Computer 142

Alaa Itaimi

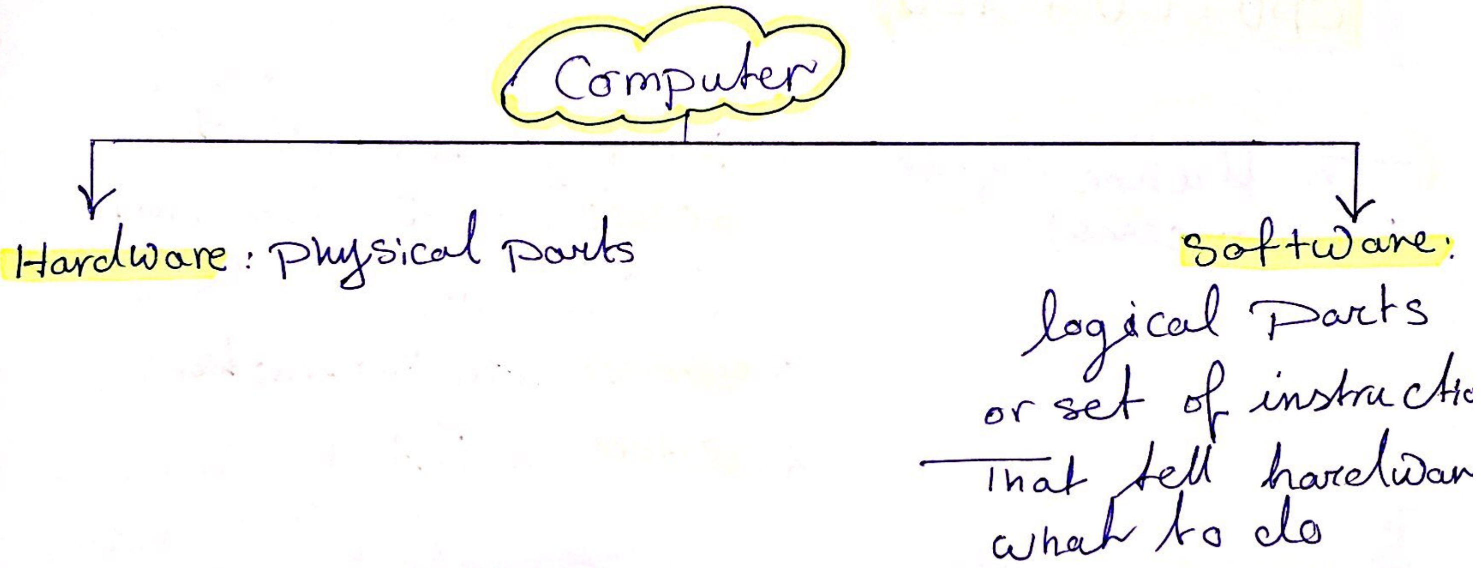
lecture 1 :-

Computer :- A device that takes data as input- Process it and produces information as output

Difference between DATA and INFORMATION :-

Data: facts : البيانات التي تدخل وهي ليست مهمة بمعنى انه لا ~~يحكم~~ يتخذ عليها قرار

Information :- يتخذ قرار على أساسها



Computer

Hardware : physical parts

Software : logical parts or set of instructions That tell hardware what to do

As a summary : A hardware consists of :-

1) CPU : Central processing unit (Brain of computer)
- ALU : Arithmmetic / logic units
- CU : Control Unit
- Register :- وحدة تخزين داخل CPU

2) storage < primary / secondary

3) Input / output

UPLOADED BY AHMAD JUNDI

Scanned by CamScanner

To explain it more :- ① CPU

a) ALU

Arithmetic : الحاسوب يستطيع القيام بعمليات جمع فقط

جمع = طرح = ضرب = قسمة

logic :- > or < or = : Basic logic

b) CU : controls everything

c) Register :- you put things that you want to reach it fast But it's not that big (it fits important things)

CPU = CU + ALU

The Machine cycle :- (4 processes)

1- fetch (get) next instruction
2- decode : ترجمتها : find what instruction is (changes instruction to commands)
3- Execute : run instructions
4- Store in RAM

② Storage :
- → Primary :- RAM
- → secondary : Disks, flash, Cd, dvd, ...

③ Input & output

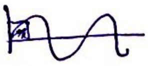
Input	output
Keyboard scanner Michrophone	Printer Speakers screen Plotter ← الخرائط (طابعة كبيرة)

Difference Between RAM & ROM :-

RAM	ROM
① Read & write	Read only
② Volatite متطاير = Temporary	Not-volatite غير متطاير

ecture 2

• Data Representation , Data can be :-

- numbers → integers, → floats
- Charachters → ASCII A → 65 a → 97
- Pictures
- videos
- sound → 
- stings "ali"

~~Bin~~ • To represent data , we change it to numbers in the Decimal system and those numbers we represent them ~~as~~ in the Binary system

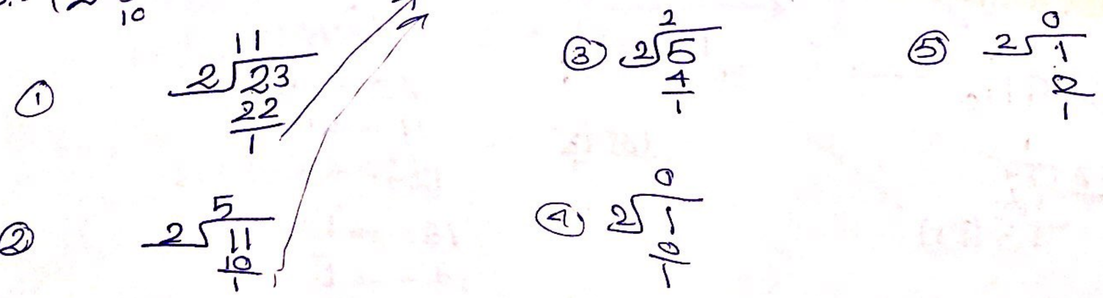
1: on 0:off

Numbering systems

- binary :- (الثنائي) 2 (0-1)
- Decimal :- (العشري) 10 (0-9)
- octal :(الثماني) 8 (0-7)
- Hexadecimal (السادس عشري) 16 (0-15) (0,-9, A, B, C, D, E, F)

How to change from decimal to binary :-

Exp:- $(23)_{10} \longrightarrow (\overleftarrow{10111})_2$



How to chang from binary to decimal

$(10111) \rightarrow (\ 23\ )_{10}$

$\times\ \times\ \times\ \times\ \times$

$2^4\ 2^3\ 2^2\ 2^1\ 2^0$

$16 + 0 + 4 + 2 + 1$

$= 23$

How to change from decimal to octal

$(23)_{10} \rightarrow (\ \overleftarrow{27}\ )_8$

① $$\begin{array}{r} 2 \\ 8\overline{)23} \\ \underline{16} \\ 7 \end{array}$$

② $$\begin{array}{r} 0 \\ 8\overline{)2} \\ \underline{0} \\ 2 \end{array}$$

How to Change from octal to decimal

$(27)_8 \rightarrow (\ 23\ )_{10}$

$\times\ \times$

$8^1\ 8^0$

$16\ \cancel{8} + 7$

- We change between any systems in the same way

As an example :- $(23)_{10} \rightarrow (\overleftarrow{113})_4$

$$\begin{array}{r} 5 \\ 4\overline{)23} \\ \underline{20} \\ 3 \end{array} \rightarrow \begin{array}{r} 1 \\ 4\overline{)5} \\ \underline{4} \\ 1 \end{array} \rightarrow \begin{array}{r} 0 \\ 4\overline{)1} \\ \underline{0} \\ 1 \end{array}$$

$(113)_4 \rightarrow (\ 23\ )_{10}$

$\times\ \times$

$4^2\ 4^1\ 4^0$

$16 + 4 + 3$

How to change from Hexadecimal to decimal

$(29)_{10} \rightarrow (\ \overleftarrow{1\ D}\ )_{16}$

$$\begin{array}{r} 1 \\ 16\overline{)29} \\ \underline{16} \\ 13\ (D) \end{array}$$

$$\begin{array}{r} 0 \\ 16\overline{)1} \\ \underline{0} \\ 1 \end{array}$$

Note :-

10 → A

11 → B

12 → C

13 → D

14 → E

15 → F

$(1D)_{16} \rightarrow (29)_{10}$

$\overset{\times}{16^1}\ \overset{\times}{16^0}$

$16 + 13 \rightarrow 29$

Examples :- Between 2 systems that we don't Know

$(12)_3 \xrightarrow{\longleftarrow} (1\ 0)_5$

$3^1\ 3^0$

$3+2$

$(5)_{10}$

$5\overline{)5}$ quotient 1, $\underline{5}$, 0 $\rightarrow$ $5\overline{)1}$ quotient 0, $\underline{0}$, 1

- Examples

① $(AB37CDF245) \longrightarrow ($ ⑯ $\rightarrow 2^4$ ⑧ $\rightarrow 2^3$

$\underbrace{001}_{1}\underbrace{010}_{2}\underbrace{101}_{5}\underbrace{100}_{4}\underbrace{110}_{6}\underbrace{111}_{7}\underbrace{110}_{6}\underbrace{011}_{3}\underbrace{101}_{3}\underbrace{111}_{7}\underbrace{001}_{1}\underbrace{001}_{1}\underbrace{000}_{0}\underbrace{101}_{5}$

$(1\ 2\ 5\ 4\ 6\ 7\ 6\ 3\ 3\ 7\ 1\ 1\ 0\ 5)_8$

→ This works only when the system can be divided on 2 like 8, 16, 32 ---

Note:-

$2^3\ 2^2\ 2^1\ 2^0$

0000 → 0
0001 → 1
0010 → 2
0011 → 3
0100 → 4
0101 → 5
0110 → 6
0111 → 7
1000 → 8
1001 → 9
1010 → 10
1011
1100
1101
1110
1111 → 15

② (37245) ⑧ $\rightarrow 2^3 \longrightarrow (3\ EA5)_{16}$ 2^4

$\searrow$ $(\underbrace{011}\underbrace{111}\underbrace{010}\underbrace{100}\underbrace{101})_2$ $\nearrow$

$(2.5)_{10} \rightarrow (10.1)_2$

$0.5 \times 2 = 1.0$

$(2.25) \rightarrow (10.01)_2$

$0.25 \times 2 = 0.5$

$0.5 \times 2 = 1.0$

لو كان الجواب
مثلاً 1.6
نأخذ 0.6 فقط

lecture 3

Ex:- $(52\cancel{*})_8 \xrightarrow{2^3} (\overset{\times}{\underset{4^2}{2}}\;\overset{\times}{\underset{4^1}{2}}\;\overset{\times}{\underset{4^0}{2}})_4 \xrightarrow{2^2} (\overleftarrow{4\;\;6})_9$

$\hookrightarrow (\underbrace{10}\,\underbrace{10}\,\underbrace{10})_2 \nearrow \qquad \hookrightarrow (\overset{2+8+32}{42})_{10} \nearrow$

$\hookrightarrow 9 \overline{)\,42}^{\,4}$, 36, 6 → $9 \overline{)\,4}^{\,0}$, 0, 4

- To make sure that the answer is right we change The octal to decimal, then we change the $(\;)_4$ To decimal & if the answer is 42 for all Then My answer is right!

Examples:-

① $(13.125)_{10} \rightarrow (\overleftarrow{\underset{2^3}{1}\,\underset{2^2}{1}\,\underset{2^1}{0}\,\underset{2^0}{1}}\,.\,\overrightarrow{\underset{2^{-1}}{0}\,\underset{2^{-2}}{0}\,\underset{2^{-3}}{1}})_2 \rightarrow (13.125)_{10}$

(مثل خط الأعداد.)

$\underbrace{8+4+0+1}_{13} \,.\, \underbrace{(0+0+\frac{1}{8})}_{0.125}$

$0.125 \times 2 = \boxed{0}.25$

$0.25 \times 2 = \boxed{0}.5$

$0.5 \times 2 = \boxed{1}.0$

To change from 2 to 8:-

$(\overleftarrow{\underbrace{001}\,\underbrace{101}}\,.\,\overrightarrow{\underbrace{001}})_2$

$\hookrightarrow (15 \,.\, 1)_8$

from 2 to 16

$(\overleftarrow{\underbrace{1101}}\,.\,\overrightarrow{\underbrace{0010}})_2$

$(\underset{\times 16^0}{D} \,.\, \underset{\times 16^{-1}}{2})$

$\hookrightarrow = (13.125)_{10}$

- **Binary operations**

$$\left(\begin{array}{r} 101 \\ +\ 10 \\ \hline 111 \end{array}\right)_2 \quad \left(\begin{array}{r} 5 \\ +\ 2 \\ \hline 7 \end{array}\right)_{10}$$

- **Rules:-**

• يجمع رقمين فقط

$$\begin{array}{r} 0 \\ +\ 0 \\ \hline 0 \end{array}, \quad \begin{array}{r} 0 \\ +\ 1 \\ \hline 1 \end{array}, \quad \begin{array}{r} 1 \\ +\ 0 \\ \hline 1 \end{array}, \quad \begin{array}{r} 1 \\ +\ 1 \\ \hline 10 \end{array}, \quad \begin{array}{r} 1 \\ 1 \\ +\ 1 \\ \hline 11 \end{array}$$

Examples:-

① $$\begin{array}{r} 1011 \\ +\ 10000 \\ \hline 11011 \end{array}$$

② $$\left(\begin{array}{r} 1\ 1 \\ 11011 \\ +\ 11100 \\ \hline 100111 \end{array}\right)_2 \quad \left(\begin{array}{r} 11 \\ +\ 28 \\ \hline 39 \end{array}\right)_{10}$$

③ $$\begin{array}{r} 1\ 1 \\ 111 \\ +\ 111 \\ \hline 1110 \end{array}$$

④ $$\left(\begin{array}{r} 1 \\ 11.01 \\ +\ 101.10 \\ \hline 1000.11 \end{array}\right)_2 \quad \left(\begin{array}{r} 3.25 \\ +\ 5.5 \\ \hline 8.75 \end{array}\right)_{10}$$

⑤ $$\left(\begin{array}{r} 1\ 1 \\ 11.11 \\ +\ 10.10 \\ \hline \underbrace{110}_{6}.\underbrace{01}_{0.25} \end{array}\right)_2 \left(\begin{array}{r} 3.75 \\ 2.5 \\ \hline 6.25 \end{array}\right)_{10}$$

Note:-

اذا كان الرقم يبدأ بـ 0 هو موجب: $0101 \to +$

اذا كان الرقم يبدأ ا هو سالب: $1101 \to -$

Two's complement

[ne]gative ← $\underline{1}01 \rightarrow \cancel{5}$ It's −3

[p]ositive ← $\underline{0}101 \rightarrow +5$ ✓

- **1's complement** : 101 → 010
  - we put 0 instead of 1 and we put 1 instead of 0

- **2's complement**
  - $$\begin{array}{r} 010 \\ +\ 1 \\ \hline 011 \end{array} \rightarrow \boxed{+3}$$
  - we add 1 to the number
  - ↳ so 101 is −3

Ex: 0110 → +6

$$\begin{array}{r} 1001 \\ +\ 1 \\ \hline 1010 \end{array} \rightarrow -6$$

$$\begin{array}{r} 0101 \\ +\ 1 \\ \hline 0110 \end{array} \rightarrow +6$$

5 − 2

= 5 + (−2)

