برمجة كينونية

للطالبتان المبدعتان

فيحاء الحديدي، سلسبيل علي

إرادة - ثقة - تغيير

Review :-

1 data type and Variable 2 Selection 3 looping 4 Function

→ 1 datatype and Varible [Ex int x ;] [النوع + اسم المتغير]

→ 2 Selection → IF / Switch [T/F إذا كان عندك Condition وبدك تشيك عليه بـ]

* IF [الشرط داخل القوس]

[إذا ما داخل القوس (T) أو (none zero) بتنفذ الجملة يلي تحت، إذا ما داخل القوس (F) بتتجاهل (skip) يلي تحته وبتكمل]

عدد الجمل: إذا ماضي ؟ 1 ← (جملة وحدة) statement one
إذا مني ؟ أ ← عند كل ما داخله تتبع لـ IF وتنفذ الجمل

extension IF للـ ⇒ IF ( ) → إذا ما داخل القوس قيمة (T) أو (non zero) تنفذ الجمل، إذا (F) تنفذ جملة else
≡ else

extension IF/else للـ ⇒ IF ( ) / else IF ( ) / else IF ( ) / ⋮ / else → بتفصل الشروط أول ما تلاقي شرط (T) بتنفذه، إذا ولا شرط بتربط فصل بتروح على else والشرط يلي تبلاقيه بتنفذه + (skip) على الباقي (الكل)

* Switch (□) ؟

Case =: —— break;
Case =: —— break;
default;

→ بتشبه الـ [else if] القيمة الـ □ بتحاول تشوف بـ أي Case موجودة إذا لا بتروح على يلي بعدها، إلخ ...
[default] بتشبه الـ else لها ميزة، إذا ولا شرط تحققت (تُنفذ)
[break] بتفصل الـ Cases عن بعضهم

الممسوحة ضوئيا بـ CamScanner

→ 3) looping ← For, while, do

* For (P_1 و P_2 و P_3)
  - P_1 : تعريف Variable واحد أو أكثر
  - P_2 : Condition \ الشرط أو لأ لتكرار (بنطبّق على Variable)
  - P_3 : Counter التعديل على

Ex) For (int i ; i>0 ; i--)

[لو الـ Counter بين الـ parts مع بعضهم البعض) ج (i++) نزّل فيه لـ infinit loop !! لهيك لازم يكون في روابط]

إذا الـ Variable بدأ في القيمة العظمى ج يكون فيه decrement (--)
إذا الـ Variable بدأ في القيمة الصغرى ج يكون فيه increment (++)

كيفية تنفيذ جملة For :- أوّلاً الـ Compayer بنفذ ① ثم ② بعدين بنزّل حافر loop ③ [بنفذ بعدين بعدّل على Counter]

٥٥ ③ ، ② ، ① لأول لفة بعمل (initialization) بعدين بشيك على الشرط ثم بنزّل على loop اللفات القادمة بنفذ ⑤ ثم ⑥ ثم ⑦ [بعدّل على المتغير بشيك بعدين بنزّل على loop]

note] P_1 بتقدر تطلع برا For لأنه بتنفذ مرة وحدة فقط ولسيت جزء من التكرار [

* While (P_1) [precheck تسمى body + الشرط تحتوي على]
  - P_2
  - }

بشيك الشرط بالأول إنه (T) بعدين بنفذ ج إذا (F) ما بنفذ
P_1 : الشرط
P_2 : مجموعة من Statement (بعمل تكرار)

كيفية تنفيذ جملة While :- بوصل الـ Compayer عند While بحسب P_1 بطلع إما (T) أو قيمة أو (F)
أي قيمة غير zero هي (T) والـ zero هي (F) ج إذا طلع (F) أو (zero) ما بنزّل على P_2
إذا طلع (T) أو قيمة غير zero بنفذ P_2 ج ظل يكرر ويصير لحدما P_1 = zero ويطلع من While

سؤال ؟! تختار While أو For ؟! بعتمد حسب الشرط يلي عندك شو بتطلب منك

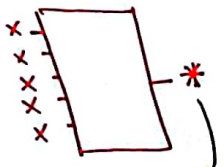
الممسوحة ضوئيا بـ CamScanner

* do → أول اشي بتنفذ كل الجمل داخل do وبس يخلص بتيجي الشرط
إذا (F) بطلع برا الـ loop [يلي تنفذت مرة وحدة على الأقل]
إذا (T) برجع بعمل تكرار

do
|
≡
|while

→ 4) Functions :- [(ليش تكتب / وين تكتب) Space/prototype]
[Block بيكون من Fun. الواحد
Code usability + modularity بتعمل support]

* لقبل أكتب الـ Fun قبل الـ main أو بعده.
الـ Fun لازم أحدد فيه
الاسم (name)
نوع المعلومات المعطاه
طبيعة الشي يلي بده يرجع



لازم ما يشابه name
Zero or more parameter داخل القوس

Result type | prototype name (typename , typename , typename)

name لـ Standard
يبدأ بـ (_) أو underscore أو كلمة أو حرف
ما يكون اسم محجوز مثل (array) + ما يغير الاسم تكرر مرتين
ما يكون فيه space

Ex) Sum (int x, Float y, char z) → X,y,z بطلب 3 قيم Sum لبس استدعي الـ

X= 30.5
y= 'A'
z= True

استخدم بـ ⇒
[Casting إذا القيمة يلي رح بعتها رح يعمل
لـ Fun أولاً وإذا لازم ازبطها إذا أمكن]

الـ Casting انه عم يعمل الـ Compayer في القيمة شان يخليها type لـ matching

X= 30
y= 65.0
z= Ascii (1)
return 50.5

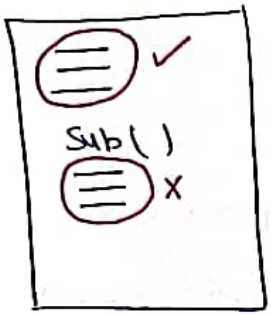
رح يستلم الـ 30.5 لـ int ولتغير X=30
الـ 'A' لـ Float ولتغير y=65.0
الـ True لـ Char ولتغير z= # في (1) لـ Ascii Code

بالنهاية الـ Fun استلم منك قيمة بس متل زي ما انت باعتهم و زي ما كتبهم في prototype
الـ 50.5 (الجواب) رح يعمله Casting حسب الـ (Result type)

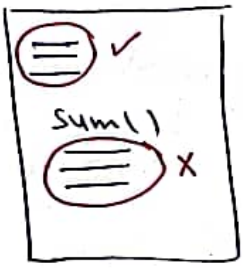
الممسوحة ضوئيا بـ CamScanner

Ex)

Sum



Sub



main ( ) ⟶ (البداية)

دائماً البداية من main [ما يطلع على اي اشي قبل main]
لانه مدخول لـ Fun استدعاء (Calling)
[وبداية البرنامج يبدأ من main وعشوفه]

Sum( )
X

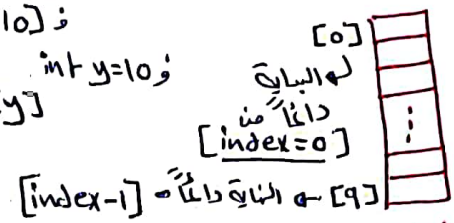
عمل (Calling) لـ Sum
يروح على Fun وينفذه
[يروح على Sum من Sum لـ Sub]

✓ الـ line يلي راح ينفذ عن الاكيد
X الـ line يلي ماراح ينفذ عن الاكيد

الممسوحة ضوئيا بـ CamScanner

5 Varibles and Arrays

- Varibles [int / Float / doubk / Char / bool / string]
  - لسيت من انواع (Data Built in) لازم اعملها include بداية البرنامج
- Arrays
  - int [10];
  - Const int y=10; int [y] ← لو حذفت (Const) ح يطلع error لانه حجم array لازم ثابت لانه يغير اثناء تنفيذ البرنامج



