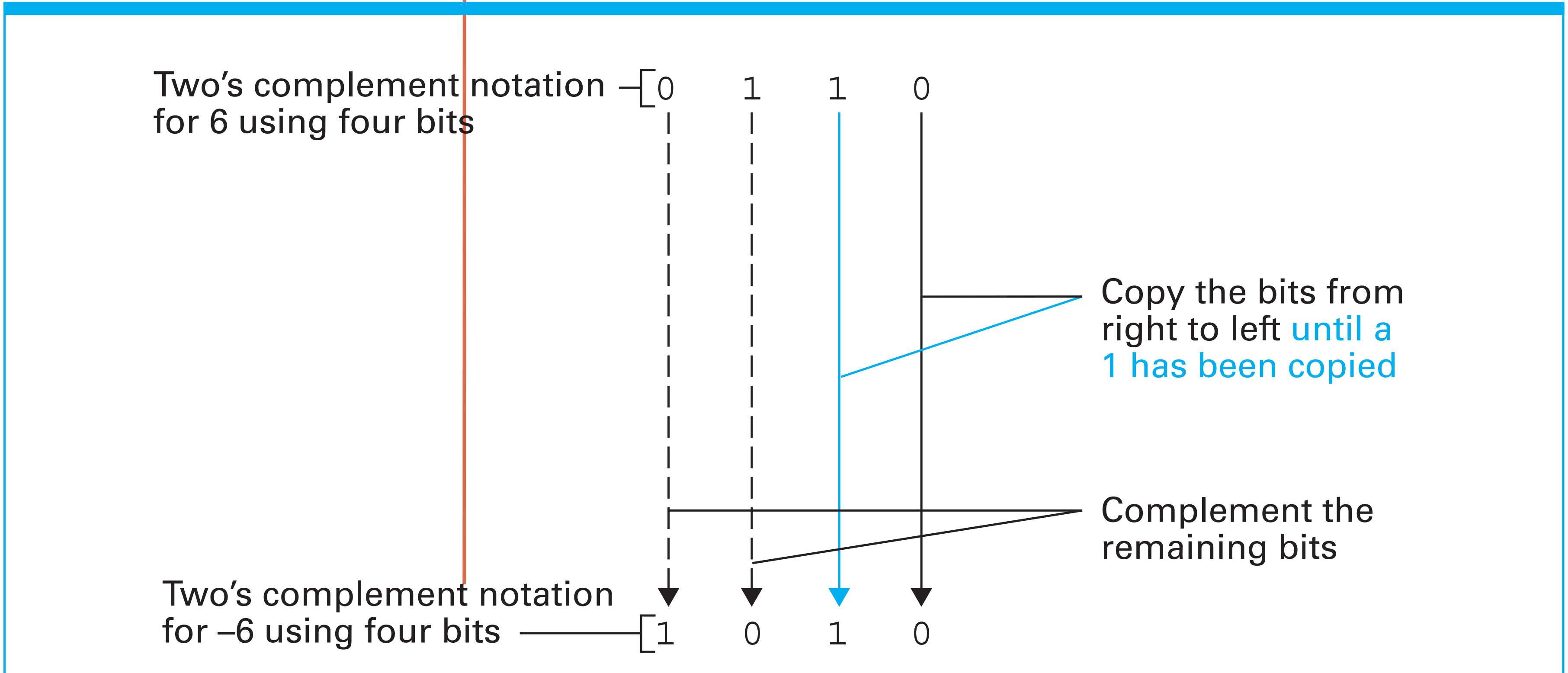
## Storing Integers

Data Compression

Communication Errors

- An algorithm for converting back and forth between +ve and -ve values of the same magnitude

**Figure 1.22** Encoding the value −6 in two's complement notation using 4 bits



## Storing Integers

Data Compression

Communication Errors

- Addition in Two's Complement Notation
  - Same algorithm that we used for binary addition
  - Any extra bit generated on the left of the answer by a final carry must be truncated

## Storing Integers

Data Compression

Communication Errors

- Example

| Problem in base ten | Problem in two's complement | Answer in base ten |
| --- | --- | --- |
| 3 + 2 | 0011 + 0010 = 0101 | 5 |
| −3 + −2 | 1101 + 1110 = 1011 | −5 |
| 7 + −5 | 0111 + 1011 = 0010 | 2 |

## Storing Integers

Data Compression

Communication Errors

- Overflow problem
  - When using two's complement with patterns of 4 bits, the largest +ve integer that can be represented is 7, and the most -ve integer is -8
  - The value 9 can not be represented, which means that we cannot hope to obtain the correct answer to the problem 5 + 4. In fact, the result would appear as -7
  - This phenomenon is called **overflow**

## Storing Integers

Data Compression

Communication Errors

- Overflow problem
  - Today, it is common to use patterns of 32 bits for storing values in two's complement notation, allowing for positive values as large as 2,147,483,647 to accumulate before overflow occurs
  - The point is that computers can make mistakes. So, the person using the machine must be aware of the dangers involved