0101   1110

$$\begin{array}{r} 0101 \\ +\ 1110 \\ \hline \cancel{1}0011 \end{array}$$

- 2 is 0010 ↳ +2
- to get $\underline{-2}$ we use 2's complement :-

$$\begin{array}{r} \cancel{0010} \\ 1101 \\ +\ 1 \\ \hline 1110 \end{array}$$

- لا نأخذ الـ1 على ~~الشمال~~ اليسار لأنه العملية ↲ كانت بين رقمين أكبرهما تتكون من اربع خانات وبالتالي الجواب يتكون من اربع خانات ونستثني الخانات من اليسار

Ex: (-7+1
(1001) (0001)

+7 من 0111
1000
1 +
1001

• -7 +
1
-6

~~7~~ من

1
1001
0001+
1010 ← -6

Ex: ~~6 3~~
-5 - 1
1011 1111

• +5 من 0101
1010 +
1
-5 ← 1011

1 11
1011
1111 +
~~1~~1010 ← -6

• +1 من 0001
1110
1 +
-1 ← 1111

Ex: -5 - 4
1011 1100

• +4 من 0100
1011 +
1011 +
1
1100 ← -4

-5 +
-4
-9

1011
1100 +
~~1~~0111 ← +7

• الجواب خاطئ لان الارقام التي دخلت في العملية تحتاج الى اربع خانات لتمثيلها ولكن الجواب (-9) وهو الصحيح يحتاج الى خمسة خانات لتمثيله

• لذلك الحاسوب يستعمل عدد خانات كبير لكي يتأكد من الجواب فيكون عدد الخانات ~~64~~ 64 و 128 او عدد اكبر من الخانات

• عندما يكون الجواب غريب او خاطئ نزيد عدد الخانات للتأكد

ecture 4:-

mple:- Using two's complement with 8-bits Solve :-

① $(75)_{16} - (34)_7 = (\cancel{1}01011100)_2 = (1130)_4$

$(01110101)_2$

$(25)_{10}$

$(00011001)_2$

$$\begin{array}{r} 11100110 \\ 1 \\ \hline (11100111)_2 \end{array} +$$

$(01110101)_2 + (11100111)_2 =$

$$\begin{array}{r} 01110101 \\ 11100111 \\ \hline \boxed{(\cancel{1}01011100)_2} \end{array} +$$

② $(20)_{10} - (15)_{10} =$

8 4 2 1

$(00010100)_2 - (00001111)_2 =$

$$\begin{array}{r} 11110000 \\ 1 \\ \hline \end{array} +$$

$00010100_2 \cdot (11110001)_2$

$$\begin{array}{r} 00010100 \\ 11110001 \\ \hline \cancel{1}00000101 \end{array} +$$

UPLOADED BY AHMAD JUNDI

Data Presenhation :-

1. integer : $\underbrace{\llcorner\,|\overset{15\,Bits}{\longleftrightarrow}|}_{16\ Bit}$ 16 Bit $\Rightarrow 2^{16} = 65536$
2. float : This bit is for signs

   -32768 | 0 | 32767

   $-2^{15} \downarrow +2^{15}-1$
3. Char :

- 2'complement is used with integers

if I had one bit ⊔ I can put either 1 or zero so $2^1 = 2$

if I had 2 bits ⊔⊔ I can put 4 diff things $2^2 = 4$

$(37)_{10} = (100101)_2$

high order → $\underbrace{0000\,0000\,0010\,0101}_{0\quad 0\quad 2\quad 5}$ → low order

- How is it saved in the Memory?

memory 8 bit
25
00

→ small Endian

- There is 2 ways :- either The Big Endian or The small Endian

$(-17)_{10}$

$+17$ $(010001)_2$

$$\begin{array}{r} (0000000000010001)_2 \\ 1111111111101110 \\ 1\ + \\ \hline \underbrace{1111}_{F}\underbrace{1111}_{F}\underbrace{1110}_{E}\underbrace{1111}_{F} \end{array}$$

Memory
EF
FF

ASCII → Extended ASCII → Unicode Universal Code

③ Char

a - 3 , A - Z , 0 - 9 , ? , #, - - - -

- ASCII (128)

↳ American standard Code for information interchange

A - 65

a - 97

d - 100

d → using even or odd parity

d → memory

$\frac{\text{Mem}}{6\ 4}$

↳ $(100)_{10} \rightarrow (\underbrace{\_\,11}_{6}\underbrace{00100}_{4})_2$

parity bit (error detection)

odd even

if using odd: $\underline{0}$ 1100100

There is Three 1 (it's an odd 1)
So We put 0 in The parity bit

if using even $\underline{1}$ 1100100

There is Three 1 (it's an odd 1)
So We put 1 and it becomes 4 which is a parity bit

As an example ahmad :- a →
h →
m →
a →
d →

a	97
b	98
c	99
d	100
e	101
f	102
g	103
h	101
i	105

5247.32

The Scientific No

is 5.24732×10^3

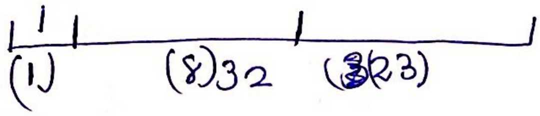
② float :-

$(-4.25)_{10}$

$\hookrightarrow (100.01)_2$

- scientific notation

$= 1.0001 \times 2^2$



ure5

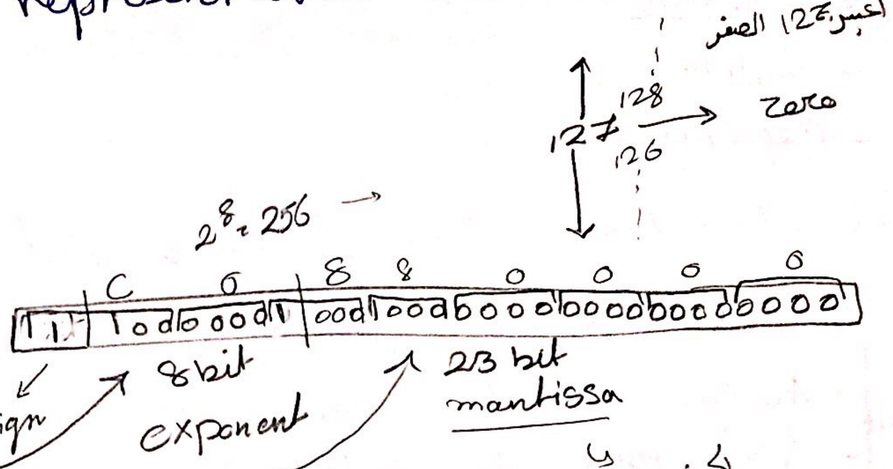
Floating point representation

$(-4.25)_{10}$

① 100.01

② ①.0001 × $2^{②}$

لا نخزن



الجزء الكسري من الرقم

⟹ 127 + 2 = 129

Memory
0 0
0 0
8 8
C 0

(0.2) × 2 = 0.4

(0.4) × 2 = 0.8

(0.8) × 2 = 1.6

(0.6) × 2 = 1.2

0.2 × 2 = 0.4

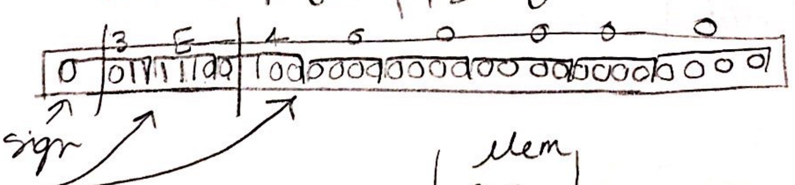
127 + (-3) = 124

$2^1 + 2^0 = 3$ نحط 127 أو ننقص منه

① 0.00110

② 1.10×2^{-3}

$= 1.① \times 2^{③}$



Mem
0 0
0 0
4 0
3 E

• Algorithms : طرق الحل

- sequence
- conditional
- Repetition

• in Algorithms There is an input , an output & processi

find The Algorith to sum any Two numbers

input
- Ask user to enter first number
- Read number and save as num 1
- Ask user to enter second number
- Read number and save as num2

processing
- Add num1 to num 2 and save result as sum

output
- Print sum to screen

• دائماً البدأ بفعل أمر
• الجملة يجب ان تكون سطر واحد (١٠ كلمات كحد أعلى)
• يمكن استعمال أي نوع من الكلمات بشرط ان يكون المعنى كامل وصحيح
• الترتيب مهم أحياناً

In The <u>Top-down design</u> :-

1- get number → 1.1 get first number → 1.1.1 Ask user
→ 1.1.2 ~~Ask~~ Read --
→ 1.2- get second number

2- first sum

3- Print sum

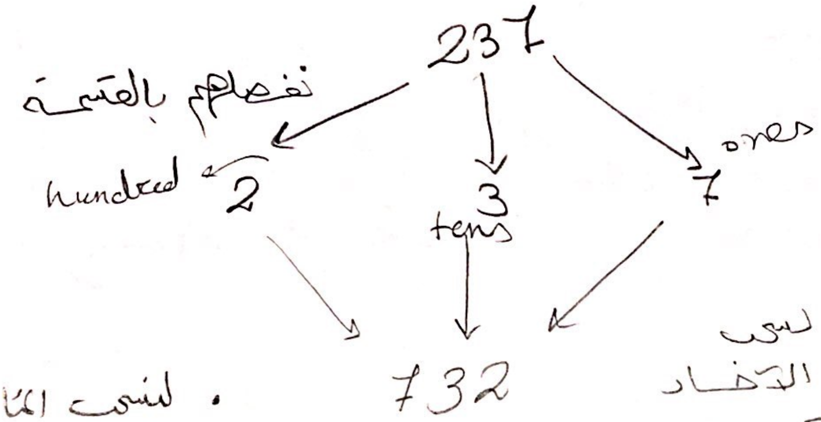
lecture 6

Write an algorithm (pseudo-code) :-

- to reverse any given three digit number

237 → 732

102 → 201



$5 \% 2 = 1$

ما هو الباقي

$2 \% 8 = 2$

But

$5 / 2 = 2$

- لنجيب الآحـاد
نقسم على 10

$10 \overline{)237}$ = 23, 230, $\boxed{7}$

- لنجيب المئات
نقسم على 100

$100 \overline{)237}$ = $\boxed{2}$, 200, 37

- لنجيب العشرات هناك اكثر من طريقة :
نأخذ آخر خانتين 37 ونقسم على 10 ونأخذ الجواب

$10 \overline{)37}$ = 3

أو نأخذ أول خانتين ونقسم على 10 ونأخذ الباقي

- The Algorithm:-

```
1- Ask user to Enter any three digit number  ] inpu
2- Read number and save as num               ]
3- Divide num by a hundred and save result as
      hunds
4- Divide num by ten and save remainder as
      ones
5- Divide num by a hunds and save remainder as
      temp
6- Direct temp by ten and save result as tens
```

7- Multiply ones by a hundred and save resu
rev

8- Multiply tens by ten and add result to rev

rev + = ten * 10
rev = rev + (tens * 10)

9- Add hunds to rev

10- Print rev to screen

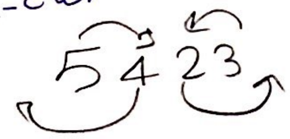
We Try:- 352

num = 352
hunds = 352/100 = 3
ones = 352 %10 = 2
temp = 352 % 100 = 52
tens = 52 % 10 = 5

rev = 2 * 100 = 200
rev = 200 + (5 * 10)
= 250
rev = 250 + 3
= 253

253

Example:- In-cut



Ex:
5123 %10 = 3
5123/10 = 512

512% 10 = 2
512/ 10 = 51

51 % 10 = 1
51 / 10 = 5

5 %10 = 5
5/10 = 0

x = num%10 ;
num = num/10 ;

X = num % 10;
num = num /10;

X = num% 10 ;
num = num / 10 ;

We use the loop

Conditional (selection)

num → neg
num → pos

X=5 put 5 in X
X==5 does 5 equal X.

>
<
≤=

- If statement

```
If num is less than zero
    Print "num is negative" to screen
Else
    Print "num is positive" to screen
END IF
```

نتيجة الشرط ⟷ Print

Conditional (Selection) lecture 7

- Write an algorithm to decide whether a given number is odd or even

```
Ask user to enter any number
Read number and save as num
Divide num by two and save remainer as rem
  If rem equals zero
      Print ~~a~~ " num is even" to screen

  Else
      Print " num is odd" to screen

  End If
```

- Write an algorithm to change marks to letter grades such that (A= 90-100 . B = 80-90, C= 70-79 , D=60-69) F= 0.-59)

```
Ask user to enter mark
Read mark and save as mk
IF mk is greater than or equal to ninety
    Print "grade is A" to screen
    Print "good job" to screen
Else If mk is greater than or equal to eighty
    Print "grade is B" to screen
Else If mk is greater than ~~or~~ sixty nine
    Print "grade is C" to screen
Else If mk is greater than or equal sixty ~~nine~~
    Print "grade is D" to screen
Else ~~If mk is greater than~~ Print "grade is F" to
    screen
    Print " see you ~~next time~~ semester" to screen
```

```
Else
    Print "grade is F" to screen
    Print "see you next semester" to screen

End If
```

To save the grades and use them in another thing we Type after every Else If

set grade to 'A'
equal

X = A and X='A' is different

X = A → This means That A is a variable
The computer will search for a of A

X='A' → This means That (A is a Constant)

```
• If ______
      ______
      ______
  Else If ______
      ______
      ______
  End If
```

• اذا تحقق الشرط اعمل كذا
اذا لم يتحقق اعمل مثله

```
If
    ______
    ______
Else
    If ______
        ______
    End If
End If
```

• اذا لم يتحقق الشرط
يدخل على If أخرى
ويرى اذا تحقق الشرط أم
لم يتحقق ؟

Note:-
* Conditions can have : And , or , Not in it

Example::

X	y	X and y	X or y	Not x
T	F	F	T	F
F	T	F	T	T
T	T	T	T	F
F	F	F	F	T

UPLOADED BY AHMAD JUNDI

- If mk is greater than eighty nine and mk is less than or equal to a hundred

```
    Print "greater is A" to screen
  End If
```

Repition :- (loops)

There is Three Types of loops :-

- While
- do/while
- for

- Write an algorithm (pseudo-code) to find & print the avarge grade for a Class of ten students

```
  - ~~Set~~ sum equal to zero
  - Set Count equal to zero
```

لوضرب بكون one

- Two things change
  - sum
  - count

lecture 8

- Write an algorithm to calculate & print the average grade for a clan of ten istudents

```
Set sum equal to zere
set Count equal to zero → initial value
While Count is ~~ten~~ less than ten → final value (condit
   Ask user to enter grade → Read grade and save as g~~rade~~
   Add grade to sum
>) Increment Count by one → change (التغيير العد)
END while
Divide sum by ten and save result as avg
Print avg to screen
```