[10 = Size array على فرض انه]

Ex) char x ; = optinal (اختياري initial Value)
type — اسم الـ Varible

!? Array / Varible العرق بين

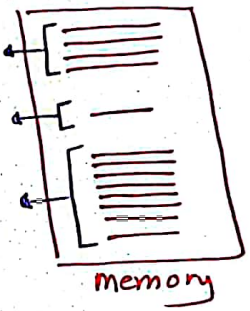
* Varibk : نوع معين + قيمة واحدة فقط
* Array : نوع معين + اكثر من قيمة

Ex) main ( ) ⇒ خطوات تنفيذ البرنامج
int x;
char y;
double z;

1. تشيك على (Check error) error [الكتابي]
2. حجز المتغيرات (Reservations for varible)
3. تنفيذ (Execute)

(الخطوة الثانية)
بحجز لكل واحد من المتغيرات في memory بالبايت بنفس الاسم اللي اخترته والمساحة اللي بحجزها بقدر على نوع Varible

x: 4 byte (int)
y: 1 byte (char)
z: 8 byte (double)



Ex) Array: double or int ar[ ][ ];
عدد الصفوف
عدد الاعمدة (لازم يتحدد)
[two dimensional array]

double ar [2] [4];
2 X 4 X 8 = 64

	①	②	③	④
①	8B	8B	8B	8B
②	8B	8B	8B	8B

لو معين لـ 4 الخانة 8 byte (double)

* حجم الـ array لازم نعرف مابنفع اسيب الحجم فاضي
ما دلالة من داخل القوس [ ] او من خلال { }

Ex) int ar [ ] = { 2,3,4,0 } ⇒ Size = 4
ح يوخد الحجم من داخل الـ ar [ ]
لانه ما جوى دلالة داخل الـ [ ] عن الحجم فالدلالة ح تكون بين { }

كل (4) مواقع / تم حجزها لانه int (4 byte)

4 X 4 = 16
array Size ← 4 byte (int)

الممسوحة ضوئيا بـ CamScanner

Ex) bool ar [ ][ ] = { {1,2,3}, {4} }

لـ يعتبر القوس الداخلي سطر واحد بصرف النظر
يحتوي على 3 من 3 أعمدة

one line

True (1)	True (2)	True (3)
True (4)	False (5)	False (6)

D

من خلال عملية الـ Casting
اي قيمة غير صفرية / تكون (T)

1 → T 0 → F
2 → T
3 → T
4 → T

bool لأنه

Ex) { { 1,2,3}, {4,5,6,7}, {0}, {1} }

T (1)	T (2)	T (3)	F
T (4)	T (5)	T (6)	T (7)
F (0)	F	F	F
T (1)	F	F	F

يرجع لي أكبر مجموعة في إذا بدي احطها في سطر واحد
كم بدها أعمدة؟! بصرف النظر عن الأعمدة

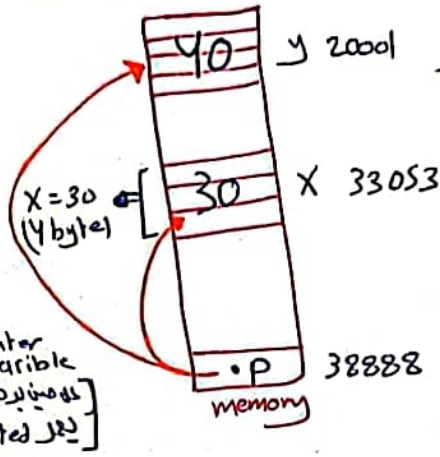
الممسوحة ضوئيا بـ CamScanner

6 Reference and pointers

Ex) int x=30;

لما بقدر من خلالها اعمل (x) لقيمة متخزنة mem. في

Reference pointers



Pointers [بديل اعمل متغير ليس قائم بذاته معمول عشان يعمل Pointers على x]

*(Pointers) كتابة: type [نوع المؤشر] *Pointer name = pointer Varible [اسم المؤشر] [هو متغير بيخزن فيه عنوان pointed يعمل]

[X *p= &X]
address

كيفية عمل الـ Comp. -> Search في mem. عن مكان فاضي ويخزن فيه

Ex) int y=40;
p= &y;
هو address

[لطيب العلم يلين بين P,X يعبر بين P,y مشوك]

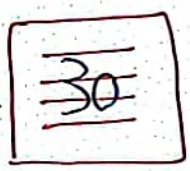
[من الجملة ؛ إنه أشر المؤشر p على Varible (X) ونوع المؤشر int يساوي int تعود على X [المؤشر (P) يعمل int data على Pointing]

[الاستفادة من Pointer ما بتجيبوا مع المتغيرات بالـ array] ← Note

* Reference [اسم تاني موجود من أول لكل موقع ماصه فيه اسمين]

int &Xyz=X; [لازم (X) يكون موجود من قبل]
يكون الاسم الأول هو X والتاني هو Xyz

X++ ≡ Xyz++ (Same location)



X = Xyz
[الموقع الواحد إله اسمين (two name)]

* لازم الاسم التاني نوعه مثل الاسم الأول

Char A[ ] = {'a','B','c','d'} ⇒ size type (التيجر صحيح)

#of location: 1000 1001 1002 1003

1 byte	1 byte	1 byte	1 byte
a	B	c	d

Char *pp= &A[2]; ⇒ [c هو الـ Pointer بدنا نأشر]

Array A من (3) (يعني الموقع الـ (3) فلي الـ Pointer بأشر على يلي بعده (P))

pp-=3; ⇒ يرجع (3) ويأشر على (a) لأنه كان عند (d)

* لو عملت (entry For loop) وال Pointer بأشر على أول (location) بكل مرة مرت الدور بستفيد (بشيد/بعدل/بعين/بقرأهم كلهم) الـ Pointer لفني عند (array) مهم

PP++ ⇒ يصير بأشر على يلي بعده (d)

① يتزود حسب النوع (الحجم)

② يزيد عدد الـ Address يتغير موقع التأشير

إعداد: فيحاء الحديدي، سلسبيل علي

الممسوحة ضوئيا بـ CamScanner

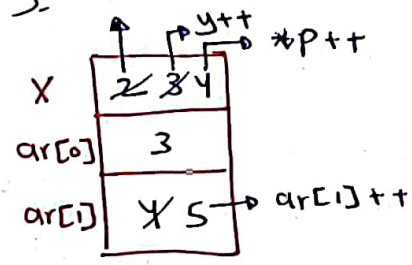
int Fun ((Value) int x , (Refance) int &y , (Refance) int *p , (Refance) int ar[])
int ar[2]
(الكالينام جاي في Fun)

- Varibl
- Reference
- Pointer
- Array

```
int x=2;
int &y=x;
int *p=&y;
int ar[2]={3,4};
```

X++; [x]
y++; [x]
*p++; [x]
ar[1]++; [x]

أدلوجف ماتغير



[التعديل أصبح على القيمة وليس على الأصل لعنى من 2 لـ 3 وكذلك]
[Calling by Value]

* Calling by

Value
[لستدعاء Fun وبعطاه قيمة]
متغير بس باسمه
لاتغير على المتغيرات الأصلية الخارجة

Refance
[لما ترجع لـ (main) الشغل يظل موجود]
*[] &
القيمة الأصلية تتعدل (نفس اللي بتأثر)

