

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

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

1. Directs the computer system to execute stored program instructions تنفيذ تعليمات البرنامج المخزن

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

e.g Addition, Subtraction , Multiplication, Division

2. logic operations:

(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 جزء من الكمبيوتر يخزن البيانات و المعلومات

- (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

إلخ ، off and on ، open and closed ، volts and no volts : مثل مكتوب البت رقمين 1 :: 0 ويتم تمثيل البيانات بسهولة closed, volts and no volts.

.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

memory
 primary storage RAM , ROM
 secondary storage Flash memory ,DVD,BD,floppy disk , fard disk ,magnetic taces

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 يخزن البرنامج المطلوب لبدء تشغيل الكمبيوتر	Allows the computer to read data quickly to run applications يسمح للكمبيوتر بقراءة البيانات بسرعة لتشغيل التطبيقات

in put device : Keyboard mouse scannar mic sensors	
out put device : screen prenter speakers camara	

Diyaa Suboh

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

To be 16 bits: (0000000011001011)₂

Tow's complement: 1111111100110100 + 1

From the table: 1111 1111 0011 0101

FF 35

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 ASCII في الذاكرة) سداسي عشري؟

الرقم 81 نعمل على تمثيله بالثنائي

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 D1₁₆

Parity bit	1	1010001	= D1 ₁₆
------------	---	---------	--------------------

نعمل على تحويل من العشري الى الثنائي

Convert the number from decimal to binary

الى ثنائي كما ذكر سابقا (62.57)₁₀ = (01011.11)₂

نعمل على تحريك الاشارة بمقدار 4 ثم تضرب ب 2⁴ (01011.11)₂ = (1.110101 * 2⁴)₂

Exponent = 127+4=131 (131)₁₀ = (10000011)₂

0	10000011	101011000000000000000000
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 أن يكون مشتتاً بتفاصيل تركيب اللغة.

كيفية انشاء الخوارزميات --

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

رموز و الاشارات (مطلوب الاشارات كتابة بالخوارزميات)

انواع الخوارميات

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 heigth of rectangle

Read heigth and save as H

ask user to enter the width of rectangle

read width and save as W

multiPLY 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

(if ... case) شرط

إذا كان لدينا شرط واحد

if condition then

Sequence

End if

إذا كان لدينا أكثر من شرط نكتب

else 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

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 fourty 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

itertive	في ثلاث صيغات هوون		
	WHILE condition sequence End while	REPEAT sequence End REPEAT	FOR iteration bounds sequence 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> <p>Start</p> <p>Set i equal to two</p> <p>While i less than or equal six</p> <p>print I</p> <p>add two to I</p> <p>end while</p> <p>End</p>	<p>Write an algorithm to calculate the average of a set of 10 students</p> <p>Start</p> <p>Set counter to zero</p> <p>Set total to zero</p> <p>While counter is less than ten</p> <p>Ask user to enter grade</p> <p>Read grade and save as gd</p> <p>Add the gd into the total</p> <p>increment counter</p> <p>end while</p> <p>Set the average to the total divided by counter</p> <p>Print “the average is ” average</p> <p>End</p>	

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