- How does the computer Work :-

Count $\overset{?}{=}$ 0 $\overset{?}{<}$10 yes

gr = 50

sum = 0 + 50 = 50

Count = 1   ~~yes~~ 1 $\overset{?}{<}$ 10 → yes

gr = 70

sum = 70 + 50 = 120

count = 2

|

|

count = 9

gr = 6

sum = 770 + 6 = 776

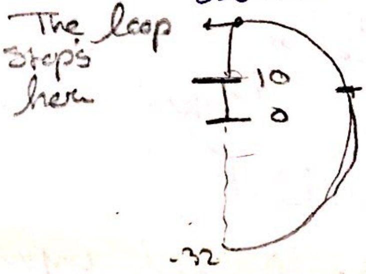
Count = 10

$$avg = \frac{776}{10}$$

$= 77.6$

Note

- If you put decrement by -1 It makes a loop of 32,000



- Write an algorithm to find the avg grade for a class With an unspecified number of Students (0-100)
  ↑ Range

```
set sum equal to zero
set Count equal to zero
[Ask user to enter grade or -1 to
 Stop
 Read grade and save as gr
While gr is not equal to -1
    Add grade to sum
    Increment count by one
  [Ask user to enter grade or -1 to stop
   Read grade and save as gr
END While
  If count is greater than zero
     set avg equal to sum divided by count
     Print avg to screen
Else
     Print " No grade entered " to screen
ENDIF
```

Priming the pump

- للعلاقة التي نعرها ونفحص متاخرة

~~- Sentinel~~

- Sentineel :

إشارة تدل على التوقف عن العد

عندما نعرف عدد اللفات

- when we don't Know the times of loop we use away called Priming the pump

- هنا الـ Sentineel لهاعلاقة بالـ input لأن الـ Range محدد

; Write an algorithm to find the average grade for a class with an unspecified number of students ②

```
    Set sum equal to zero
    Set count equal to zero

Ask user to enter whether to continue or not (y/n)   yes / no
Read answer and save as ans  →  ال user يدخل ال initial value
While ans is not equal to 'n'  →  final value
        Ask user to enter grade
        Read grade and save as gr
        Add gr to sum
      Increment count by on   →   change
      (The rest is the same)

End while
```

* هنا ال sentinel ليس له علاقة بال input لان ال Range غير معروف

IN the Computer :-

```
  #include <stdio.h>
int   main(  )
 {
        int  n1 , n2 ;
        int  sum  ;
```

lecture 9

1 Ask user to enter first number

2 Read number and save as num1

3 Ask user to enter second number

4 Read number and save as num2

5 set sum equal to num1 plus num2

6 Print sum to ~~screen~~

```
※ include <stdio.h>
int main( )
{
   Int num1, num2;
   int sum ;
 1 printf ("Enter first number \n")
 2 scanf ("%, &num1);
 3 printf ("Enter second number");
 4 scanf ("%d", &num 2);
 5 sum = num1 + num 2;
 6 Print f ("sum=%d", sum);
   return (0);
}
```

standard input/output header file

data Type

integer

variables

\n → This means go to the next line

stops the program

adress

عنوان واسم اكل الذي سوف توضع فيه القيمة .

Assignment statement

success

▇ : always exists you have to know Them by heart

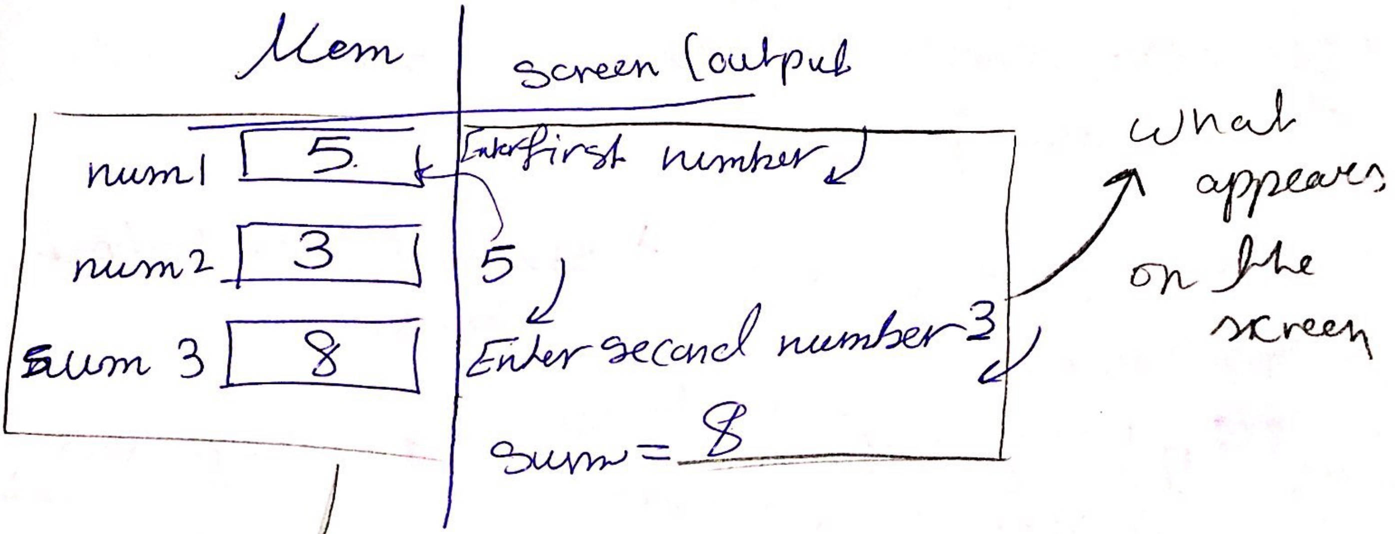
. important Search for Camle Notation

= Notes :- 1- Since num1 and num2 is the same type we can put them in one line and we end it With (;). if they were differe we split them with (;)

2- Variables name can include letters + numbers + underscore But it can't start with a num s.a 2nx → wrong / Fir_num } right

→ you can't name variables With Key Words (res
words) such as main , int .. (Words that
exists in the basic sentences of the program

• In The Memory :-



what happens in the memory (we don't see it)

• On the computer :-

compile (build)
Code Block
run (link)

• If u made a miskake It either shows an error or a warnings

error:
- you can't run the program So you should have 0 errors
- logical error
- run-time errors s.a: number/0 المسا -- صفر
- syntax كتابة !

warnings: → . < sometimes they don't affect your program

Try to make them 0 warnings

But you should correct it Bcz sometimes it's dangerous to leave them

UPLOADED BY AHMAD JUNDI

Examples of errors

① $X = 5$;

$y = X - 4$;

$y = y - 1$;

$Z = 3/y$; ⟹ run-time error

② if the program is to sum and i put − instead of + ⟹ logical error

③ $avg = (\overset{50}{g_1} + \overset{100}{g_2})/2$; → you should put ( )

avg → 100 ⟹

lecture 10

- constants

```
#include <stdio.h>
# define PI   3.14
int  main ( )
  { int rad;
    float area ;
    Print f ("Enter radius\n");
   scan f ("%d", &rad);
   area = PI * rad * rad;
   Print f ("Area=%f", area);
  return 0;
 }
```

constants

- usually we use All capitals for constant
- Constant ␣ it's valu (No =)
  space

- Types
- To read and write x :-

if x is	int	float	double	char

	scan f("%d", &x);
	Print f("%d", x);
		Scan f ("%f, &x);
		print f("%f, x);
			scan f (% lf", &x);
			print f ("%f , x);
				scan f ("%c" &x);
				print f ("%c", x);

space

Exampl

```
int age;
char gender;
Print f ("Enter ~~age~~ age and gender /n");
scan f ("%d %c", &age, &gender);
```

↳ we put a space so it doesn't take the chartcher "space"

- When you enter you put a space if the 2 numbers are float

- binary operations

Arithmetic operations.

+	3+2=5
−	3−2=1
*	3*2=6
/	3/2 =1
%	3%2=1 , 6%10=6

Precedence Rules :-

$$X = 5 + \underbrace{2 * 3} + \underbrace{4/2} * 5$$

$$= 5 + 6 + \underline{2 * 5}$$

$$= 11 + 10 = 21$$

* / % } stronger
+ − : weaker

$$X = [(5+2) * (3+4)] / (2*5)$$

?

$$= (7 * 7) / 10 = 4.9$$

- Type casting

```
int   X=3, y=2; a;
float  Z=2.4;

X=3/2 ;   → 1
Z=3/2;    → 1.0
X=3/2;    → 1
Z=3/2.0;  → 1.5
```

لكي تحصل على الجواب الكسر
يجب ① ان يكون البسط او المقام
او الاثنين مكتوبا به ك float
مثل 2.0
② يجب ان يكون المكان الذي
نضع فيه الجواب float
مثل Z

a = x/y ; → 1

b = x/y ; → 1.0

$\underbrace{a = x}$/y.0 ; →

↳ To make it float for once we use type casting

⇒ a = (float) x/y ; → 1

z = (float) x/y ; → 1.5

z = x / (float) y ; → 1.5

z = (float) x/ (float) y ; → 1.5

z = (float) (x/y) ; → 1.0

int [illegible] floats

↑

Note : • We can use z = x ; without type casting

But • x = z ; ∴ you have to explain that you want z to become an integer

↓ int

• x = z % y ;

↳ int

Example: $V = \frac{4}{3} \pi r^2 h$

$= \underbrace{4 / 3.0}$ * PI * r * r * h

↳ you have to put a zero

• output formatting

52␣3␣12345

2305 232

2␣345␣67

lecture 11

- output formating

x = 524 , y = 3 , z = 12345
= 2 = 1245 = 6
= 12345 = 1 = 713

Print f ("%d␣%d␣%d, x, y, z);

It will print

```
524␣3␣12345
2␣1245␣6
12345␣1␣713
```

If we put:-

Print f ("%8d␣%7d%9d", x, y, z).

```
_ _ _ _ _ 5 2 4|␣|_ _ _ _ _ _ 3|_ _ _ _ 1 2 3 4 5
            2            1 2 4 5                6
      1 2 3 4 5          1                  71 3
```

for integers and characteres

• اذا عدد الخانات 8 يكفي كله

لو عدد الخانات ما كفت بنه على الحد

مثال: %3d %7d

|1 2 3 4|

عدد الـ7 يبدأ من هنا

لو كان في space مثال:

%3d␣%7d

|1 2 3 4|␣

نضع الـ space ونبدأ

لو كان عدد الخانات سالب.
نلزقه على الشمال

Print f ("%5d%-2d%-7d", x, y, z)

```
|_ _ 5 2 4|3 _|1 2 3 4 5 _ _
         2|1 2 4 5|6 _ _ _ _
```

for a float

By default أخذ خانات

X = 73.624927   y = 5.261 , Z = 324.52

```
Printf ("%10.2 f%7.1 f_%-8.4 f" , x, y, z);     %__.__f
```

width ← 10 ; 2 → عدد الخانات (التقريب)

%__.__f : width , number of digits

```
|______73.62|____5.3|_324.5200
```

• الرقم يجب ان يكتب صحيحة ، اذا لم تكفي الخانات تكتب الرقم خارج الخانات

- Comments To write them there is 2 ways :

```
#______
int ______
  {
```

① /* ≡ */   /*——*/

or ② `Sum = X+y;   //add x to y`

- files

```
#include <stdio.h>
int main ( )
{ int n1, n2, sum;
  FILE * in;
  in = fopen ("data.txt", "r");
  fscanf (in, "%d %d", &n1, &n2);
  sum = n1 + n2 ;
  Printf ("sum=%d", sum);
  fclose (in);
  return 0;
}
```

• يعني هذا البرنامج يعتمد على Data موجودة في فايل اسمه data.txt

• يعني سهم يؤشر على file (Pointer)

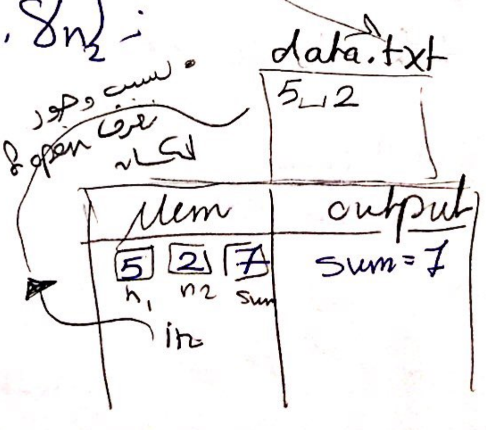
mode of the opened fi

• يعني هذا الملف سأفتحه للقراءة

أعطت عنوان لـ in

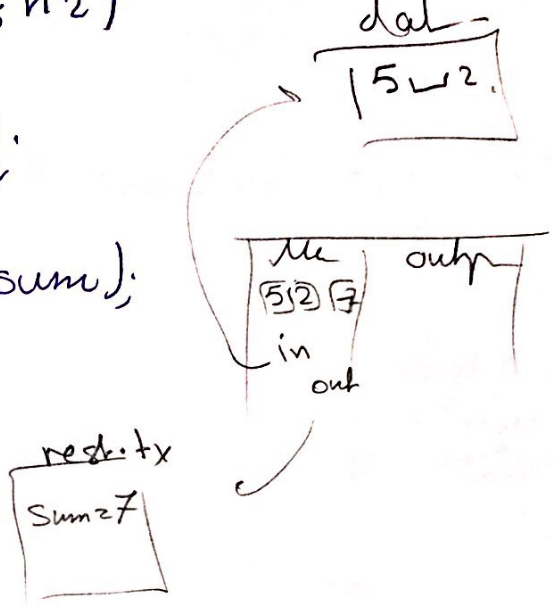
• بسبب وجود fopen عرف المكان

تلغي العلاقة بين in و data.txt



```
#include   <stdio.h>
int main ( )
{ int  n1, n2, sum;
File *in, *out;
in =fopen ("data.txt", "r").
out= fopen ("rest.txt", "w");
fscanf (in, "%d%d", &n1, &n2)
Sum = n1+n2.
Printf ("sum %d ", sum);

fprintf (out, "Sum=%d", sum);
fclose (out);
fclose (in);
return 0;
}
```

مهمة
يعني اذا كان
الشي غير
مكتوب
اكتبه واحتفظه
بدونها بكون الملف فارغ
او معلومات فرغ



lecture 12.