* كيف نعرف الصح داخل الـ Fun إذا Value أو Ref. ؟!
[[ ] أو (&) أو (*) من اشارة] ← Reference
[(int x) إذا المتغير بس من اسمه] ← Value

* الفرق بين *p++ و p++
*p++ ← بتغير القيمة زيادة القيمة بمقدار 1 (الـ * ارتبطها مع القيمة)
p++ ← بتغير الـ address (تغيير المؤشر للأمام)
اذا كانت p-- (تغيير المؤشر للخلف)

الممسوحة ضوئيا بـ CamScanner

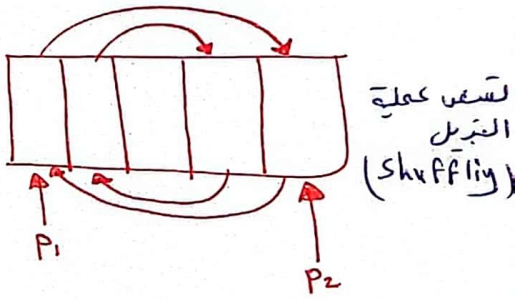
Ex) Write a c++ Function that uses two pointer to shuffle the elemnt of oddsize array.

task هي (التبديل)

Review: الـ Fun (شو بترجع، شو اسمها، شو بتاخد)

Sol

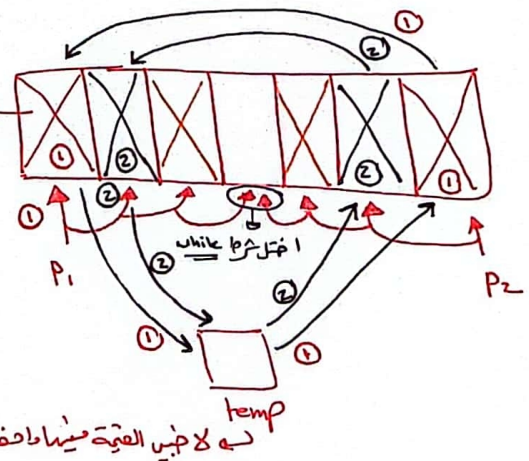
```
void Fun(int A[], int size)        // A[] بدل AR ، size بدل n
{
  if (size%2 == false)  ⇒ condition true (size even) ⇒ stop   (وقف ما تكمل اشي)
  exit (0);   ⇒ stop
  else
  {
    int *p1, *p2
    p1 = & A[0]
    p2 = & A[size-1]

    while (p1 != p2)  ⇒   // ما بتزبط لانه ما بعرف عدد الـ Counter للـ For
    {                     // (لازم نحط داخل while شرط التوقف)
                          // (في اول لفة T)
      int temp = *p1;
      *p1 = *p2;
      *p2 = temp;
      p1--;
      p2--;
    }
  }
```



لتشغل عملية التبديل (Shuffling)

```
main()
{
  int AR[9] = {  ,  ,  ,  ,  ,  ,  ,  ,  };
  Fun(AR[], 9)
}
```

*p1 (قيمة) او A[0] اسمها



بالحال ما عندي نتيجة (تحسب ترجع الجواب) ∴

ما ترجع اشي ∴ Fun باك void (no return)

Note الـ array إذا مررت حجمها فيه [ ] صار بيقرأ من الـ Comp. شو بدك بكتبه

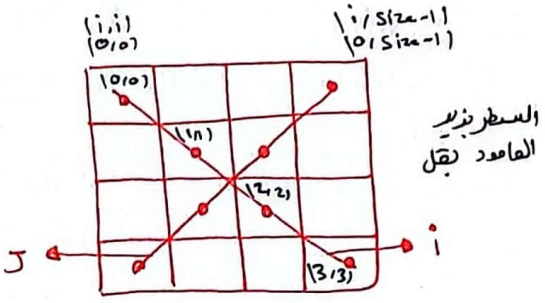
(AR[], 7) size جنبه size

الممسوحة ضوئيا بـ CamScanner

Ex) write a c++ Function that Calculates the Sum of all elemnt on the two diagonals for a two dimensional array of double.

```
double fun (double A[][#], int dim)
                              size
{
  int i=0, J=dim-1, sum=0;
  for ( ; J>0; )  ⇒ موجود Counter لانه For
  {
    sum += A[i][i];
    sum += A[i][J];
    i++;
    J--;
  }
  return Sum;
}
```

كمط اشي بـ For لانه كل اجراده
optimal الوسطانيين
لو تركته ح يصير
loop ∞ (لازم يكون
في شرط)



السطر بزيد
العامود بقل

لو افترضنا انه (two dim) مساويين بـ لانه العدد معروف مسبقاً
(them by them) = 4X4 / 6X6 / ...

note لبستنى جواب لانه طالب (sum)

```
main()
{
  double xyz[ ][#] = { ------ };
```

# of elemnt (9 or 16 or 25)

⚠ لـ ③ ⇒ Fun بكتبه بنفس الرقم [#]

Note بزبط الرتل (row dim) فاضي
أما (Coulm) dim لازم الحبيط

أول لفة
[1] A[ ][ ] A[ ][ ] (0) dim-1

تاني لفة
[2] A[ ][ ] A[ ][ ] (1) dim-2

Ex) what is the final Value of Sum?

```
int sum=0;
For(int J=0; J<100; J++)
{
  IF ( !(J % 17))
    Continue;
  else IF ( J>=80)
    break;
  else
    sum+=2;
}
```

Continue ← (بالعربي) ignore انه اتجاهل اشي تحته وارجع لـ loop وستوف
Break اذا خلصت هو وعملت كاملة (For loop) اطلع من

1%17 = 1 ∴ T ⇒ !(T) = F
∴ skip (تفزليلي تحتها)
1>=80 ?! NO ∴ sum = 2

80-5 = 75 م

!(zero%17) = !(zero) ⇒ not false T ∴ Continue
خارج اوجد الـ Sum (كمل)

J= 0 1 2 ... 17
X _ _ ... X
X: True (خارج اوجد الـ Sum)
_: False (ح اوجد الـ Sum)

note وجود J<80 زيادة else if(J>=80) break
لهيك الـ (100) طالما داعي عندي 80 لفة مبدأ (0)
خارج اوجد فيها عند الـ sum لهيك زيادتهم 75 بكل مرة
الـ sum+=2 (sum = 75X2 = 150)

Rang (0-79)

0 +17→ 17 +17→ 34 +17→ 51 +17→ 68
X X X X Y
① ② ③ ④ ⑤

skip = 5

الختصر عا 100 لبدال break صلاتهم (80) مال Continue صلاتهم (75)
(sum) 150 = 75X2 ∴ 2 كل مرة بزيد

إعداد: فيحاء الحديدي، سلسبيل علي

الممسوحة ضوئيا بـ CamScanner

Ex) Find the final value of X, y, *p, array, After executing the code.

```
Void Fun (int X, int &y, char *p, char R[])
{
  Switch (X+y)
  {
    Case 4:
     X+=5;
    Case 8:
     y+=5;
    Case 10:
     X+=5;
    default:
     y+=5;
  }
  p=R;

  For(int i=0; i<5; i++)
   if (i%2 == false)
    *R= *p+3;
   else if (i==3)
  * (R+1) = 's';
   else
   * (R+2) = ++(*p);
}
 main ( )
{
 int X=3, y=5;
 char z='a';
 char *p= &z;
 char AR[10]= {'A','B','C','D','E','F','H','I','J'};
 Fun (X, y, p, AR);
}
```

أولاً البداية من main دائماً بالتتبع المتغيرات وشوف الـ (initialization) عشان تكون عارف القيمة يلي رح توخذها من Fun (X, y, *p, R[])

كل متغير حسب موقعه والصيغة تاعته يابكون التعديل ← مؤقت (by value) ← دائم (by reference)

