

## computer Basic

computer : الكمبيوتر is a device that receives, stores, and processes data

.Data: raw facts representing objects and events البيانات : الحقائق الأولية التي تمثل الاشياء والاحداث

.Information: data that is organized, meaningful and useful المعلومات : بيانات منظمة وذات مغزى ومفيدة

القدرة على التخزين  
 Fundamental Characteristics of Computers : 1- speed Reliability\*2 capability Storage\*3  
 خصائص الكمبيوتر

Diyaa Suboh

computer system components المكونات المادية  
 1. Hardware: the physical components of a computer system  
 e.g., monitor, keyboard, mouse, hard drive

2. Software : the programs that execute on the computer المكونات البرمجية  
 e.g., word processing program, Web browser

3 People: المبرمج : يكتب البرمجيات  
 1. Programmer: writes software

2. End-User: purchases and uses software المستخدم النهائي : عمليات الشراء و الاستخدام للبرنامج

### انواع الحواسيب Computer Classes

- 1- computers Personal حواسيب شخصية
- 2- computers Portable أجهزة الكمبيوتر المحمولة
- 3- Servers الخوادم
- 4- Computers Super
- 5- Devices Handheld الأجهزة المحمولة
- 6- Systems Embedded الأنظمة المنظمة

### Types of Computers

- super computers : powerful but expensive (e.g., weather forecasting, engineering design and modeling)
- desktop computers : less powerful but affordable (e.g., email, Web browsing, document processing)
- laptop computers: similar functionality to desktops

### Hardware

- 1 Control Unit تنفيذ تعليمات البرنامج المخزن  
 1. Directs the computer system to execute stored program instructions  
 2. Communicate with memory and ALU  
 3. Sends data and instructions from secondary storage to memory as needed  
 يرسل البيانات والتعليمات من التخزين الثانوي إلى الذاكرة حسب الحاجة.

### 1 - cpu

2. Arithmetic Logic Unit  
 1. Arithmetic operation  
 2. logic operations:  
 e.g Addition, Subtraction , Multiplication, Division (Equal, Less than, Greater than...)

3. Registers  
 1. High-speed temporary storage areas مناطق تخزين مؤقتة عالية السرعة  
 2. Storage locations located within the CPU مواقع تخزين داخل المعالج  
 3. ❖ Work under direction of control unit العمل تحت اشراف الوحدة المعالجة المركزية  
 4. □ Accept, hold, and transfer instructions or data قبول و حفظ و نقل البيانات  
 5. Keep track of where the next instruction to be executed or needed data is stored تعقب مكان تخزين البيانات او التعليمات المطلوب تنفيذها

### 2- Memory : is that part of a computer that stores programs and data

(modern computers are digital devices, meaning they store and process information as binary digits (bits))

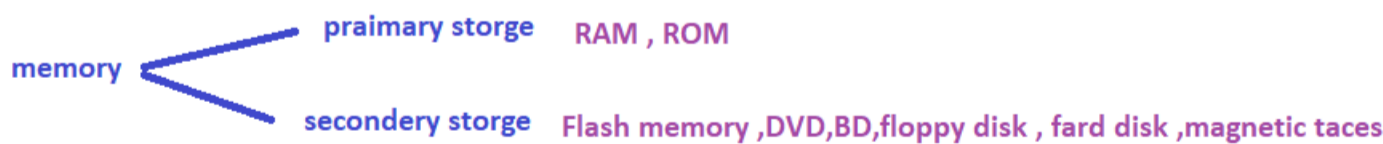
الكمبيوتر الحديث هي اجهزة رقمية مما انها تخزن المعلومات و تعالجها كارقام ثنائية

. two values of a bit are written as 0 and 1, but the values could just as easily be represented as off and on, open and

closed, volts and no volts. الخ , off and on , open and closed , volts and no volts : مثل منكتب البت رقمين 1 :: 0 ويتم تمثيل البيانات بسهولة