- functions

Built start here

```
#include <stdio.h>
int sum (int, int);
int main ( )
{
```

→ • function Prototype (اسمها، نوع الاشياء التي تاخذها، نوع الاشياء التي تخرجها)

هنا: sum, int, int

fun starts here

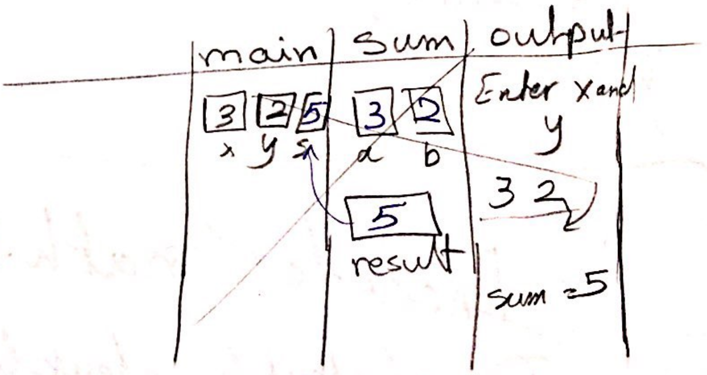
```
int x, y, s;
Printf("Enter x and y \n");
scanf ("%d %d", &x, &y);
s = Sum(x, y);
Printf ("sum = %d", s);
return 0;
}
```

→ function Call لاستخدامها

```
int sum (int a, int b)
{
int result;
result = a+b;
return result;
}
```

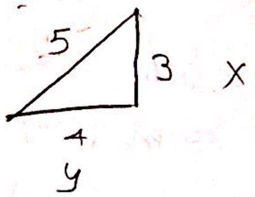
→ definition

→ a small program to explain sum



• Two Types

1) system defined functions
2) User defined functions



$z = \sqrt{x^2 + y^2}$

1) S.d.f :

```
#include <stdio.h>
#include <math.h>
int main()
 {
  int x, y;
  float Z;
 Printf ("Enter x and y\n");
 scanf (" %d%d, &x, &y);

Z = sqrt( pow (x,2) + (y*y) );
Printf(" Z = %.2f , z);
 return 0;
}
```

Include <math.h>

	Pow (double , double)	$\xrightarrow[us]{gires}$	double
	sqrt (double) ;	→	double
أكبر عدد صحيح ←	floor (double) ;	→	int
يأخذ الرقم الصحيح اللي بوضع الأكبر 2.5→3 ←	ceil (double);	→	int
القيمة المطلقة int ←	abs (int) ;	→	int
" " double لـ ←	fabs (double) ;	→	double

sin(double) → double

cos (~) → ~

tan (~) → ~

sec (~) → ~

cos (~) → ~

Cot (~) → ~

2) U.d.f

sum (int, int) → int ✓

sum ( ) → int

sum (int, int) → (Nothing) void ✓ يأخذ

sum ( ) → (Nothing) void الارقام

①

lecture 13

void

Void: يأخـــذ ولا يرجع

```
#include <stdio.h>
int sum(int, int);        → if we put void ( هذا الفنكشن لا يعطينا شيء )
int main()
{
   int x, y, ~~s~~;           we cancel s
   printf("Enter x and y\n");
   scanf("%d %d", &x, &y);
   ~~s=~~ sum(x,y);
   ~~printf("sum=%d", s);~~

   return 0;
}
 void
~~int~~ sum(int a, int b)
{
   int result;
   result = a+b;
   ~~return result~~;
           printf("sum=%d", result);
}
```

Question for void :-

2 numbers and It prints the two numbers (No output) (output is done by Sum Not main)

for the first (No void)

main	Sum	output
2 4 6 x y s	2 4 6 a b	Enter x and y 2 4

main	Sum	
2 4 x y	2 4 6 a b res	Enter x & y 2 4 sum=6

( )

```
int sum( );
int main( )
{
{ int s;
 s=sum( );
 Printf(" sum=%d", s);
 return 0;
}
```

يقوم ب Processing & input

هو الذي يقوم ب output

( ) : لايؤخذ ويرجع

مثل main( )

```
int sum ( )
{
 int x,y, result;
 Printf ("Enter x and y\n");
 scanf ("%d%d", &x, &y);
 result = x+y;
 return result;
}
```

main	sum	output
$\boxed{6}$ s	$\boxed{2}$ $\boxed{4}$ x y $\boxed{0}$	Enter x and y Sum=6

void + ( ) :-

```
 Void sum( );
 int main ( )
{
    sum( );
 return 0;
}
```

```
void sum( )
{
 int x,y, result;
 Printf ("Enter x&y\n")
 scanf ("%d%d", &x, &y).
 result = x+y
 Printf ("sum=%d"}, s result)
{
```

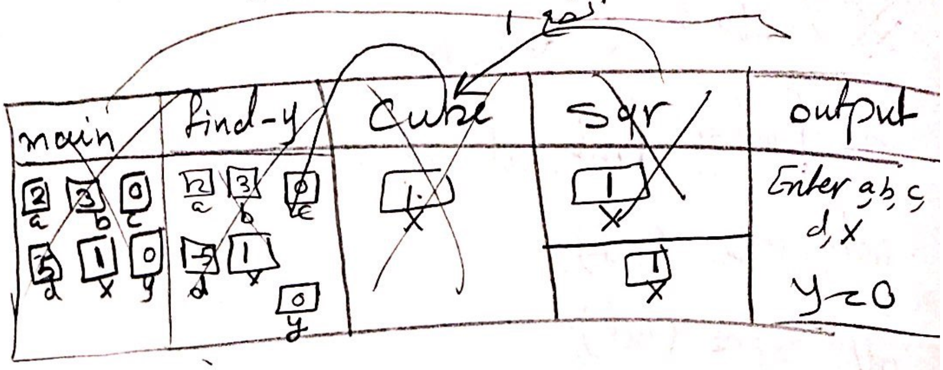
pli: $y = \underset{a}{3}X^3 - \underset{b}{2}X^2 + \underset{\overset{0}{c}}{0x} + \underset{d}{5}$

برنامج يدخل a,b,c,d و x و يطبع y

```
#Include <stdio,h)
int find-y (int,int,int,int,int).
int cube (int);
int sqre (int);
int   main ( )
{
  int a,b,c,d, x, y ;
 Printf (" Enter a,b,c,d,x\n");
  scanf ("%d%d %d %d%d ", &a, &b, &c, &d, &x);
  y = finct y ( a b c d x);
 Print f ( " y=%d" , y;
  return 0 ;
 }
 int find-y (int a, int b, int c, int d, int x)
 {
  int y ;
  y = a * cube(x) + b * sqr(x) + C * x + d;
  return y;
 }
```

```
int cube (int x)
{
  return ~~x**~~ ·sqr(x) * X;
}

int sqr (int x)
{
  return X * X;
}
```

$2x^3 + 3x^2 - 5$



lecture 14

Selection

```
if (x>5)
 Printf ("good");
else
   printf ("bad");
```

First :-
→ relational operators :-

<	less then
>	more (greater) than
<=	less or equal
>=	greater then or equal
==	← للمقارنة
!=	not equal

اما = هي
معطى

Ex:-

```
if (x% 2==0)
   printf ("even");
else
  printf ("odd");
```

But if you write it like this :-

```
if (x%2)
   printf ("odd")
else      ~        ~ ("even")
```

false

• المعنى هنا اذا قسم الرقم
على 2 وكان
الجواب صفر
فانه تعتبرها zero/
ويذهب إلى else
اما اذا كان الجواب أي
شيء غير الصفر فانه
يعتبرها True ويطبع
odd

second :- logical operator :-

- &&	and
- ||	or
- !	Not

• Note :- 1 This is wrong

```
if (x=5)
else
```

should b ==

2 In C language

zero → false

non-zero → true

3 If you write :-

x=2; x=0 , x=5

```
if (x=2)
   prinf("good");
else
   printf("bad");
```

It will print good for three cases

If you put :-

```
if (x=0)
```

It prints bad because zero = false In C

Ex1- If ((age > 20) && (gender == 'f') || (age > 85))

only in C
If it gi

```
If (x)
  printf("good");
else
  = ("bad");
```

Now If x=0
Then false
(It prints bad)
if x is anything
else It prints
good

* It's either all of it is positive
or all of it is false

X	y	x && y	x || y	!x
T	F	F	T	F
F	T	F	T	T
T	T	T	T	F
F	F	F	F	T

Ex ag = 15 , gender = 'f' , avg = 70

if (F && T || !(F))
= (F && T || T) it starts with and
= (F || T)
= T

→ !( ) is the most important
→ && is more important than ||
→ parenthesis ( ) is the most important

if (F && { T || !(F) })
(F && { T || T })
(F && T)
F

```
If (num >=0 );
   Printf ("%d is positive", num);

else
   Printf ("%d is negative", num);
```

```
Ex :   If (mark >=90); && (mark <=100)
       {
          printf ("grade is A \n");
          printf (" good job \n");
       }
```

*This means all the conditions should be True to take the result

```
else if   (mark >=80)
      printf ("grade is B\n")

else if   (mark >=70)
      Print ----- is c
else if   (mark >= 60)
      ---------- is F
```

. It's a one If So It goes to each condition and if one is True it stops.

Short circuits

```
X=5;      y=10;
if((X >6)  && (y<20))
   good
else
   bad
printf("%d \n", y);
```

If X=15

F T

=20

T

Now if It was T && F = F so

y =0 It is false It prints bad and 0

It will print good and 20

* a program to know wheather a letter is a vowel or not:-

```
Char letter;
printf ("Enterletter\n");
scanf ("␣%c ", &letter);
if (letter == 'a' ||'o' ||'i' ||'u'|| 'e' )
   printf (" %c is a vowel, letter);
else
   printf ("%c is not a vowel, letter);
```

Those are not zero so It's All T

This is Wrong

you write : if (letter == 'a' || letter == 'o' || letter == 'i' || letter == 'i' || letter 'e' - - - - - - -

Lecture 15 :-

• Questions:
*Gives you switch and Wants If for the same case or opposite

```
If (num ==1)
    printf ("one\n");
else if (num== 2)
    printf ("two\n");
else
    printf ("No such Number\n");
```

---

using switch

•Notes
•• It works only with == (Doesn't work with < >)
•• 〃 〃 〃 things that we know what Comes after or be fore (it doesnt work with float & string)
↳ S.a : integes & Ascci code (a,b,c--)

```
switch (num)
{
Case 1: printf ("one\n");
        break;
Case 2: printf ("two\n");
        break;
default: printf ("No such number\n");
}
```

break; → break is important to stop the program

default → you can put break but no need after default bcz the program ended any way

Exp: Vowels using switches:-

```
switch (letter)
{
Case 'a': Case 'i' : Case 'o' : Case 'e' : Case 'u':
Printf ("%c is a vowel", letter);
break;
default: printf ("%C is not a vowel", letter);
}
```

it doesnt have to be here

- x y

```
if (x>y)
   printf ("%d is larger", x);

else
   printf ("%d is larges", y);
```

- x y - z a b
- we assume that the first number entered is the min
- X y z a b
  5 2 4 3 2

↳ assuming x = min;

```
          if (y< min)
             min = y;
          if (z< min)
             min = z;
```

- you can't use swith in this case
- you need a loop

Ex : A program to print the number that is between

→ x y z
  2 1 7

It has to print 2

```
if ( (x>y) && (x<z) ) || ((x>z) && (x<y))
       printf ("%.d", x);                  )
else if ((y>x ) && (y<z)) || (
       printf ("%.d", y);
else print ("%.d", z );
```

Ex: Enter a formula & get the answer

nput → 5+2

output 5+2=7

```
printf ("Enter formula\n");
scanf ("%d ␣ %c %d", &n1 , &op , &n2);

switch(op)
{
Case (+) :   result = n1+n2;
             break;

Case (-) :   ________
             ______
             |
             |
             ,

}
printf ("%d %c %d =%d", n1, op, n2, result);
```

on the screen
Enter formula
5+ 3
5+3= 8

Nested if:-

```
   x=5 , y=3;
   if (x >y);
     ~~{~~
     if (x==5)
adding      printf("good\n");
       ~~}~~
        else
     {
   add printf ("bad\n");
   printf ("bye\n");
     {
```

else for the first if

• on the screen — after adding

good	x=5, y=3 good	x=5 , y=3 good bye	x=5, y=3 good bye
—	x=5, y=8 bad bye	x=5, y=8 bye	x=5, y=8 bad bye
	bad bye	x=3 , y=5 bye	x=3, y=5 bad bye
bad bye	—	x=7 , y=5 bad bye	x=7, y=5 bye

ecture 16

Boolean function:

function لا ترجع ↑ عدد
لكنها ترجع جواب نعم أو لا .
(True or false

```
int isEven(int n)   ← it's like asking is this even?
 {
 if (n%2==0)
      return 1;

 else
     return 0;
 }
```

```
main()
{
 int x=5;
 if (isEven(x))
     Printf("Even");
```

Another way

```
int  isEven(int n)
{
  if  (n%2)
       return 0;

  else
       return 1;
}
```

The shortest way:-

```
int  isEven(int n)
{
   return !(n%2);
}
```

* loops :-

while
for
do/while

• 3 Basic Components of loops:-

```
Ex: int x=2                 ⤳ 1- initial value
    While (x<=5)            ⤳ 2- final value
    {
      Printf("hi\n");
      X++;                  ⤳ 3- change
    }
```

```
2  hi
3  hi
4  hi
5  hi
6  ?
```

UPLOADED BY AHMAD JUNDI

Scanned by CamScanner

Ex1 X=5;

X++;

printf("%d",x) ; it prints 6

Ex2 X= 5;

++X;

Printf ("%d" , x); it prints 6

Ex3 X=5;

y = ++X → Add 1 to X and Then do everything

printf("%d", x); it prints 6 But y = 6

Ex4 X =5.

y = X++. → Do everything then add 1

printf ("%d," x); 6 But y = 5

Ex5 X=5;

Printf ("%d," X++); it prints 5 Bcz he does The print then he adds one.

Ex6 X = 5.

printf ("%d, ++X); it adds one Then it Prints

So it prints 6

- you can use the ++, -- way for ~~to~~ operations : S.a:-

```
X= X+7;  ≡  X+=7;

X= X*2;  ≡  X*=2;

X++;     ≡  X+=1;
```

Ex:- factorial :-

n!

5 → 5! ⇒ We need a loop !