1 X: اي كعديل عليه رح يكون مؤقت (by value) ما انت بتبعت قيمة الشيء وليس الشيء نفسه

oo الـ X = 3 داخل الـ main رح تبعث الـ (3) وليس الـ (X) بالتالي الـ processing على الـ (3) أي ان بدون ما انفذ الـ Fun رح تكون X = 3 عشان (Calling by value)

2 y: بصفته (as Reference) أي ان (y) رح توخذ إسمًا وقيمة لـ Fun بالتالي اي تعديل بتعمله الـ (Fun) على (y) رح يكون حقيقي (real) على قيمة y

3 Pointer p: القيمة يلي باشر عليها (by Reference) اي ان (Pointer p) باشر على موقع (z) يلي منه جوه الـ (q) لما يروح الـ (Pointer) لـ (Fun) إذا انشال الـ Pointer ونحط له مكان تاني بتضل القيمة النهائية أو المكان يلي صار باشر عليه من جديد بتكون 'a' depend على الـ Fun نفسها

4 array: تنصبت (by Reference) بالتالي بتجعل الـ (array) مع مسمياتها والـ Fun بتصير كعدل لـ نفس (array)

oo ← الـ (X) موجودة منها نسخة في (main) ونسخة في (Fun) نسخة الـ (X) في (main) ليس لها علاقة بالنسخة يلي رح تروح لـ (Fun) أما بالمقابل الـ (main) و (Fun) بشوفو نفس (array + Pointer + y) لأنهم (by Reference).

Sol	X	y	*P	AR
	3			

y
*p
AR
يتبع الحل ...

الممسوحة ضوئيا بـ CamScanner

Fun(3, 5, 'a',) AR [A|B|...|I|J] R

3 10 15

* أول اشي X = 3 بتبدأ Calling لأنه (by Value)

* الـ switch تنفذ (بتوخذ المعادلة الموجودة داخلها X+y) بتشوف ايا واحد بتربط بـ Cases جميعا تنفذ Case بتطلع من الباقي ومن default إذا فيه break بين إذا ما فيه break عند match يلي صار وبتكمل للآخر من عند الـ match مشكلة لتبعزه

[match = تطابق الشرط]

8 = 3+5 = X+y

ح يدور على Case يلي منها (8) [ح يعمل SKip لـ Case4 لأنه ما بطابق المعادلة ح ينفذ Case8] + بعدك تشوف إذا فيه break أولاً مشان تعرف إذا بدك تكمل Execution أو لا [علما انه فاضي break ح اعمل (انفذ) الـ Case يلي صار عنده match ويلي بعده ← Case8 ← لسبب تحقيق المعادلة (match)

break ← ح ينفذوا بشرط عدم وجود [Case10, default]

y+=5 ← ح تتزيد لأنه Calling by Reference ح تزود عن الأصل y=10 بمعنات ≤

X+=5 ← X=8 / X=8 ما لهم علاقة في بعض (Calling by Value)

y+=5 ← y=15 بل كانت 10 يلي كانت ≤

Note لو بيزت الـ (y) للقسم (hum) ← Fun(X, hum, P, AR); ← التغيير hum ح تغير (y)

لأنه فيه Calling الـ Fun من main بتشوفه في الاسم يلي تعرفت عليه يلي هو y ≤ y 15 hum

اي ان (y) ح تزيد عن طريق الاسم (hum) [مشكلتهم المتغير يكون بنفس الاسم تحت التأثير بنعكس] القسمين لنفس المكان في main

* قبل (For loop) في P=R ← (Pointer = Array) الـ AR ح تتعامل معه زي كأنها R

الحلة صحيحة

AR [A|-----|J] R

AR ← main
R ← Fun
إلهما القسمين

AR[0] ← main ← القسم (A)
*AR

R[0] ← Fun ← داخل الـ
*R

حضانها P=R [let P Points to the same location as of R]

int x[2] = {1,2}

X [1|2]
X[0] X[1]

Cout << X[0] << *X

[Array = Pointer]

إذا عملت (2 Pointer) الـ [2] ح يأشروا على نفس المكان للمكان R لأنه بدي النقل قيمة R لـ P

* جمل IF/else ← [Single entry] هو مثل الـ switch الـ switch

إذا فاضي break وصارت match تنكل اما في IF/else اذا صار في match عند الـ IF ما تنكل حتى لو فاضي break ← T (ما ح تكمل)

اذا ولا وحدة match عند الـ else لا فيرة ← F (ح تنزل على else IF)

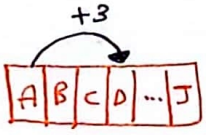
الممسوحة ضوئيا بـ CamScanner

* أول لفة في شرط IF [i=0 ⇐ zero = 0%2 ⇐ T] تحقق ∴ ماشي تدخل الباقيتين

[*R=*p+3] بأضداد Value تبقى p وازيدها 3 وادخل اوديها للمكان اللي بأشر عليها *R [Pointer R]

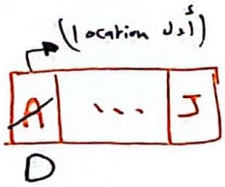
الآن الـ (Pointer R) والـ (Pointer p) بأشروا على نفس المكان نتيجة الجملة [*R=*p+3]

قبل تنفيذ الجملة الـ (Pointer p) كان بأشر على (A) بعد الجملة فكيت الارتباط تبعه مع (A) وصار على (D)



Note

The value of Pointer(R) = value of Pointer(P+3) ⇐ *R=*p+3



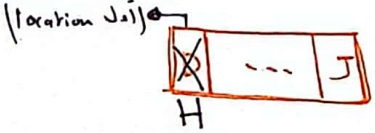
لما نرجع لـ main ح نشوف قيمة (R) بعد المسح [Calling by Reference]

D ← A تحولت من R ← (location) أول / Y=15 / X=3 لما نرجع لـ Fun الـ

[current loop] ماخل اشي في

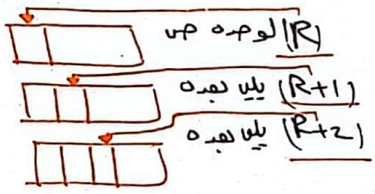
* ثاني لفة الـ i=2 ⇐ (Counter = 2) ⇐ T ⇐ zero ⇐ 2%2 IF]

العنوان اللي بأشر عليها (Pointer p) هو نفسها (Pointer R) نتيجة الجملة [P=R] اصبحت D زيها عاد H ⇐ 3



H = index أول location ∴ قيمة

* لفة رقم 3 ⇐ i=3 ⇐ F ⇐ notzero ⇐ 1 ⇐ 3%2 IF [نروح على (i==3) elseif

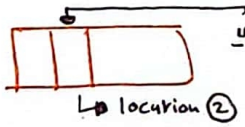


*(R+1) = 'S'

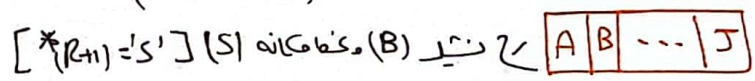
صاي الجملة ما بتصرلك الـ (Pointer)

[الجمل اللي تجعل movement (±) ⇐ مضاعفها الموقع الذي يبعدنا (Pointer) إلى الأمام أو الخلف [دون تحويل الـ [Pointer]

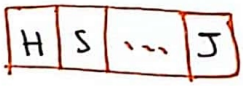
*(R+1) ⇐ الموقع الذي يبعدنا Pointer إلى الأمام بـ (one location) ← مسافة مجهدة اللي هي