.memory capacity is usually specified in bytes- عادة ما تكون بشكل البايت

terabyte = 1024 gigabyte / gigabyte = 1024 megabyte / megabyte = 1024 kilobyte / 1 kilobyte = 1024 byte / 1 bits : 8 byte ملاحظات هامة



| ROM  | RAM   |
|--|---|
| Read only memory (ذاكرة القراءة فقط)   | Random Access Memory (ذاكرة الوصول العشوائي)  |
| Non-volatile (غير متطاير)  | Volatile (متطايرة... متطايرة)   |
| Permanent storage (التخزين الدائم)   | Temporary storage (التخزين المؤقت)  |
| Read only (للقراءة فقط)  | Read and Write (القراءة و الكتابة)  |
| Stores the program required to initially boot the computer<br>يخزن البرنامج المطلوب لبدء تشغيل الكمبيوتر | Allows the computer to read data quickly to run applications<br>يسمح للكمبيوتر بقراءة البيانات بسرعة لتشغيل التطبيقات |

|   |
|---|
| in put device : Keyboard mouse<br>scannar mic sensors |
| out put device : screen prenter<br>speakers camara    |

**Software** refers to the programs that execute on that hardware

program is a collection of instructions for the computer to carry out in order to complete some task مجموعة من التعليمات التي يجب على الكمبيوتر تنفيذها لكمال بعض المهام e.g., word processing program, Web browser, Adobe, Photoshop

**1-operating system**

: is an interface between hardware and applications

it is responsible for the management and coordination of activities and the sharing of the limited resources of the computer

مسؤولة عن إدارة وتنسيق الأنشطة وتقاسم الموارد المحدودة للكمبيوتر

**2- OS CONT** Is a collection of programs that controls how the CPU, memory, and I/O devices work together

manages the execution of all application programs controlling how data and instructions are loaded into memory and accessed by the CPU

عبارة عن مجموعة من البرامج التي تتحكم في كيفية عمل وحدة المعالجة المركزية والذاكرة وأجهزة الإدخال / الإخراج معا

operating system provides an interface for the user to interact with the computer (GUI)

يوفر نظام التشغيل واجهة للمستخدم للتفاعل مع الكمبيوتر

**numbring systems** Diyaa Suboh

| Decimal System                                  | Binary System                                  | Octal System    | Hexadecimal System                  | Binary System  |
|---|--|-----------------|-------------------------------------|----------------|
| Most People Use decimal representation to count | Computer is electronic machine with two states | Using 8 digits  | Uses 16 digits                      | uses 7 digits  |
| The base is 10                                  | the base is 2                                  | The base is 8   | The base is 16                      | the base is 7  |
| 0,1,2,3,4,5,6,7,8,9                             | 0 1  | 0,1,2,3,4,5,6,7 | 0, 1,2,3,4,5,6,7,8,9 ,A,B,C,D,E,F,G | 0,1,2,3,4,5,6, |

word = 2 byte = 4 nibble = 16 bits = 16 digits / Nibble = 4 bits = 4 digits = 1/2 byte / byte = 8 digits = 8 bit

Bit = digits (مطلوبة حفظ) 2 = 1+1 منطقيا وثنائي 10

**Binary Addition**

1 1101 + 0110 = 10011

01111 + 00110 =  
 0 1 1 1 1  
 + 0 0 1 1 0  
 -----  
 1 0 1 0 1

11010011 + 01010110 =

1 1 1 1  
 1 1 0 1 0 0 1 1 → (211)<sub>10</sub>  
 + 0 1 0 1 0 1 1 0 → (86)<sub>10</sub>  
 -----  
 1 0 0 1 0 1 0 0 1 = (297)<sub>10</sub>

101001<sub>2</sub> ... ( )<sub>10</sub>  
 منحت هون محل الصفر القيمة صفر ومحل الواحد القيمة ٢ للتكرارها ٤٨٢ الخ  
 1 + 0 + 0 + 8 + 0 + 32 = 41

(55)<sub>10</sub> = ( )<sub>2</sub>  
 منقسم العدد ع ٢ في حال كان زوجي يكون الباقي ٠ اما فردي يكون ١  
 110111 =