Storing Integers

Data Compression

Communication Errors

- Data Compression
  - For the purpose of storing or transferring data, it is often helpful (and sometimes mandatory) to reduce the size of the data
  - The technique for accomplishing this is called **data compression**

Storing Integers

**Data Compression**

Communication Errors

- Data compression
  - Generic Data Compression Techniques
- Compressing images
- Compressing audio and video

Storing Integers

## Data Compression

Communication Errors

- Generic Data Compression Techniques
  - **Lossless** schemes
    - **No loss in the info**
  - **Lossy** schemes
    - May lead to the loss of info
    - Provides more compression

Storing Integers

Data Compression

Communication Errors

- **Run-length encoding**
  - Replace sequences of identical data elements with a code indicating the element that is repeated and the number of times it occurs in the sequence
  - For example, less space is required to indicate that a bit pattern consists of 253 ones, followed by 118 zeros, followed by 87 ones than to actually list all 458 bits.

Storing Integers

**Data Compression**

Communication Errors

- **Run-length encoding**

visualization

Storing Integers

**Data Compression**

Communication Errors

- **Differential Encoding**
  - Record the differences between consecutive data units rather than entire units
  - Each unit is encoded in terms of its relationship to the previous unit
  - Can be implemented in either lossless or lossy form
- Dictionary encoding: (can be lossy)
  - Lempel-Ziv methods: most popular for lossless — adaptive dictionary encoding
  - Lempel-Ziv-Welch (LZW): used a lot - GIF

Storing Integers

Data Compression

Communication Errors

- Images Compression
  - image consists of dots — pixels
    - 0—white; 1—black
    - colors — use more bits —
  - red, green, blue components
  - 3 bytes per pixel
  - example: 1024 × 1024 pixels
  - need to compress

Storing Integers

## Data Compression

Communication Errors

- Images
  - GIF — Graphic Interchange Format
    - ◆ allows only 256 colors
  - PNG — Portable Network Graphic
  - JPEG — photographs
- Audio and Video
  - MPEG — Motion Picture Experts Group
  - MP3/MP4 most common for audio
- For audio/video — use properties of human hearing and sight

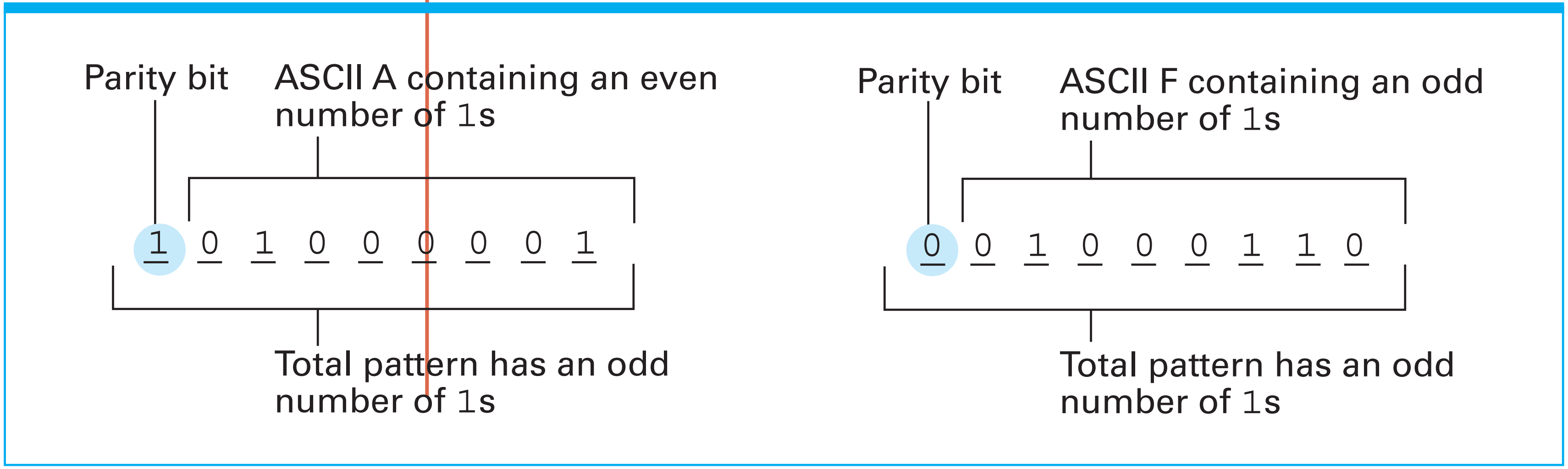
Storing Integers

Data Compression

Communication Errors

- Reasons of data errors
  - Transferring
  - Malfunctioning circuit
  - Corrupt storage
- How to detect errors?
  - Parity bits
    - Odd, Even

**Figure 1.28** The ASCII codes for the letters A and F adjusted for odd parity



# How to detect and even correct errors?

- Checksums (hashing or parity)
- Hamming distance – number of different bits
- 01010101 and 11010100
- Hamming distance 2

# Assignment

- Read the rest of chapter 1
- Exercises:
- 1
- 7
- 10
- 12
- 16
- 24
- 26
- 32
- 51


Thanks!