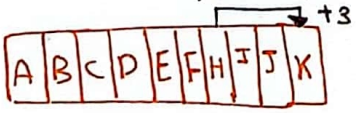
اللي منها 'B' ح نشيل (B) ومكانها [S] ['S'=*(R+1)]



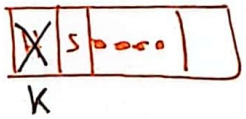
* لفة رقم 4 ⇐ T ⇐ zero ⇐ 0%2 IF [



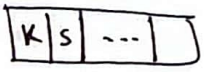
R = H+3 ∴ H الـ (P) والـ (H) ماتحركت لهذه عند *R = *P+3 الـ (R) لهذه



* آخر لفة 5 ⇐ F ⇐ 5<5 ⇐ 5=i) تطلع من loop



∴ آخر قيمة



الممسوحة ضوئيا بـ CamScanner

```
/*******************************************************
// INTRODUCTION TO CLASSES AND OBJECTS (ACCESS SPECIFIER)
*******************************************************/
      #include <iostream>
      using namespace std;

1.  class myclass {                    بداية ال class ب {
2.                    int a;
3.          public:
4.                    int b;
5.                    void setab(int i);
6.                    int geta();
7.                    void reset();
8.                  };                 نهاية ال class ب };

9.  void myclass::setab(int i)
10.       {
11.       a = i;
12.       b = i*i;
13.       }

14.       int myclass::geta()
15.       {
16.       return a;
17.       }

18.       void myclass::reset()
19.       {
20.       setab(0);
21.       }

22.       int main()
23.       {
24.       myclass ob;
25.       ob.setab(5);
26.       cout << "ob after setab(5): ";
27.       cout << ob.geta() << ' ';
28.       cout << ob.b;
29.       cout << '\n';
30.       ob.b = 20;
31.       cout << "ob after ob.b=20: ";
32.       cout << ob.geta() << ' ';
33.       cout << ob.b;
34.       cout << '\n';
35.       ob.reset();
36.       cout << "ob after ob.reset(): ";
37.       cout << ob.geta() << ' ';
38.       cout << ob.b;
39.       cout << '\n';

40.       return 0;
41.       }
```

module 1
أسم ال class

block
معرف
fun setab
(تابع لل class)

block
معرف
fun geta
(تابع لل class)

الممسوحة ضوئيا بـ CamScanner

Set 1 : 1

* C++ تسمى : [Varible based programmiy]
* data structures تسمى : [object based programmiy]
  - تعتمد على مبدأين
    - modularity
    - Code usability [إعادة استخدام الكود]
* الـ class عبارة عن [modularity] لازم تكون بها الـ main لأن مزايا صاد [modularity]
  [من line 1 الى line 8] عبارة عن module بداية [{] وبالنهاية [}] مع [;]
  عبارة عن [block] واحد داخل [;] بتحجبي فيا أكثر السطور شي أكثر الـ main

→ [lin 1 → line 8] : عبارة عن [module 1] اسمه [myclass] نوعه class

* في [object based programmiy] ما نقدر نحط class عن طريق [module]
  - [data members] Varibles
  - [member Function] Functions
  - تتقدر تحط داخله

→ [line 2+line 4] : data member
→ [line 5+line 6+line 7] : member Fun.

* Access Specifier : محرّف الوصول تحوي على نوعين
  - private
  - public

→ [line 2] : private / public ← وهكذا يقدر حسب الـ main قديش الـ [9]

Ex: Cout<<ob.a; ← الجملة صحيحة إذا public وتكون خاطئة إذا private والـ [9] خاص بكلاس لا يوجد
خطيها public اي انه الـ main ما يقدر يشوفها ∴ [X] خاطئة الجملة

Ex: Cout<<ob.b; ← الجملة قبلها public يقدر الـ main يشوفها ∴ [✓] صحيحة الجملة

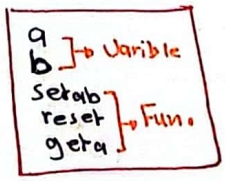
* تأثير الـ Public [بعض تأثيرها ما شي للدرجة إذا ما ميزتها متل الـ switch بدون break]

→ [line 5] : بترجع [int] وبتوخذ [void]
→ [line 6] : بترجع [int] وما بتوخذ اشي
→ [line 7] : ما بتوخذ اشي ولا بترجع اشي [void]

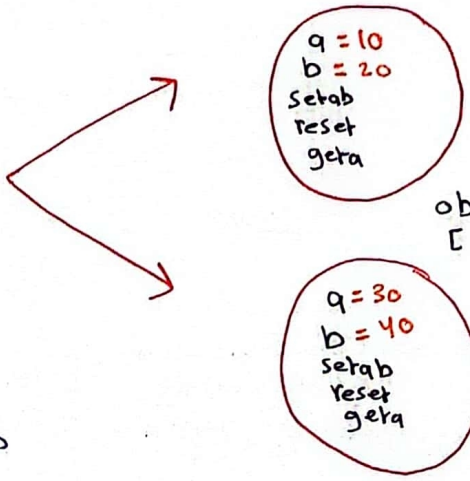
→ [line 9] : initilization عبارة عن [::] رمز الـ
Calling عبارة عن [ob.□] ← [o] أما رمز الـ

الممسوحة ضوئيا بـ CamScanner

* الـ class من ضمن الكلمات المحجوزة في [object oriented]



my class [الـ module كامل]



[Coutcca;] ← لو كتبنا
بيطلع error !

السبب [عندي نسختين لـ a] 10 / 30
لازم اسبق الـ [a] بـ اي [a] بالضبط ؟!
مثل [line 25]

الـ a/b طالما متيم فلازم نعمل module
طالما الـ module بنعمل منه نسخ [فمحتاجين]
عشان نتكلم فكرة
[Code resuability]

* تقسيم الكود الى blocks
1. تقليل dependability في الكود
2. تسهيل من اكتشاف ومعالجة الـ error
3. تسهيل الفهم وتسهيل تتبع الكود

* الفرق بين [module 1] و [module]
← إذا عملت initialization للـ (a) الموجودة في (1) طارح تاثر على الـ (a) الموجودة في (2).

← تسمى النسخ بـ [object]

* الـ class يكتب لمرة واحدة وكل ما استنسخ منها بتتج عندي [object]

Ex T OR F

← الـ class بتحجز إلها مكان في mem. ؟!

Sol (F) لانه الـ class مجهولة لا سحب منها object والمبيه لما المبيه بجيبه متيم بجيب object موجود، إذا ما سحبت منها نسخة ما بجيبي Fun. يلي ما بتبدعيه.
ولا الـ class ما بتوخذ مكان في memory الا الـ object بتوخذ مكان في memory.

إذا ما عندي Fun في جوا الـ lines ما بقدر استدل على Varible في line 2 / line 4 → [line 5 + line 6 + line 7]

* الـ class المثالية تحتوي على [Varibles] multiples data member
multiple Functions

→ [line 24] : [Fun. الـ size + جميع التغيرات .size = size object] memory يحجز في
الموجودة داخل class

* لو حذفنا [lin 5] ⇐ [line 9 ← line 13] error يكون
* وظيفة الـ Function member [تعديل على الـ Varible المحجوب من برا]

الممسوحة ضوئيا بـ CamScanner

* متى بدي اعمد إذا بدي C++ أو object ؟!
← إذا قدرت توصف الـ Varible حتى وحدة اشتغل C++ [Varible based]
إذا الشيء يلي بدي البرمجه إله أكثر من صفة اشتغل object [Class]

Ex Write a program student ؟! الـ student لأبرمجه بدها أكثر من Varible
مثلاً [التخصص، السنة، الرقم الجامعي، العمر، إلخ ...]