5x4x3x2x1
or 1x2x3x4x5

```
i=n      (initial value)
while (i >= 1)
{
   result = result * i;   // result *= i;
   i--;
}
```

- In case your going down

→ result = 1
i = 5
→ result = 1*5 = 5
i = 4
→ result = 5*4 = 20
i = 3
→ result = 20*3 = 60
i = 2
→ reslt = 60*2 = 120
i = 1
→ result = 120*1 = 120

• in case your go
up

```
i=1
while (i ~~<=1~~ <=5)
{
  result = resullt * i; //result *= i;

  i++;
}
```

Ex: x^y $2^3 = 2*2*2$

```
 result=1;
 i=1;
while (i<=y)
 {
   result= result * x ;

   i++;
 }
```

$3^4 \Rightarrow x=3, y=4$

~~#~~ i=1

result = 1*3=3

i=2

result = 3*3=9

i=3

" = 9*3=27

i=4

" = 27*3=81

• if i wanna make it as a function

```
int myPow (int x, int y)
{ int result = 1;
  int i = 1;
  while (i <= y)
{
    result = result * x;
    i ++;
}
~~rett~~ return result;
}
```

```
int z = pow(7,50);
(5 9 7);
```

```
sum=0;
i=0;
while (i<10)
{
   printf ("Enter grade \n");
   scanf ("%d", &grade);
   sum += grade;
    i++;
}
  avg = (float) sum/10;
```

```
printf (" Enter grade or -1 to stop \n");
scanf ("% ", &grade);
while (grade != -1)
{  sum+= grade;

}
```

lecture 17

Types of loops
while
for
do/while

for loops

```
x=1 ;
while (x<= 5)
{
    printf ("good\n");
    x++ ;
}
```

The same

```
for (x=1 ; x<=5 ; x++)
    printf ("good\n");
```

- If more than one variable controls the loop

```
x=1 ; y=10;
while ((x<=5) && (y>=5))
  {  printf ("hi\n");
     ~~x++~~
     x++;
     y--;
     ~~x++~~
  }
```

in the same way:-

```
for ( x=1, y=10; ((x<=5) && (y>=5)) ; x++, y-- )
        printf ("hi\n");
```

X^y

```
result = 1;
i = 1;
while (i=<y)
{
    result *= X;
     i++;
}
Printf ("result=%d", result);
```

in the same way:-

```
for ( result = 1, i=1; i <=y; i++)
        result *= X;
Printf ("result = %d", result);
```

Ex:
```
for (i=1; i<=1000; i++);
        Printf ("good\n");
```
→ this means don't do anything

• ف هنا يطبع good مرة واحدة

• تستعمل لعمل Pause للعرض في حالة أردنا ان نفصل مدة زمنية بين كل output

do/while

• برنامج يحسب عدد الخانات

```
X=12547;
sum=0;
Count=0;
While (X > 0)
{  digit = X%10;
   X= X/10
   Count ++;
   sum = sum + digit
}
Printf ("Count=%d", Count);
  "    (" sum=%d", sum);
```

• How it works

X= 1254x
count=0

X= 1254
count=1

X= 125
count=2

X= 12
count=3

X=1
count 4

X=0
count=5

If you want to add digits

Using do while:-

```
X=12547;
Count=0;
do
  {  X=X/10;
     Count ++;
} while (X>0);
```

• نضعها لان الشرط جاء في أخر الحلقة

• do/while is the only one That you have to enter the loop for at least once

• برنامج يطبع قواسم العدد 100

• Ex

i=1;
while (i<=~~100~~ 50) or (i<~~[illegible]~~100

```
{
     100
 if (n%i == 0)
      printf ("%d\n", i);
    i++;
}
```

```
i = 2;
while (i<n)
{
    (n%i == 0)
       Printf ("%d is not prime\n", n);
   else
     {Printf ("------ is prime", n);
   i=n}
}  i++;
```

• برنامج يعطينا الأرقام الأولية

We use a Technique called: Flag Technique

↳ you don't print till you make sure

```
Prime = 1;      (True)

i = 2   ;
 while (i<n)
 { if (n%i == 0)
     Prime = 0;
   i++;
 }
```

lecture 18

break & continue :

```
X= 3;
while (X<7)
{
  printf ("%d \n", X);
  if (X==5)
  ~~break~~; continue
  X++;
}
Printf ("bye\n");
```

• لا تكمل اللفة بل ارجع افحص الشرط

it means →

continue → it prints
3
4
5
5
5
:
infinte loop

break → it means ازا تحقق الشرط اوقف ال loop

= it prints
3
4
5
bye

Example :-

```
X= 3;
while ( X++<10)
{
  printf("%d\n", ++X);
  if ( X==7)
  ~~break~~; continue;
  printf (%d\n", X);
}
printf ("%d\n", ++X);
```

break - It prints
5
5
7
8

* continue - it prints
5
5
7
9
9
11
11
13

```
int i, prime, n;
prime = 1;          →   for (n=1 ; n<=10000 ; n++)
for (i=2 ; i <n; i++)
    if (n%i ==0)
    {
```

```
    prime=0;
    break;
    }
if (prime)
  printf("%d is prime"", n);
~~else~~
  ~~printk~~ ("~~%d is~~ not ~~prime~~", n);
```

Nested loop:- (added by)

. They don't have to be the same loop

Prime function :-

```
#include <stdio.h>
int is-Prime (int);
int main(    )
  {
    int i,
    for(i=1; i<=10000; i++)
      if (isPrimie(i))
      printf("%d\n", i);
  return 0;
  }

int isPrime(int n)
{
    int i=2;
{ { While (i<n)
   if (n%i==0)
         return 0;
 }   i++;
                          return 1;
                          }
```

```
for (i=2; i<n; i++)
  if (n%i==0)
       return 0;
return 1;
```

ample: Give me the perfect numbers (number = مجموع قواسمه

ماعدا لـ ع نفسه

6=1+2+3 مثال

22=1+2+4+7+14

Example:-

```
n=4;
*
* *
* * *
* * * *  → To print those:-
           for (i=1; i<=n; i++)
               printf ("*");
```

```
To print * * *  ← Rows
         * * *  ←
         * * *  ←  ←
         ↑ ↑ ↑
         colums
```

```
for (i=1; i<=n; i++) i
{ for (j=1; j<=n; j++)
      printf ("*");
  printf ("\n");
}
```

```
i=1
* * *
i=2
* * *
i=3
* * *
```

To print

```
  *
  * *
  * * *
```

(Added by)

To print

```
* * *
* *
*
```

To print

```
- - *
- * *
* * *
```

```
for (i=1; i<=n , i++)
{ for ( j                 )
      printf ("   ");
  for ( K                 )
      printf (" * ");
  printf ("\n");
}
```

Lecture 19

```
- - - *
- - * *
- * * *
* * * *
```

```
for (i=1; i<=n; i++)
{
  for (j=n, j>i, j--)
     printf("  ");
                2*(i-1)
  for (K=1; K<=i; K++)
     printf("*"); printf("%d,i)
  printf("\n");
}
```

```
      1
    2 2 2
  3 3 3 3 3
4 4 4 4 4 4
```

you print K

```
      1
    1 2
  1 2 3
1 2 3 4
```

```
      1
    2 2
  3 3 3
4 4 4 4
```

→ you print i

printf("%d",i) instead of printf("*")

files and loops

```
int status;
status= scanf ( "%d", &x);
```

- you can use scanf to stop (when something is odd and not Known)

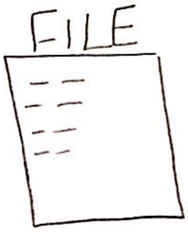
-1

```
status= fscanf (in,"%d", &x);
```

```
float avg;
int g1, g2 status;

FILE * in;
in = fopen ("data.txt", "r");
status = fscanf (in, "%d%d", &g1, &g2);
while   (status!= EOF)
{
      avg = (float) g1 /g2;
     Printf ("avg = %.2f", avg);
      status = fscanf (in, "%d%d", &g1, &g2);

}
```

• برنامج يستقبل علامات من File
و يتوقف عندما يطلب
منه ذلك

لو الملف فارغ
او غير موجود
يرجع EOF



Pointers := dynamic addreses

العنوان 5

```
int  x=5; z=30
int * y = &x;
```

في الذاكرة : لكل شيء مكان
وعنوان

Pointer نوع

X [5]
(100) → adres

printf ("%d", &x); → X↔100

يطبع الـ address ولكن
ليس الحقيقة
و هنا يطبعها 100

→ y

x [5]

(y) → 100

$\boxed{y = \& z;}$

• ولو كتبت
هيك تصير
y تؤشر على z

• يعني y تتغير

• الفرق لو طبعها X تبقى
ثابتة
ولكن y تتغير

مثال

y →
z [7]
200

• يطبع 200

What is the output? ②

```
int x=3 ; y=2;
int *a = &x ; *b;
Printf ("%d \n", a); ①
printf ("%d \n", &x);
b = &y; ②
Printf (%d \n", b);
a = b; ③
printf ("%d\n, *b);
* a = 20; ④
printf ("%d\n", y);
```

ما بتأشر على حدا

يطبع 100

يطبع 100

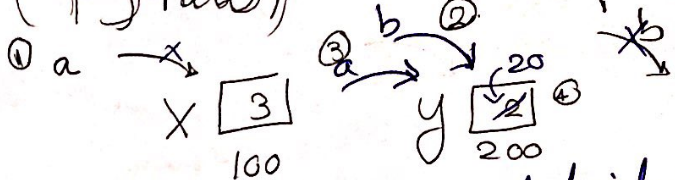
اصبحت طآؤشر على y

يطبع 200

مسموح لانه a و b نفس النوع

- تختلف عن int *
- تسمى بتحته indirection
- تعني أعطيني ما تؤشر عليه b
- يعني تطبع 2
- & تعني ال address

(Draw)



- what it prints (answer)

```
100
100
200
2
20
```

Notes:-

- a = X ; خطأ لأنه a و X ليسا من نفس النوع (int)

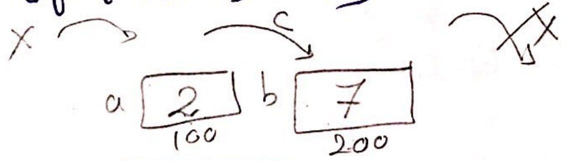
a = &X

- صحيح لان الاثنين pointers

What is the output

```
int a=2, b=7;
int *C=&b, * X;
printf (%d\n", *C);
*C = a+2;
printf ("%d \n", b);
X=&a;
* X = *X+5;
Printf (%d\n", a);
Printf ("%d\n, X);
```

a=b+a
Printf ("%d\n", *X);



```
7
4
7
100
7
7
```

UPLOADED BY AHMAD JUNDI

lecture 20 ①

تزيد عدد ال Pointers

- ترجع شيء function واحد ولكن تستطيع ارجاع اكثر من شيء عن طريق ال Pointers

#include <stdio.h> . Int*

~~int~~ ops (int, int, int *);

output parameters

int main ( )

{

argument int x=5 y=7, sum, diff;

. ops (x, y, &sum, &diff)

~~sum = ops (x, y, & diff);~~

~~End~~ ; printf ("sum=%d" diff =%d", sum, diff);

return 0;

}

• لكي ترجع ~~اكثر من~~ (بالرصاص)

•void ~~int~~ ops(int a, int b, int *d) , int *s) int a = x; int *d = &diff;

{

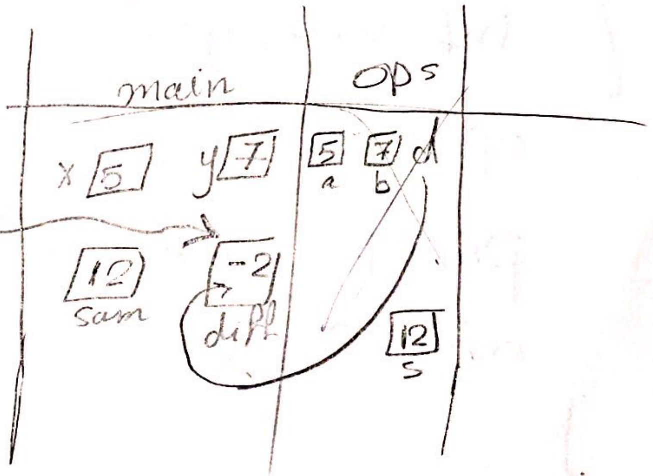
~~int s~~;

s = a+b;

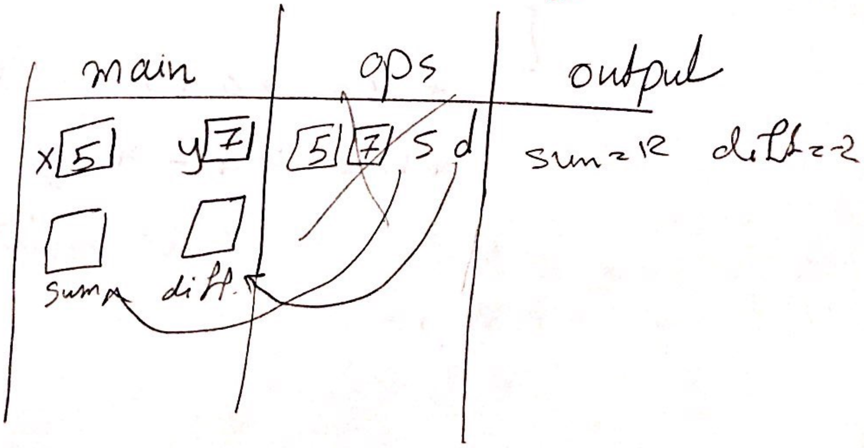
• *d = a-b;

~~return s~~;

}

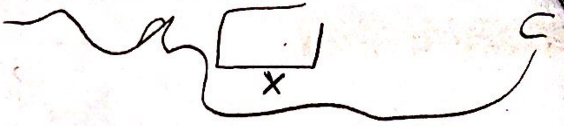


الفعلية تختلف



Scan f :-

int scanf (——, int * c)



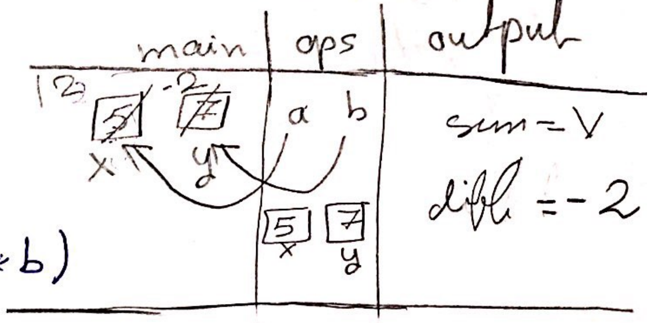
• اذا ما حطيت & ليدخل الـ Pointer لـ Trash ، يعني لا يوخذ ما داخل المربع بل ليذهب الى مكان غير معروف في الذاكرة

• مع & : ليؤشر على المربع واذا حطيت 5 سيضعها في المربع

• There is a Void * c (Void pointer) علومة المستقبل البعيد

```
#include <stdio.h>
Void ops (int*, int *);
  int main (   )
{ int x=5, y=7;
  ops (&x, &y);
  printf (" sum=%d  diff=%d ", x, y);
  return 0 ;
}
  Void ops (int *a, int*b)
     {
       ~~*a = *a + *b~~ ; wrong
       ~~*b = *a - *b;~~
      int   x= *a  , y = *b;
      * a = x+y;
      * b = x-y;
     }
```