(66)<sub>8</sub> = ( )<sub>10</sub>  
 6 \* 8 = 48  
 6 \* 1 = 6  
 48 + 6 = 54

(892)<sub>8</sub> = ( )<sub>2</sub>  
 منقولها مباشره الى ثنائي يعني ٨ الى ثنائي (كل رقم لحال)  
 (1000 1001 0010)

10010101<sub>2</sub> = ( )<sub>h</sub>  
 1001 0101 = (95)<sub>h</sub>

11100011<sub>2</sub> = (E3)<sub>h</sub> H.W

Let's convert the value (39)<sub>10</sub> to Hexadecimal

| Input | Result | Remainder |
|-------|--------|-----------|
| 39/16 | 2      | 7         |
| 2/16  | 0      | 2         |

(39)<sub>10</sub> = (27)<sub>h</sub>

Convert the following numbers to decimal

a. (72)<sub>8</sub> = (58)<sub>10</sub>  
 b. (72)<sub>16</sub> = (114)<sub>10</sub>  
 c. (DE)<sub>16</sub> = (3553)<sub>10</sub>

ملاحظة: Signed Numbers هو عبارة عن قلب الصفر واحد والواحد صفر واذا كان العدد الموجب منكتب Positive واذا كان سالب Negative مثلا 5 = 101 وهي عبارة عن 0101 ونقلب اول رقم تصبح سالب 1101 مثال ٢: الرقم 100101 هو عبارة عن 011010 بالسالب

**Converting Fractions**

| Convert 11.375 <sub>10</sub> to it's binary | (0.2) <sub>10</sub> |
|---|---------------------|
| 11 = 1011                                   | 0.2 * 2 = 0.4       |
| 0.375 * 2 = 0.750                           | 0.4 * 2 = 0.8       |
| 11.375 = 1011.011                           | 0.8 * 2 = 1.6       |
|   | 0.6 * 2 = 1.2       |

(0.2)<sub>10</sub> = (0.0011)<sub>2</sub>

ملاحظة: جمع الكسور الثنائية نفس طريقة الجمع العادية للثنائي

**Binary Subtractor**

Example: 00110010<sub>2</sub> + (-125)<sub>10</sub>

125 → 01111101

1's complement → 10000010  
 2's complement → + 1  
 10000011 → (-125)

00110010<sub>2</sub> + (-125)<sub>10</sub>

00110010 + 10000011 = 10110101

50  
 - 125  
 -----  
 -75

**Data Representation**

تمثيل البيانات هون راح نمثل البيانات ب 0, 1

ملاحظة: even زوجي، odd فردي

Ex. Ahmad

| Parity bit | ASCII                      | Parity bit | ASCII                      |
|------------|----------------------------|------------|----------------------------|
| A 1        | 1001001 = 71 <sub>16</sub> | h 0        | 1101000 = 68 <sub>16</sub> |
| x 0        | 1101101 = 6D <sub>16</sub> | a 0        | 1100001 = 61 <sub>16</sub> |
| d 0        | 1101100 = 64 <sub>16</sub> |            |                            |

مثلا اكتب اسم احمد عبر ASCII بشرط ان يكون odd فردي

| Memory |
|--------|
| C1     |
| 68     |
| 6D     |
| 61     |
| 64     |

ملاحظات Exponent هي دائما قيمتها 127 بالاضافة الى عدد الارقام بعد الرقم الاول

امثلة على تمثيل البيانات .....

1. (-203):

| Input | Result | Remainder |
|-------|--------|-----------|
| 203/2 | 101    | 1         |
| 101/2 | 050    | 1         |
| 050/2 | 025    | 0         |
| 025/2 | 012    | 1         |
| 012/2 | 006    | 0         |
| 006/2 | 003    | 0         |
| 003/2 | 001    | 1         |
| 001/2 | 000    | 1         |

**= (11001011)<sub>2</sub>**

To be 16 bits: (0000000011001011)<sub>2</sub>

Two's complement: 1111111100110100 + 1

From the table: **F F 3 5**

35  
FF

Low-High

Exp : Using the **even parity** bit to represent the character **Q (Q = 81 in ASCII)** in memory (Hexadecimal) ?