* يوجد طريقتين لبرمجة الـ classes ثم [geta , setab]
← لحتى يكون الـ class بعد الـ main لازم اكل class لـ Prototype
note من [line 1] لـ [line 21] خارج اتفرج كليهم لهنا نهائياً كمبتز إذا الـ main سمحلك فيه
البداية دائماً من main [line 24]

class الـ Code Fun. احطه داخل الـ class
بعد اسمه مباشرة.
او اكل المتبع داخل البرنامج من الكود بتعنا
[line 9 → line 13 + line 14 → line 17]
بقدر اعملهم CPP واطلعهم الـ class تحت كل اسم خاص فيه.

line 9 : void myclass :: setab (int i)
الكائن :: الكائن
(اسم) Contents :: element (اسم)
(اسم الـ class التابعة لها) (اسم المتغير او Fun.)

⇒ * الـ [::] بتوقعهم خارج main
* الـ [.] بتكون داخل الـ main
كل صفة اسم النسخة

[line 24] : [creat object] ob استنسخ من class نسخة اسمها
من الـ class يلي اسمه myclass

note
main بيشافو من كل مكان، يحق لك استخدامه في
int b
setab
geta
reset

أما الـ [a] ← لا يحق لك استخدامه في main، لانه private شو الحل ؟!
يوجد Fun. يسمى geta وهو [Public] وسيط لـ (a) لانها private
وبتان اقدر اوصلها من main.

* [الـ Fun بالغالب بتكون Public والـ Varible بالغالب بتكون Private]
* الفرق بين الـ Public ← انه علاقة برؤية القيمة
والـ Constant ← انه علاقة بثبات القيمة

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لتطلع الـ [output] ooo

[1] [ ] ⇐ [line 24] تطلب من [object] class اسمه [myclass] سميته ob

[2] [ ] ⇐ [line 25] بدي العبة الـ [ob] object لـ [setab] Function ومرر له (5)

رحت لـ Fun. ⇐ [line 9] دخلت عل Fun. تجوه من بدل كل (i) بـ 5 ⇐ i=9 ⇐ i*i=b
9=5
b=25

لهم ها برجع لـ [main] وبحفظ بالقيم داخل الـ [ob] object

[3] [ ] ⇐ [line 26] جملة طباعة cout وبكتب الجملة كما هي بين [" "]

⟶ ob after setab(5):

[4] [ ] ⇐ [line 27] جملة طباعة cout بيعت لـ [ob] لـ Function [geta] ⇐ [line 14]
الـ Fun برجع قيمة (9) هي [ob] object وآخر قيمة لـ (9) تخزنت هي (5) لهم ها space اعتبار [' ']

[5] [ ] ⇐ [line 28] جملة طباعة cout واستدعيت الـ (b) الموجود هي [ob] object
آخر قيمة لـ [b] تخزنت هي [ob] هي (25)

[6] [ ] ⇐ [line 29] جملة طباعة تحوي على [\n] معناها [newline] مثل endl

[7] [ ] ⇐ [line 30] غير القيمة (b) هي [ob] object لـ 20

[8] [ ] ⇐ [line 31] جملة طباعة تظهر كما هي وكاتبة بين [" "]

⟶ ob after ob.b = 20:

[9] [ ] ⇐ [line 32] العبة [ob] object لـ [geta] Fun. ⇐ [line 14] برجع قيمة 9 (ما تغيرت) ⇐ (5)

[10] [ ] ⇐ [line 33] جملة طباعة cout واستدعاء الـ [b] من [ob] object وآخر تعديل عليها هي (20)

[11] [ ] ⇐ [\n] ⇐ [line 34] newline

[12] [ ] ⇐ [line 35] العبة الـ [ob] object لـ [reset] Fun ⇐ [line 18] بداخله setab (0) برجع لـ
[line 9] ⇐ [setab] Fun خوه من بدل كل (i) بـ 0 ⇐ i=9 ⇐ i*i=b
9=0
b=0
وخزن آخر تعديل هي object [ob]

[13] [ ] ⇐ [line 36] جملة طباعة cout وتظهر كما هي وكاتبة بين [" "]

⟶ ob after ob.reset

[14] [ ] ⇐ [line 37] جملة طباعة cout العبة الـ [ob] object لـ [geta] Fun. ⇐ [line 14] برجع
قيمة الـ (9) وآخر قيمة تعدلت هي [0]

[15] [ ] ⇐ [line 38] جملة طباعة cout واستدعيت الـ (b) من [ob] object وآخر تعديل تعدل عليها هي [0]

[16] [ ] ⇐ [\n] ⇐ [line 39] newline

الممسوحة ضوئيا بـ CamScanner

الممسوحة ضوئيا بـ CamScanner

```
/*************************
// CONSTRUCTORS/DESTRUCTORS
*************************/
#include <iostream>
#include <cmath>
using namespace std;

class point {
float x;
float y;
public:
point():x(1),y(1) // non paramerterized constructor
{
      cout<<"non paramerterized constructor initialized x and y to 1"<<endl;
      }

      point(float a,float b):x(a),y(b) // paramerterized constructor
      {
      cout<<"paramerterized constructor initialized x to "<<a<<" and y to:
   "<<b<<endl;
      }

      float distance_origin();

      void show_xy()
      {
      cout<<"The value of x is : "<<x<<"the value of y is :"<<y<<endl;
      }

      };

      float point::distance_origin()
      {
      return sqrt(pow(x,2)+pow(y,2));
      }

      int main()
      {

      point p1;

      point p2(4,10);

      point p3 = point(-2,-6);

      cout<<"The distance between p1 and the origin i:s "<<
   p1.distance_origin()<<"\n";

      cout<<"The distance between p2 and the origin i:s "<<
   p2.distance_origin()<<"\n";

      cout<<"The distance between p3 and the origin i:s "<<
   p3.distance_origin()<<"\n";


      return 0;
}
```

بداية ال class

نهاية ال class

الممسوحة ضوئيا بـ CamScanner

Set 1 : 2

* بالرجوع للكود السابق متى [a , b] أول ماتقل الـ main تكون [garbage] لو البرنامج بدأ ينفذ من المتى وشاي يقل [initialization] ح يعطي error ! كل اشي بنفذه لحتى اتجنب هذه النقطة لستقل [Constructor]

← [Constructor] :- عبارة عن Fun اسمه مثل اسم الـ Class ، يُستدعى تلقائياً كما يفعل obj لازم نيكتب قصت كلمة [public] إذا فوق كلمة Public ح يعتبرلك ! error في main انا ما بفعله استدعاء هو ليشتغل لوحده لما اعمل [object]

- بالرجوع للكود السابق [لازم في line 24 يقل Constructor ، عشان ماتسمح للبرنامج ما يكون عنده متى]

* الـ Constructor اسمه مثل اسم الـ Class ما برجع شيء لكن يوخذ مجموعة من [Variables]
* لما تقل Search في Class زيدش الاسم تكرر اكثر من مرة حاد يكون [Constructor]
- الـ module في هاد البرنامج حار اسمه Point

→ [line 8] : point ( ) ⇒ [Constructor عن Class اسمها مثل اسم الـ] لا يرجع ولا يوخذ اشي ، اسمها مثل اسم الـ Class

- الـ Constructor ممكن يوخذ متى ، ممكن ما يوخذ متى .