Important

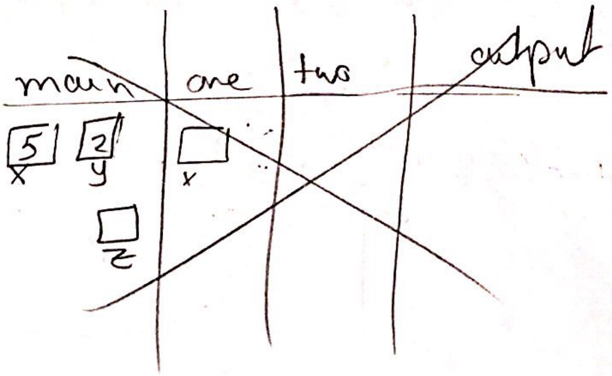


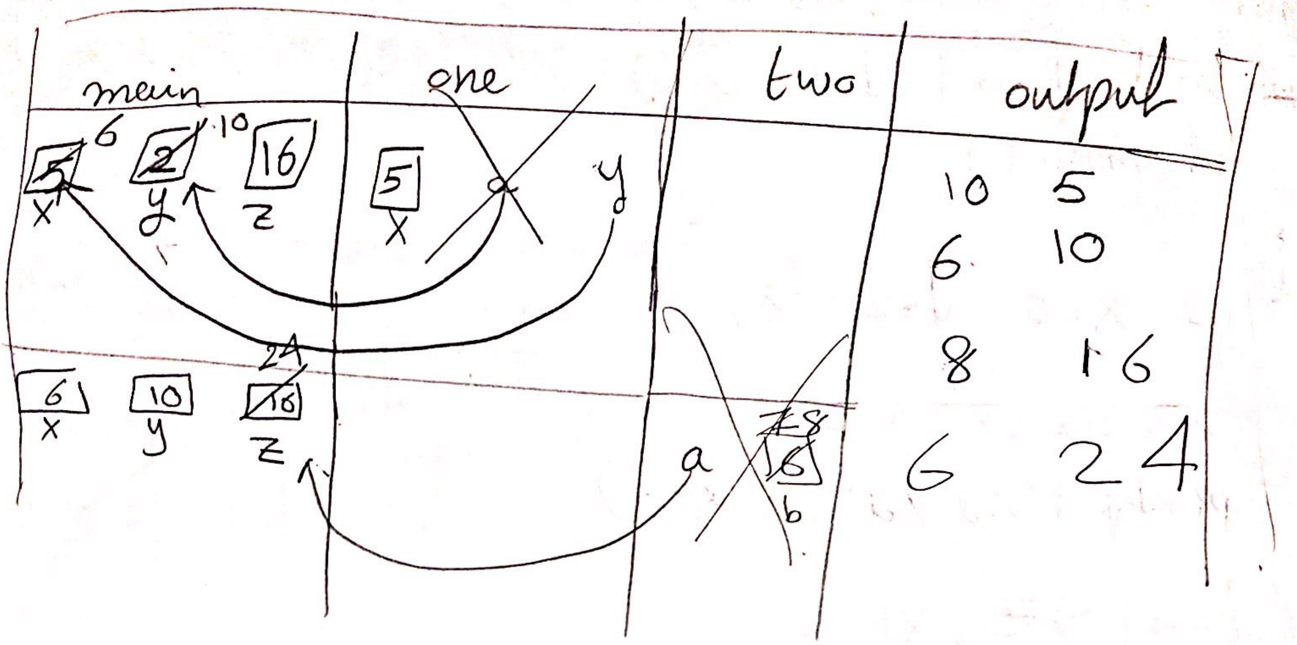
UPLOADED BY AHMAD JUNDI

Scanned by CamScanner

What is the output?

```
# include <Stdio.h> ;
int one( int, int* , int *);
~~int~~ void two ( int*, int );
  int main ( )
{
  int X=5, y=2 , z;
  z= one ( x, &y, &x);
  printf ("%d %d\n", x, y);
 two( &z, x);
 Printf ("%d %d ", x, z);
  return 0;
  }
int one (int x , int *a , int *y)
 {
   *a = x+*y;
   printf (%d %d\n", *a, x);
  (*y) ++;
   return *a+ *y;

 }
Void two (int *a int*b)
{
  b++;
  Print f ( "%d%d ", ++b, *a);
 *a = *a+b;
```




main
one
two
output
~~5~~ 6
~~2~~ 10
16
x
y
z
5
x
a
y
6
10
~~16~~ 24
x
y
z
a
~~7~~ 8
~~6~~
b
10 5
6 10
8 16
6 24

lecture 20

* global vs local variabiles

```
#include <stdio.h>
 int x=5.                 → global variabl
int one (int);
int main( )
 {
   int y
   y=x;
    ;
    ;
 }
int one (intb)
 {
   int x              →
   x=10;
    ;
 }
```

- لو في متغير اسمو نفس اسم الـglobal (X) مثلاً فانه بيغطي عليه وبيستعمل هو
- لو ما كان في ، بستخدم الـglobal
- لو int x ما كانت موجودة تتغير قيمة الـglobal وتصبح 10 مش 5

what is the output?

n	y	b	c
3	7	5	

```
#include <stdio.h>
int n=3.
int' first ( int *, int);
Void sec( int*, int ,int* ).
int main(.)
{
  int y=7, b=5, c;
  c= first (&b, n);
```

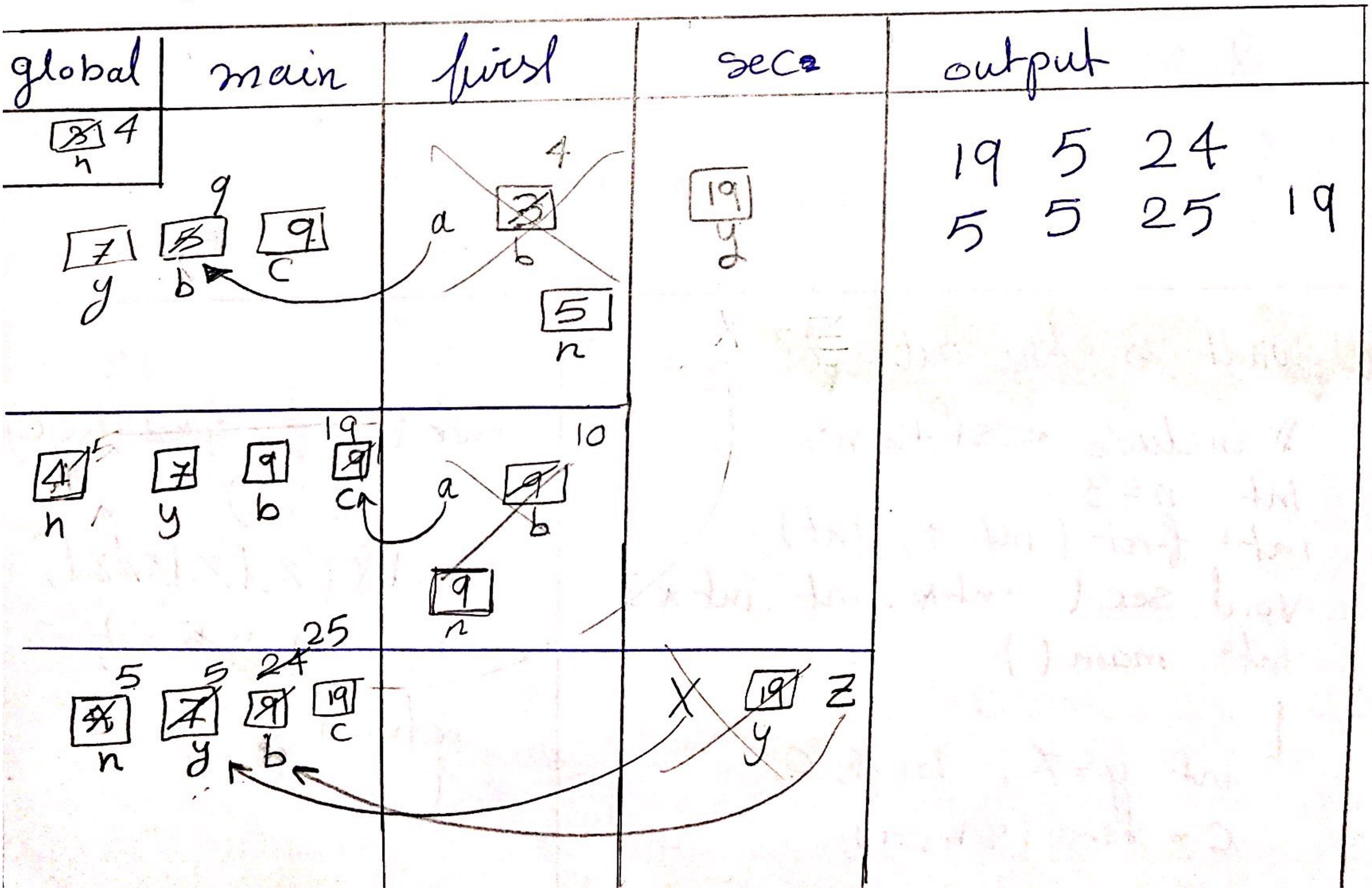
19

```
see( &y, first(&c,c)
     ,&b ).
printf("%d%d%d%d",
       n, y, b, c);
 return 0;
}
```

UPLOADED BY AHMAD JUNDI

Scanned by CamScanner

```
int first(int*a, int b)
 {
  int n=*a;
  b++;
  *a =n+b;
  return b+n;
 }

Void sec ( int *x, int y, int*z)

{
    *x=++n;
   *z= *x+y;
  printf ("%d%d%d\n", y,n,*z);
  (*z)++;

 }
```



```
19 5 24
5 5 25 19
```

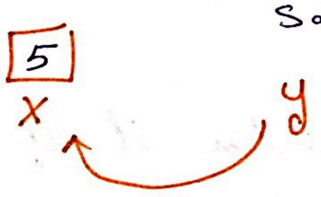
Pointers :-

↳ Dynamic address

→ How to define it ?

$\underbrace{\text{int}}$ $*y = \&x$;

نوع المؤشر ↑   ↑ الموقع الذي يؤشر عليه



- Notes :
  if a is a pointer and x is a variable Then you can't say x = a
  but you can say a = &x

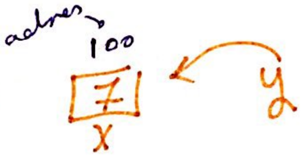
→ What is the Output ?

If you have a pointer *y :

→ Printf ("%d", y); → it prints the address ~~they~~ that y is pointing to it.

→ Printf ("%d", *y); → it prints the value inside the adress

Ex int x = 7;
*y = & x;



output

```
100
7
```

Printf ("% d", y);
Prinf ("% d", *y);

→ How to use pointers in functions ?

- Remember that a function ~~is~~ Can return one value only
  But we can use pointers to return more than one value from a function .

How ?
first
The function should be Void / Example: function to calculate sum / multipl division & substr

Void ops (int, int, $\underbrace{\text{int *, int *, int *, int *}}$)

You Can add As many As you Want

Second :-

When Calling the function :-

```
ops (X,y, & sum, &multi, &div, & sub)
```

Third :-

In the function it self :-

```
ops (int a , int b, int * d , int * S ,int *F, int *E)
  {
    *d = a+b;
    *S = a*b;
    * F = a/b;
    * E = a-b;
  }
```

Note

When it Comes to Printing when you have many function ; let's say we have 2 functions : Sum and sub

```
Z = Sum (X, y, S ----);
Printf ("%d\n", Z, x);

E = Sub (N, F ---- );
Printf ("%d\n", N, F ---- );
```

You work out Z

If there is another Print in the function it comes first

Then This

same

lastely this

Global vs local variables

- If you define a variable before main function. it's a Global variable

Ex
```
#includ____.
    int X=5
    |
    |
    |
    int main( )
```

- in functions, If there is a variable same as a global variable then :-

```
int one (int b)
{
   int X
   X=10;
```

You take this Value

But If it didn't say It's value then you take X=5

• لو كانت مش موجودة
نطبع قيمة global عشرة وليس خمسة

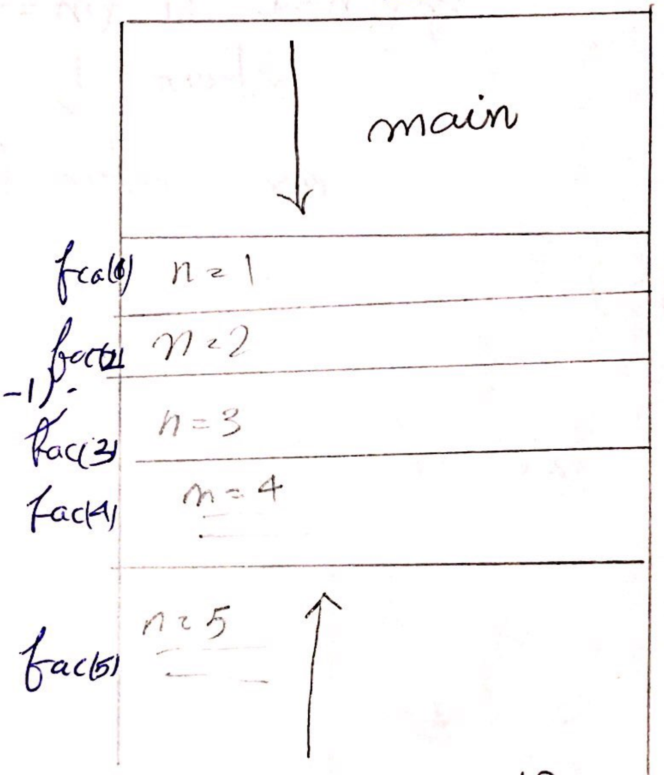
lecture 26

Recursion

fac(5) = 5 * 4 * 3 * 2 * 1
= 5 * fac(4)
= 4 * fac(3)
3 * fac(2)
2 fac(1)

Stack: مكدسة
يعني خط من فوق
مش من تحت
LIFO
last in first ou

```
int fac (int n)
{
  if(n==1)
    return 1;
  else
    return n * fac(n-1);
}
```



```
int unknown(int x, int y)
{
  if (y==1)
    return x;
  else
    return x+unknown ( x, y-1);
```

12
unknown (3, 4)
9
return 3+unknow(3,3)
6
3+unkow(3,2)
3
3+unknown (3,1)
ret 3

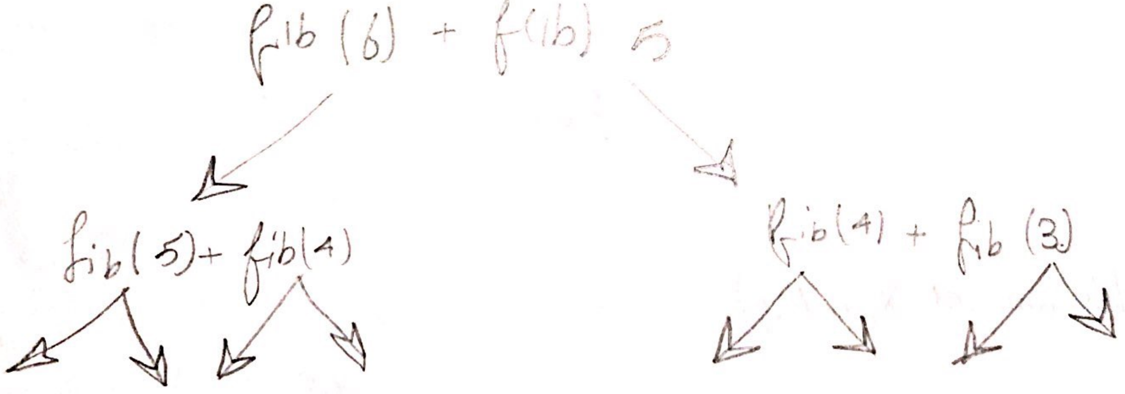
fibonacci :-

1	1	2	3	5	8	13	21 — — —
n=1	2	3	4	5	6	7	8 — — —

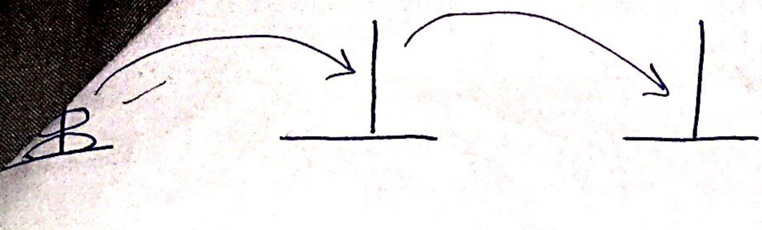
```
int fib(int n)
  {
     if((n==1) || (n==2))
          return 1;
        else return (fib(n-1) + fib(n-2));
```

• هنا البرنامج يعطي جواب , لكن يأخذ وقت طويل

Ex:- fib(7)