استخدام بت التكافؤ الزوجي لتمثيل الحرف Q (Q = 81 في الذاكرة) سداسي عشري؟

الرقم 81 نعمل على تمثيله بالثنائي  $D1_{16} = (81)_{10} = (01010001)_2$

نضع رقم الاخير وهو 0 كرقم اضافي ونعمل على اضافة صفر او واحد لنجعل الرقم الثنائي رقم زوجي كذلك ف يجب اضافة الرقم 1 فيصبح لدينا اربعة ارقام تمثل بالثنائي تمثل واحد زوجي ثم نعمل على تحويله الى سادس عشر باخذ كل اربع ارقام مجموعة. فيمثل الرقم  $D1_{16}$

|             |         |
|-------------|---------|
| Parity bit  | 1       |
| Q           | 1010001 |
| = $D1_{16}$ |         |

**نحول** Convert the number from decimal to binary **نعمل على تحويل من العشري الى الثنائي**

الى ثنائي كما ذكر سابقا  $(62.57)_{10} = (01011.11)_2$

نعمل على تحريك الاشارة بمقدار 4 ثم تضرب ب  $2^4$   $(01011.11)_2 = (1.110101 * 2^4)_2$

Exponent =  $127+4=131$   $(131)_{10} = (10000011)_2$

|   |          |                      |
|---|----------|----------------------|
| 0 | 10000011 | 10101100000000000000 |
| 4 | 1        | D 6 0 0 0 0          |

|    |
|----|
| 00 |
| 00 |
| D6 |
| 41 |

### algorithm

An **algorithm** is a **procedure** or **formula** for solving a problem. الخوارزمية هي إجراء أو صيغة لحل مشكلة.

- **Pseudocode** is a kind of structured English for describing algorithms. It allows the designer to focus on the logic of the algorithm without being distracted by details of language syntax

• **Pseudocode** هو نوع من الإنجليزية المنظمة لوصف الخوارزميات. انها تسمح للمصمم التركيز على منطق الخوارزمية دون ان يكون مشتتاً بتفاصيل تركيب اللغة.

كيفية انشاء الخوارزميات --

|   |                                 |  |
|---|---------------------------------|--|
| <b>in put : Read ...optain .. get</b>     | read num 1 and save as n        | الجمع : ADD                                      |
| <b>out put : print ...display .. show</b> | print("sum")                    | قسمة : div                                       |
| <b>compute : compute , calculate</b>      | calculate avg ... calculate sum | ضرب : multiPly                                   |
| <b>initilize : set</b>                    | set num 1 equal zero            | ناقص : subscriber                                |
| Add one: INCREMENT                        | increment counter               |  |
| Greater than > ... Greater than           | Or equal >=... less than <      | less than Or equal <= .. equal = ... <>Not equal |

رموز و الاشارات ( مطلوب الاشارات كتابة بالخوارزميات )

### انواع الخوارزميات

write an alogaithem to find and print the sum of two integers

**تسلسلي sequitial**

**start**

ask user to enter the first intege

read the first integer and save as F\_1

ask user to enter the second integer

read the second integer and save S\_1

add F\_1 to S\_1 and save result as sum

print ("sum") to screen

**end**

Write an algorithm to find and print the area of rectangle

مساحة المستطيل = الطول \* العرض

rectangle area = heigth \* width

**start**

ask user to enter the heighth of rectangle

Read heighth and save as H

ask user to enter the width of rectangle

read width and save as W

multiPIY H by W and save the ruselt as area

Display area

**end**

write an alogaithem to find and print the area of square

square = side \* side

**Start**

ask user to enter the side

Read side and save as S

Multiply s by s and save result as E

display ("E")

**End**

### مشروط Conditional

شرط (if ... case)