→ [line 12] : point ( ) ⇒ [Constructor] تابع

* في حال وجود متغير بـ نفس الاسم مرتين ح يسبب مشاكل ! إلا إذا ميزناهم عن بعضهم [Function overloding]

line 8 ← point ( ) ← ما ح امرر لإلها متى [لما نستدعيه ما برر لاله متى] ← Ex
line 12 ← point (#, #) ← ح امرر لإلها متى [لما نستدعيه برر لاله متى] ← Ex
line 28
line 29

Note : ما يقدر اعمل Point مع (3) متى ح تعطي ! error

→ [line 8] : : x(1) ,y(1) ⟶ معناهم [لازم يكونو :] x=1 , y=1

← لما تعمل [Constructor] يوجد قدامك طريقين إما بدون متى او مع متغيرين فقط

← الفرق بين line 16 و line 17 ← ما في داعي اكتب الاسم كامل لانه داخل الـ Class الـ body بعده .
↳ طلعت الـ body برا الـ Class لازم اطلعه واستدعيه بدك تكتب الاسم كامل مثل line 22

[Point :: distance-origin] لانه الـ Class انتهى عند 21

اسمها :: (Fun) اسم الـ Class

الممسوحة ضوئيا بـ CamScanner

- [line 28]: Constructor line 8 ← ما مررت قيم هه لمبتدئي ← [انشاء obj ليسمى p1 من class ليسمى Point]
- [line 29]: Constructor line 12 ← مررت قيمتين هه لمبتدئي ← [انشاء obj ليسمى p2 من class ليسمى Point]

* لو بدت [line 38] حذفت عبارة [cout<< x ;] ما بتعرف لأنه ما صدرت أي x

لأنه لدي الـ x [3 انواع] p1 p2 p3 ← شو الحل ؟! ما بنربط أكتب p1.x p2.x p3.x أو بدل الـ x احط y

لشي ؟! لأنه الـ x والـ y مبزار public الـ main ما بتعرف عليها الحل الصحيح أكتب اي اشي موجود في public مثل [distance origin] or [showXy]

- [line 30]: Constructor line 12 ← مررت قيمتين هه لمبتدئي ← [انشاء obj ليسمى p3 من class ليسمى Point]

* Note ← p2(4,10) ← في الخيارات دير بالك على نوع المتغير [9 ح تروح على x والـ x Float] x=9 b=y

لدي مشكلة 4.5 ح اخذ 4.5 لو الـ x كانت int ح يطلع y [line14]

* لو حذفت [line12 + line8] ← line 28: ينفذ لأنه لمبتدئي الـ default صمن [non parameterized Constructor] default الـ

line 29: ح يعطي error !

Destructors ← Function هي نفس اسم الـ Class مالها return data ويجب ان يسبقه ~ destructor حتى يكون operator type [لتتخذ للهدم]

* كم destructor لقيت اعمل ؟! واحد فقط لأنه وظيفته هدف

- يصير Calling للـ destructor عند انتهاء الـ main

الـ destructor [آخر اشي انشئ من الـ obj هو آخر شي يتم تحطيمه]

الـ destructor والـ Constructor [useful tools إضافات ملزمين في استخدامهم]

الممسوحة ضوئيا بـ CamScanner

* الكود لتوضيح أهمية الـ Constructor

```
1  class test
2  {
3      public:
4          int x;
5          test();
6      void main()
7      {
8          test object;
9          cout << object.x;
10     }
```

← الـ module الأول
test [عبارة عن class تحوي على متغير واحد x]
← الـ class ما إلها أي module إلا لما تعمل لإلها object
← الـ module الثاني
test object [لما اذكر اسم الـ class متبوع باسم ثاني ؛ كأنه بحكي للـ Complier اعمل object المعلومات تالته مسحوبة من class ليس test]
* الـ object كلمة غير محجوزة لكن الـ class كلمة محجوزة

* لو نفذنا جملة 9 شو رح يصير؟!
الـ (x) قيمته private مش مسموح لنصل من main فالحل بنحط مكان كلمة public line 3
[بسبب public خلت اي module تزال class يقدر يشوف الـ x]
- بعد وضع كلمة public شو رح يطلع الـ output ؟ تحولت الجملة من Syntax error الى بنفع تطلع
- شو رح يطلع ؟ رح نقطين error ! لنوعه [Unitialization Varible] ؛ لأنه ما في [Constructor محطوط Fun] داخل الـ class اسمه زي اسم الـ class ما برجع اشي ولا حتى initilization الـ [Varible
لحل المشكلة نضيف [Constructor] مثل [line 5] (test();) عملتا لا عمل (initialization)

لو وحطت مثل هيك بدون قيم ؛ ما حققت الغاية من Constructor لأنه الغاية اعمل [initialization] للقيم

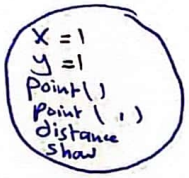
- يلي بتغير بتحول من [Syntax error] إلى [Unitialization Varible] ← garabage [logical error]
* مشان اتخلص من (Syntax والـ logical) error نضيف [test(){x=30;}] الـ x=30 حل لكل المشاكل ؛ لكن هل بتحقق الهدف من Constructor !? لأ ؛ لأنه بكل مرة بدي استنسخ رح تكون قيمة x=30
يلي بدي اياه بدي اسحب obj ① واعطيه قيمة و obj ② واعطيه قيمة تانية مختلفة عن obj ①
← لذلك بظل الصفة [test(int z){x=z;}] بترتب عليه انه اغير line 8

test object (30)
← امرره قيمة = [test(int 30){x=30;}]
اكل لا مثل بكون عندي [Constructor] بتغير ؛ واخير بصير مع كل جملة انشاء ايزا قيمته = القيمة مباشرة تنعكس على Constructor

test object(40)
[test(int 40){x=40;}]

الممسوحة ضوئيا بـ CamScanner

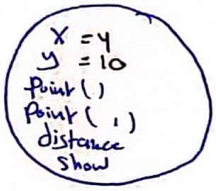
ooo تتبع الكود لنطلع [output] ooo



P_1

[1] البداية من [main] ⇐ [line 28] انشئ obj اسمه P_1 من Class اسمه [Point]
لكن ما مررت قيم له ⇐ استدعي الـ [non parameter Constructor] ⇐ [line 8]
اعطي قيم لـ x و y ← 1 ⇐ واطبع اول جملة طباعة لانه تالية لـ [non parameter Constructor]

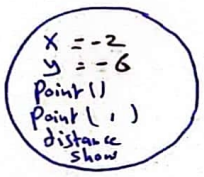
→ non parameterized Constructor initialized x and y to 1



P_2

[2] [line 29] ⇐ انشئ obj اسمه P_2 من Class اسمه [Point] ولكن مررت له قيم
⇐ استدعي الـ [parameter Constructor] ⇐ [line 12]
اعطي الـ x لـ a و الـ y لـ b والتنين [Float] واطبع جملة الطباعة التالية لانها

→ Parameterized Constructor initialized x to 4 and y to 10



P_3

[3] [line 30] ⇐ انشئ obj ماله اسم من Class اسمه [Point (-2, -6)] Point
وبعدها انقله لـ obj اسمه P_3 من Class اسمه Point ولانه مررت قيم
⇐ استدعي [Parameter Constructor] ⇐ [line 12]
اعطي الـ x لـ -2 والـ y لـ b والتنين [Float] واطبع الجملة التالية لانها

→ Parameterized Constructor initialized x to -2 and y to -6

[4] [line 31] ⇐ اطبع الجملة وبعدها اطبع [P_1] بحتواية ومبتعه لـ Fun [line 22] ⇐ [distance-origin]
الـ Fun عبارة عن معادلة رياضية $\sqrt{(x)^2 + (y)^2}$
$\sqrt{2} = \sqrt{(1)^2 + (1)^2}$ ⇐ Sqrt (pow(x,2) + pow(y,2))

$\sqrt{2} = 1.4$

→ The distance between P_1 and the origin i : is 1.41421

[5] [line 32] نفس [line 31] لكن الـ obj [P_2] بحتواية ومبتعه لـ Fun [distance origin]