• الـ loop افضل من Recursion



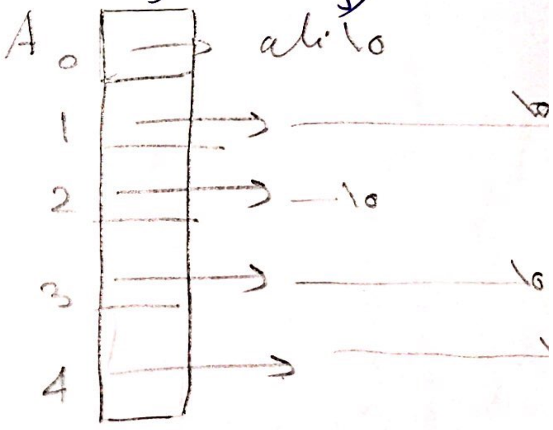
5234 16

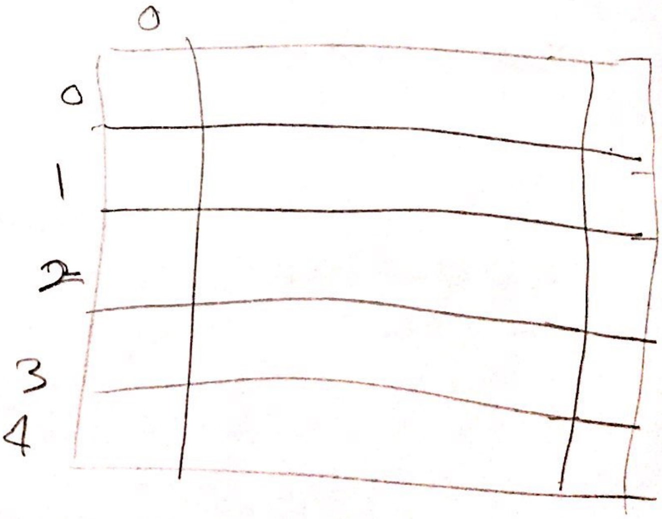
To print

5 2, 3 4 16

- array of pointers

char * A[5] = {"ali", "Ahmad", — };





• Parallel Arrays :-

	names
0	Ali
1	Sawsan
2	
3	
4	

grades	0	1	2
0	70	30	100
1	60	90	10
2			
3			
4			

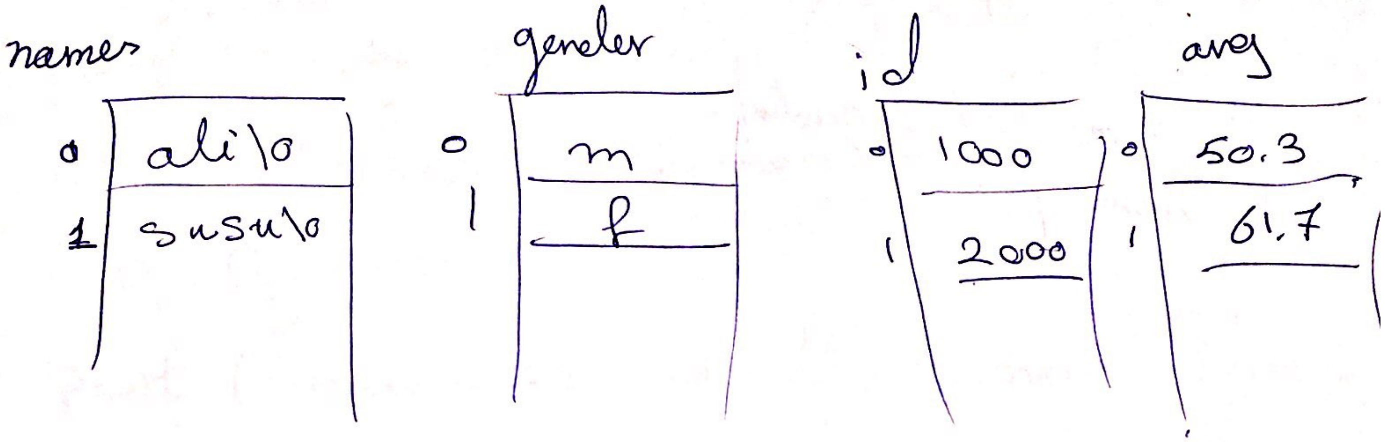
	ids
0	1000
1	2000
2	
3	
4	

	gender
0	m
	f

avg (float)
6.35
--

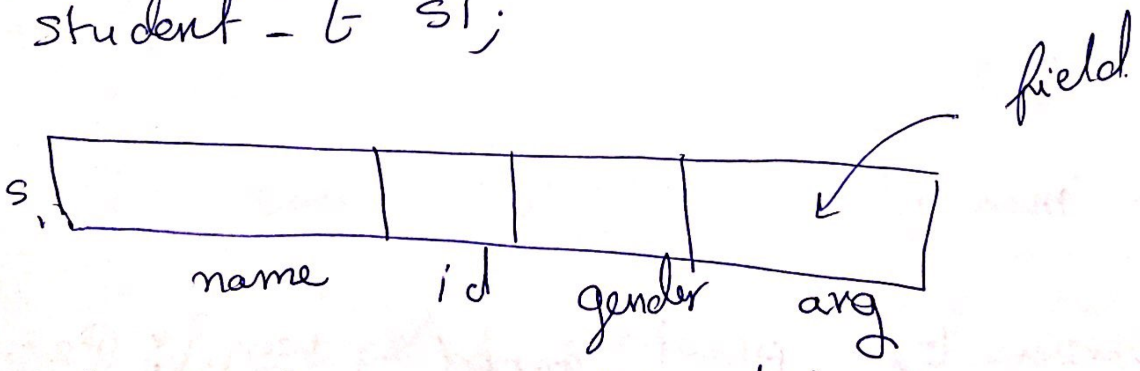
lecture 27

Structures :-



```
Typedef      struct
{
      char      name [10];
      int       id ;
      char      gender;
      double    avg;
}
     student_t ;
```

```
student_t s1;
```



```
strcpy ( s1.name , "ahmad" );
s1.id= 1000;                 s1.avg= 70.7.
s1. gender = 'm'.
```

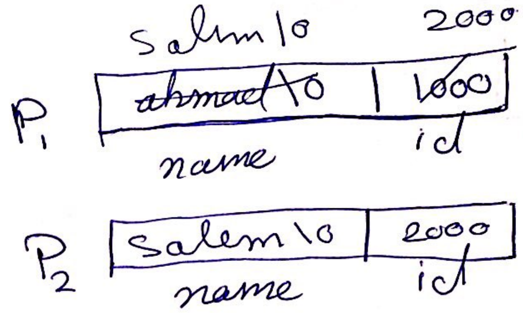
Array of structures:

student_t   students [10];

students

0	name	id	gender	-
1	name	id		

```
#include  <stdio.h>
#define  S 10
typedef  struct
{
    char  name [S];
    int  id ;
}
    person_t;

  x   y ;

int  main ()
{
 person_t;   p1= { "ahmad", 1000 }; P2;
p2 = P1; ✓   You Can do that
if(P1 == P2) X   You Can't do that.
printf ("Enter name & id \n");
scanf ("%s%d", P2. name, &p2.id);
  P1 = P2;
Printf ("name = %s  id%d", P1.name , P1.id);
 return 0;
}
```



output

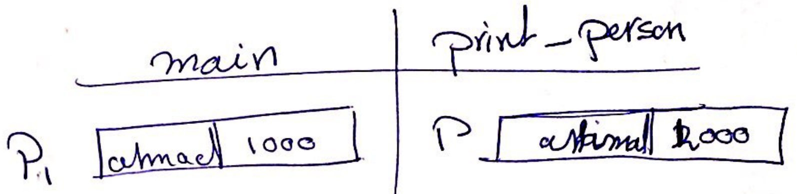
```
Enter na——.
salem  2000
```

Using a function

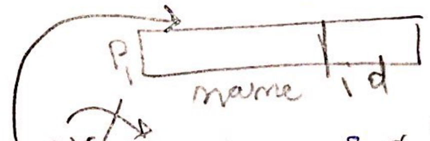
same definition

```
Void print-person (person_t);
 int main ()
   {
```

```
person_t    p1 = {"ahmad", 1000};
print-person (p1);
 return 0;
}
void print_person ( person_t   p)
  {
     printf (" name = %S   id =%d", P.name ,P.id);
  }
```



```
output
name = ahenel
  id = 1000
```

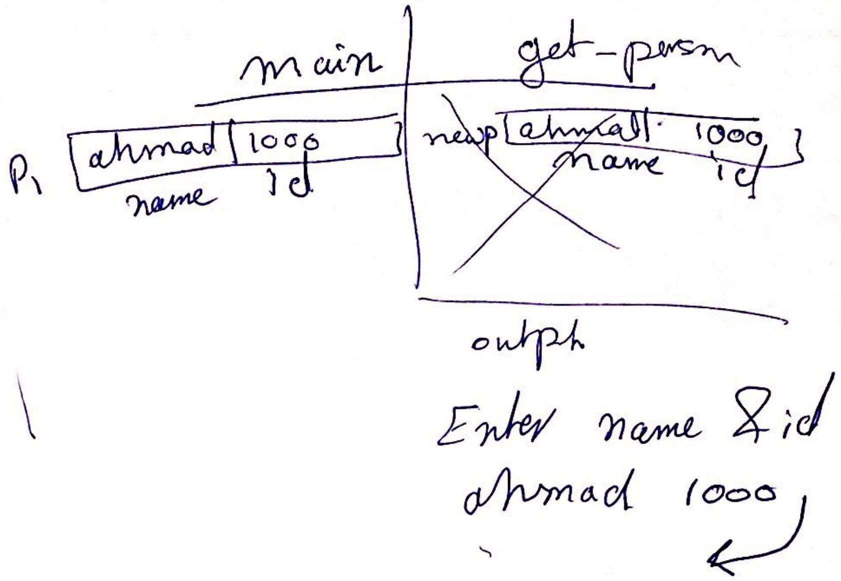


(*p_ptr).name { or p-ptr→ name

```
#
~~*~~ define  S 10
typedef  struct
{
  char  name[S];
  int  id;
} person_t;
Person_t  get_person ( );
int main ()
```

```
*p-ptr;
person_t  p1;
p_ptr = &p1
p1 = get_person();
return 0;
}
person_t  get_person
{ person_t  newp;
printf("Enter name
       & id \n");
```

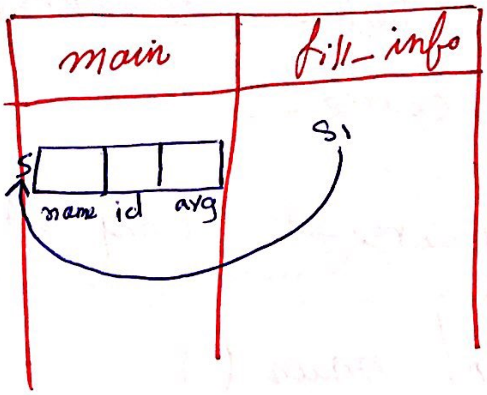
```
nf ("%s %d" , newp.name , &newp.id );

    return newp;
}
```



```
Enter name & id
ahmad 1000
```

lettor 28

```
#include <stdio.h>
#define S 10
typedefr struct
{
    char name [s];
    int id;
    double avg;
}
Void fill-info (stud-t);
int main ()
{
    stud-t s;
    printf ("Enter name, id and ~~z~~avg for student\n");
    fill-info (&s);
    printf ("%s %d %.2f", s.name, s.id, s.avg);
    return 0;
}
Void fill-info (student * s1)
{ ~~pf~~ scanf (" %s", (* S1).name);
       scanf ("%d", &(* S1).id);
     scanf ("%lf", &(* S1).avg);
}
```



ولكن string ما بنحط &

لو int / double بنحط &

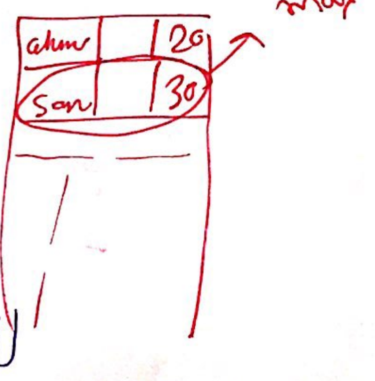
you can write
s1 → name
&s1 → id
&s1 → avg

UPLOADED BY AHMAD JUNDI

```
type-def     struct

{
    char   instructor [5];
    int    num ;
    stud.t    students [20];

} course-t

course-t  comp124;
```

```
int main ( )
 {

    stud_t   students [10], max_stude , temp;
    int  i ;
    for ( i=0; i <10 ; i++)
        fill info ( & student-[i] ); → or

    max_stden = students [0];
    for (i=0;  i<10; i++)
        if  ( students[i] .avg > max-student-.avg )
            max -student = students [i];
            printf (" nam___ = %s", max -st_.name);


    for (i=0;  i<5-1 ; i++ )
      for (j=0; j<5-1; j++)
      if ( student [j].id >
                    student [j+1].id)
```

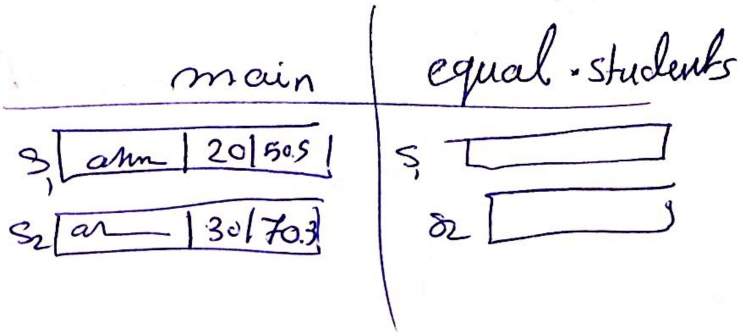
```
{      students[i]
                .ma
  scanf("%s, )
scanf("%d", &stud
               [i].id

scan_     _
    )
    )
}
```



```
temp = Students [j];
students [j] = Students [j+1];
 Students [j+1] = temp;
}
  Sum=0, Count =0;
for (i=0; i<10; i++)

 if (Strcmp(students[i].name, "ahmad")==0)
   {
     Sum += Students [i].avg;
     Count ++;
    }
   avg-ahmads = (float) sum/count;
```

```
#include <stdio.h>
int equal-students(student_t, student_t);
int main( )
 {
   Student_t    s1 = {"ahmad", 20, 50.B};
                s2 = {"ahmad", 30, ~~30.~~ 74.3};

    if (equal-student (s1, s2) )
       printf (" same\n");
       else
      printf(" diff \n");
    } return 0;
```

```
int ~~*~~ equal_Students ( stud-t s1 , stud-t s2);
{
    return (strcmp( s1.mame , s2.name) ==0
              && s1.id == s2.id
              && s1.avg== s2.avg)

}
```



lecture 29 :-

• Files :-

```
FILE  *in;
in = Fopen ("data.txt" , "r");

    if ( in == NULL)
    {
        printf ("Cannot open file");
        exit ( 0);
    }
```

يعني كلشي عام

. exit ( ) نقدر نحط أي رقم

```
FILE *in;

Char   filename [10];
printf ( "Enter fill name \n");
Scanf ("%s", filename);
in = fopen (file name , "r");
if (in == NULL)
 {
    Printf("Cannot open file %s", file name);
    exit (2);
 }
```

هذا الرقم يعني انه هناك Erro

Enter fith name

— data

الاسم غلط لانه الموجود في السيستم هو data.txt لذلك

— Cannot open

— Enter files

الجواب هون NULL الي يعني ع أحد الكف.