اذا كان لدينا شرط واحد

**if condition then**

Sequence

**End if**

اذا كان لدينا اكثر من شرط نكتب

**else if** الشرط نستخدم

**if condition then**

Sequence 1

**else if condition then**

Sequence 2

**else if condition then**

Sequence 3

**else if condition then**

Sequence 4

**else**

Sequence 5

**End if**

1 exp

Write an algorithm to print passed OR failed based on the students Grade

**Start**

ASk user to enter student grade

Read grade and save as stg

**if** stG greater than or equal I sixty then

print ("pass")

**else**

print ("failed")

**end if**

**End**

2 Write an algorithm to find and print the miximum element of a set of 4 integers

**Start**

Ask user to enter the first integer

Read the first integer and save as F

Ask user to enter the Second integer

Read the Second integer and save as S

Ask user to enter the thirty integer

Read the thirty integer and save as T

Ask user to enter the fourty integer

Read the fourty integer and save as FF

let max equal to the F

**if** max less than S then

set max to s

**else if** max less than T then

set max to T

**else if** max less than FF then

set max to FF

**End if**

print "the max integer is" max

**End**

# Diyaa Suboh

Write an algorithm to find and print the smallest of four given numbers (assume all) numbers are different

```

Start
Ask user to enter first number
Read number and save as num1
Ask user to enter second number
Read number and save as num2
Ask user to enter third number
Read number and save as num3
Ask user to enter fourth number
Read number and save as num4
If num1 smaller than num2 and num1 smaller than num3 and num 1 smaller than num 4 then
    print "num1 is the smallest"
else If num2 smaller than num1 and num2 smaller than num3 and num 2 smaller than num 4 then
    print "num2 is the smallest "
else If num3 smaller than num1 and num3 smaller than num2 and num 3 smaller than num 4 then
    print "num3 is the smallest"
else
    print "num4 is the smallest"
end if
End
    
```

```

Write an algorithm to read a
number x and display its sign
Start
Ask user to enter a number
Read number and save as X
If x is greater than zero then
    print x "is positive"
else if x is equal zero then
    print x "is zero"
else
    print x "is negative"
end if
    
```

```

Write an algorithm that will input student average. If the average is greater than or equal to 60 and
less than or equal to 70, the algorithm should display "Passed". If it is greater than 70 and less than or
equal to 80, print "Good". If it is greater than 80 and less than 90, print "Very good". If it is greater
than 90 , print "Excellent". If it is less than 60 the prints "Fail"
Start
Ask user to enter student average
Read average and save as ag
If ag is greater than or equal to sixty and ag is less than or equal to seventy then
    print "Pass"
else if ag is greater than seventy and ag is less than or equal to eighty then
    print "Good"
else if ag is greater than eighty and ag is less than ninety then
    print "Very good"
else if ag is greater than ninety then
    print "Excellent"
else
    print "Fail"
end if
End
    
```

## في ثلاث صيغات هوون

| iterative   | WHILE condition<br>sequence<br>End while | REPEAT<br>sequence<br>End REPEAT | FOR iteration bounds<br>sequence<br>End FOR  |
|---|--|----------------------------------|--|
| <p>Write an algorithm if the first number of the value x is 2 where x is greater than or equal to 6</p> <pre> Start Set i equal to two While i less than or equal six     print i     add two to i end while End                 </pre> |  |                                  | <p>Write an algorithm to calculate the average of a set of 10 students</p> <pre> Start Set counter to zero Set total to zero While counter is less than ten     Ask user to enter grade     Read grade and save as gd     Add the gd into the total     increment counter end while Set the average to the total divided by counter Print "the average is " average End                 </pre> |

Write an algorithm that will count the number of student pass in a class and the amount failed. The pass mark is more than or equal to 65. Suppose the number of students are 30 . The algorithm should output the amount fail and

```

Start
Set counter to zero
Set passCounter to zero
Set failureCounter to zero
While counter less than thirty
    Ask user to enter student average
    Read average and save as ag
    if ag greater than or equal sixty five then
        increment passCounter
    else
        increment failureCounter
    end if
    increment counter
end while
Print "pass counter =" passCounter" and failure counter ="failureCounter
End
    
```

Diyaa Suboh