بيكمل وبيدخل في Loop ويرجع

```
FILE *in;

Char filename [10];
Printf (" Enter file name\n");
for ( scanf ("%s", filename); (in = fopen(filename,"r")) == NULL
    ; scanf ("%s", filename))
```

يطبع هذه ويطبع اللي بعدها ويطلب اسم ملف آخر

```
{
printf ("Cannot open file %S ", filename);
printf ( "Re enter file name \n");
}
```

8 تساوي Null

output (again)

```
Ent____
data
Cannot____ _ _
Re enter file____
dataa.txt
Canno____
Re____
data.txt
```

disadvantage
1- Nor readable
2- different between systems

hi.exe

Binary files :-

disadvantages of text files:-

1- processing time

2- precision problem

في حالة الـ float تكون هناك ارقام ناقصة بسبب التقريب

3- space

```
FILE *out;
int X=3456 , y=74;
fprintf (out, "%d %d" , x, y);
fscanf (in , "%d%d", &x, &y);
```

Memory chars مجموعة

```
X   010-----   → (3450)
                   10
y   01---
```

هذه char هذه الكلمة تأخذ الرقم

(3465)

↓

decimal حوله إلى

↓

(3465)

```
fprintf (out, "%.2f", x);
```

double وكان لدينا

File

```
3456␣74
5.27
```

How to open Binary files?

```
FILE *in;
in = fopen ("data.bin", "rb");
```

"rb" → read binary

عشكله وضع أي اسم

"wb" → write binary

```
fread (—, —, —, —);
fwrite (—, —, —, —);
fclose (in);
```

- الاختلاف يكون في القراءة والكتابة

↲ للقراءة نستخدم fread

↲ للكتابة نستخدم fwrite

كيف أكتب وأقرأ من file

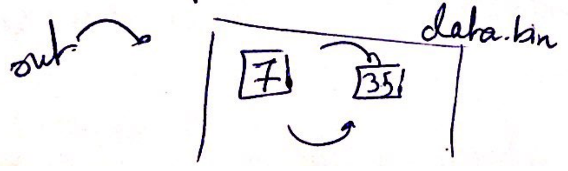
```
int x=7 , y ; float a=3.5 , b;
FILE * out;
Out = fopen("data.bin", "wb");
fwrite (&x ,sizeof (int), 1, out);
```

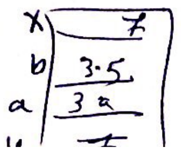
- عنوانه x
- اقتطع حجم int من الذاكرة memo
كم
على الشاشة

```
fwrite (&a , sizeof (float), 1, out);
fclose (out);
out =fopen ("data.bin", "rb");
fread (&y , sizeof (int) , 1, out);
fread (&b , sizeof (float) , 1, out);
printf (" X=%d . b=%.2f , x b);
```

8-bit → 2 byte
4 byte → float

int n=sizeof (int); → 2
sizeof (float); → 4
sizeof (X) → 2





X=7 , b=35

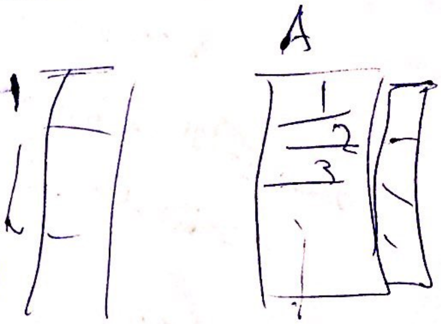
```
typedef struct
{
  char name [10];   → 4byte
   int id;          → 2byte
} stud_t;

 stud_t s1 = { "ahmad", 10};
  fprintf (out , "%s %d", s1.name , s1.id);
  fscanf (out , "%s %d" myname , & my id);
  fwrite( &s1 , sizeof (stud_t) , 1, out ).
  fread (&s2 , sizeof(stud_t), 1,out).
```

بصير الفايل وحطه في B آخر out

لما أعرف Array في الذاكرة لازم يكونوا مع بعض

write عكس read

```
int  B[10];
 int  A[5]={ 1,2,3,4,5 };
 fwrite (A, sizeof(int), 5,out);
 fread ( B, sizeof(int , 5 out ).
 fread ( &B[5], sizeof(int), 5,out) .   9
```

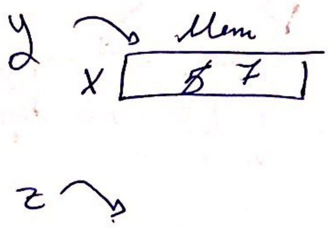


```
 70
 int n;
 n=fread (A, sizeof (int), 1000 , out).
```

ترجع { عدد صحيح قرأه من File

lecture 30

Dynamic allocation :-



```
int x=5;
int *y = &x;
int *z;
*y = 7;  ✓
*z = 3;  ∝
int A[1000];
int *A ;    int A[ ];
```

موجودة داخل #include<stdlih.h>

- malloc (memory allocate)

تنفع لأي نوع

```
int *x;
float *x
x = (int *) malloc( size of(int)) ;
```

ي

يعني خذ الشيء العام من malloc وحوّله ل int

void pointer

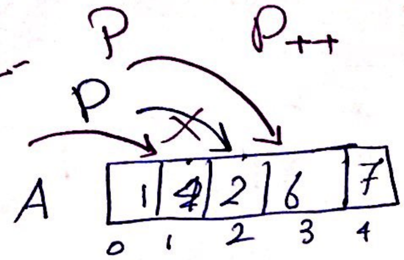
يحجز ١٠ int

```
x = (int *) malloz (sizeof (int) * 10)
```

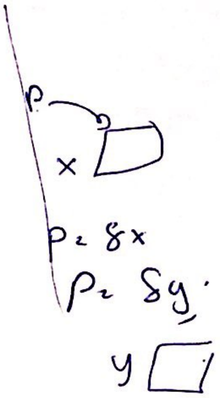
Calloc تستخدم لحجز مستمر

```
x = (int *) Calloc (10, sizeof(int));
```

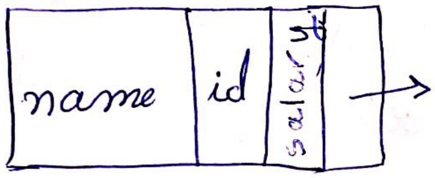
```
int A[5] = {1, 4, 2, 6, 7};

int *P;
```

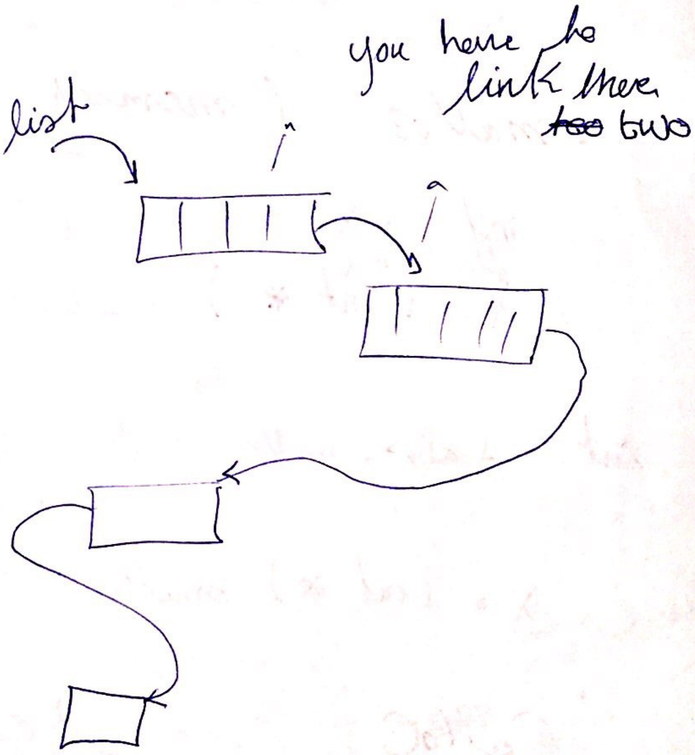


```
P = A;
P = &A[0];
P = & A[2];
```



- Structure ال آخر
linked list:





```
typedef struct stud_a
  {
    char name [10];
    int id;
    struct stud_a *next;
  }
    stud_t;
```

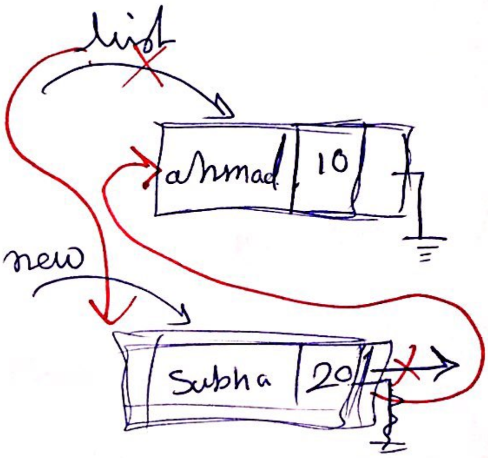
UPLOADED BY AHMAD JUNDI

Scanned by CamScanner

```
int main ( )
{   Char name[10]; int id;
    stud-t *list;

 Printf ("Enter name and id\n");
 scanf ( "%s %d", name, &id);
 list = ( stud-t *  ) malloc (sizeof (stud-t));

strcpy (list → name , name);

 list → id = id;
 list → next = NULL;
```

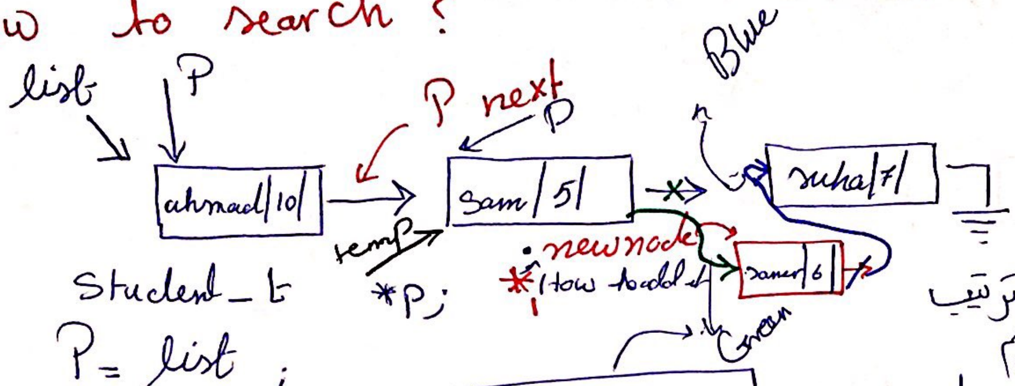
You define new ~~like~~ list linked

```
  new → next = list.
    list = new;
```



lecture 31

How to search?



```
Student_t *p;
P= list;
```

• لازم يكون بهاد الترتيب
ممنوع ان تعكسهم

```
while ((p != NULL) SS (p->id != X))
    P= p-> next;
if (P != NULL)
printf ("name = %S", p->name);
    else
printf ("%d No such id", x);
```

u have to add this →

How to add?

```
*1 :- newnode = (stud-t *) malloc( sizeof (stud-t));
      newnode -> next = P->next; This step (Blue)
      p -> next = newnode; This step (Green)
```

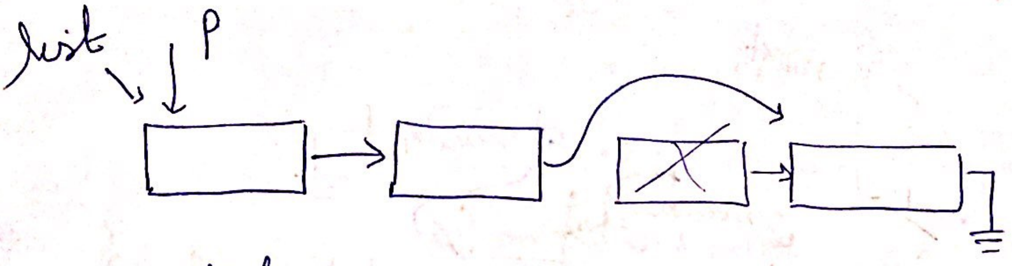
How to delete?

```
temp = p->next;
p->next = temp -> next;
free(temp);
```

(or you can say :- p->next = p->next->ne;)

function عكس الكان

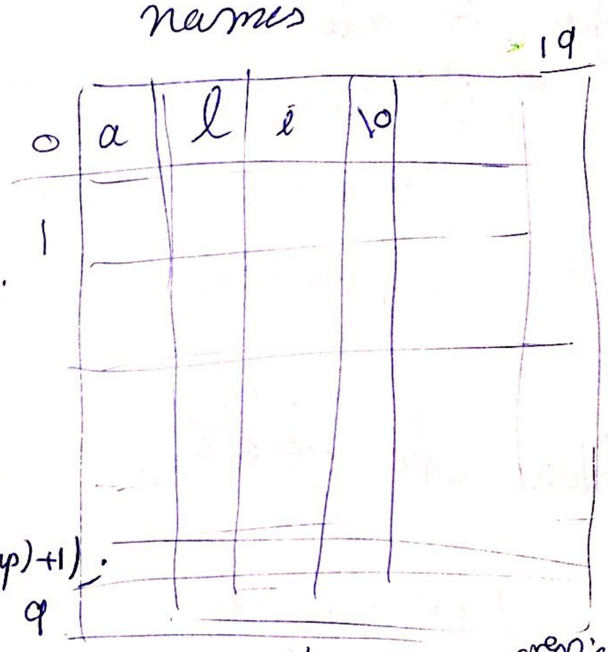
How to preint the list ?



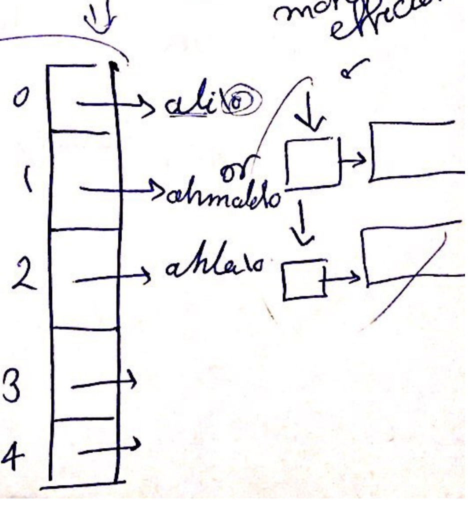
```
p = list;
while ( p != NULL)
{
    printf ("%s\n", p→name);
    printf ( "%d\n", p→id);
    p = p→next;
}
```

~.~.~.~.~.~.~.~.

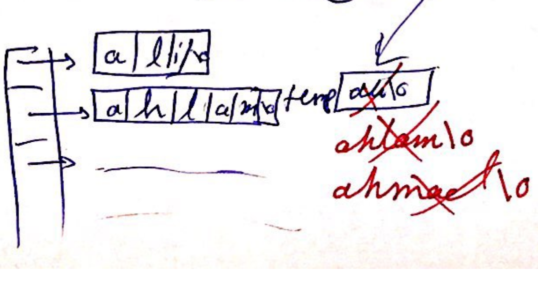
```
Char   temp [20];
Char   name [10][20];
char   * name [10];
for (i=0; i<10; i++)
  {
    Printf (" Enter name \n");
    Scanf (" %s", temp);
علّق
وحطّه →  names [i] = (Char *) malloc
          (sizeof (Char) * (strlen(temp)+1));
                         ضرب
    strcpy (names [i], tmp);
  }
```



more efficient



توضيح أكثر


a
l
i
\0
a
h
l
a
m
\0
temp
ali\0
ahlam\0
ahmad\0