$\sqrt{116} = \sqrt{(4)^2 + (10)^2} = 10.7703$

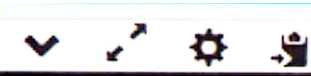
→ The distance between P_2 and the origin i : is 10.7703

[6] [line 33] نفس [line 31] لكن الـ obj [P_3] بحتواية ومبتعه لـ Fun [distance origin]

→ The distance between P_3 and the origin i : is 6.32456

Note ⇐ دائماً عند انشاء الـ obj لستدعي الـ Constructor و اوعى تنسى جهاز النقطة
لانه قلبها والكود اذا لقيت ترجعله ⇐ تفقر اذا جعل طباعة خاصة بـ Constructor

الممسوحة ضوئيا بـ CamScanner



```
non paramerterized constructor initialized x and y to 1
paramerterized constructor initialized x to 4 and y to: 10
paramerterized constructor initialized x to -2 and y to: -6
The distance between p1 and the origin i:s 1.41421
The distance between p2 and the origin i:s 10.7703
The distance between p3 and the origin i:s 6.32456


...Program finished with exit code 0
Press ENTER to exit console.
```

الممسوحة ضوئيا بـ CamScanner

```
/************************************************************
// OVERLOADING CONSTRUCTOR AND MULTIPLE OBJECT INSTANTIATION
************************************************************/
1. #include <iostream>
2. #include <cmath>
3. using namespace std;

4. class point {
5. float x;
6. float y;
7. public:
8. point():x(1),y(1) // non paramerterized constructor
9. {
10.      cout<<"non paramerterized constructor initialized x to 1 and y to 1 "<<endl;
11.      }
12.      point(float a,float b):x(a),y(b) // paramerterized constructor
13.      {
14.      cout<<"paramerterized constructor initialized x to "<<a<<" and y to:
   "<<b<<endl;
15.      }
16.      float distance_origin();
17.      void show_xy()
18.      {
19.      cout<<"The value of x is : "<<x<<"the value of y is :"<<y<<endl;
20.      }

21.      ~point()
22.      {
23.      cout<<"the point (onject) with x = "<<x<<"and y = "<<y<<"is exiting"<<endl;
24.      }

25.      };
26.      float point::distance_origin()
27.      {
28.      return sqrt(pow(x,2)+pow(y,2));
29.      }


30.      point p1;

31.      void test_function()
32.      {
33.      point p2(4,10);
34.      }
35.      int main()
36.      {

37.      point p3 = point(-2,-6);

38.      test_function();


39.      return 0;
40.      }
```


الممسوحة ضوئيا بـ CamScanner

Set 1 : 3

* Overloading في C++ الـ [Fun. overloading] يعني تشابه ، تشابه [Constructor] يعني نقدر أعمل Copy الـ [Constructor] بشرط يختلفوا بشيء ما .

Ex [line 8 + line 12] : Fun. متشابه في الاسم لكن تختلف في شيء ما ، حتى Complier يميزهم ليستعملهم .

الفرق بين [line 8 / line 12] : واحد منهم يتم والتاني ما لمنه يتم ، امامن ناحية الاسم فكلهما نفس الاسم من ناحية الـ [Return data type] الـ Constructor ما يرجع شيء بالتالي ما نقدر اميزهم من الشيء يليا يرجعوه ، بميزهم من [Parameter list] .

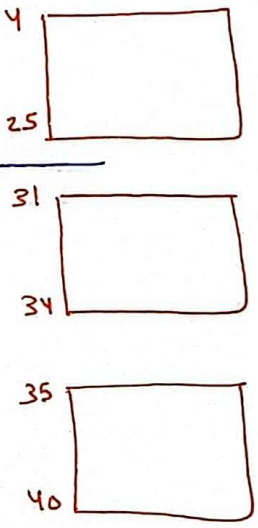
← الـ destructor : ما نقدر اعمل أكثر من [destructor] واسمه قبل اسم الـ Class و ما يرجع [data type]
حتى فقدت الوسيلة إنه اعمل [Destructor] اميزهم عن بعض
⚠ دائماً لنسخة واحدة إن وجدت الـ destructor
كيف بميزه ؟! من العلامة [~] EX [line 21] ~Point()

* وقت الإنشاء ما بنهتم فيز إذا كان في [destructor]
* دائماً وأبداً اعملت obj دور على [Constructor]
* بدل تصفن obj دور على [destructor]
* ترتيب الإنشاء 1,2,3 ترتيب الحذف [destructor] بالعكس 3,2,1 [يلي عرفته آخر شيء رح ينحذف أول شيء]
الـ Global لازم يكون موجود لأنه اول من انشئ من الـ obj حتى آخر من يُحذف هو]
← 1 Global 2 main 3 داخل الـ main [Functions] و ترتيب الإنشاء .
← لو حذفت [line 10 + x(1),y(1)] رح تعرف دائرة الـ Constructor بدون ما يتم يتصل [garbage إلى Syntax error]
[الفائدة انه اعمل initilization للـ x و y]
← لو حذفنا [line 8 + line 9] رح تعبر [Syntax error : Uninitilization Varible]
← لو حذفت [line 8] [Constructor] لازم كل الـ obj يتعملو صيغة [line 12] و لو كان Constructor موجود وعملت obj بدون ما يتم رح يعطيني error !
EX Point P1; [line 21] ⇐ * الـ destructor اسمه نفس اسم الـ Class + اسم الـ Constructor عشان اميزه انه destructor بحط (~) و لتبتدي لما اعمل الـ Constructor للـ object و نهاية عمر الـ object لتبتدي الـ destructor .
* الـ destructor خيار [optional] و لأنه الـ Complier بتكفل انه كل شيء ينحذف في memory
* الـ Constructor إجباري تنضاف و لا تجيب الـ error !
* عدد الـ destructor = عدد الـ object يلي بدي اسجبهم
* الـ destructor بيبقى بعد الـ object او Constructor يلي بدي ينشال

الممسوحة ضوئيا بـ CamScanner

* من line4 لـ line25 [Block class] ⇐ بطلع عليه بس اعمل object
* من line26 لـ line29 [جزء من class لكنها مكتوبة برّا ما بحط فيها box لكن هي فعلياً تتبع الـ class الدليل Point [الاسم جوّا لكن التعريف برّا]
* line30 ⇐ [Global object]
* من line31 لـ line34 [Block Function] ⇐ بطلع عليه إذا استدعيته من main
* من line35 لـ line40 [Block main]

* Blocks لتقسيمهم لـ مشان اعرف مين وين انفذ من object + main معرف خارج main ليس [Global object] مش تابع لأي Block



(وقت التنفيذ)
[main بداية من line30 (main قبل) ، نهاية بعد main] Point P1;

* إذا أنا بـ برنامج لها اجي يبدأ انفذه بطلع على البرنامج من بدايته لنهايته ، اذا مش [Global] بنفذه
(برسم دائرة وبجيب الصيغ) بعين بروح بس main بنفذه.

* لو مسكت line30 وحطيته قبل line4 تحت بصير line3 بصير error! لانه الـ Complier ما بعرف انه بصير من class لذلك إذا بدي اعرفه بحطه تحت line4 [تحت الـ class]

* الـ Global بنفذه حسب موقعه
قبل ما بجي الـ main بروح بنشاء من memory آخرهم ؛ لانه كتلة two block [main + Function test]
[او بنشاء قبل الـ main هو بروح بنضرف بعد main]

* ما بقدر حاطط الـ class اكتب Point خطأ ⚠
لانه لسه ما خلصت الـ class لاعمل object الـ Complier بستنى الـ [؛] مشان يعرف مواصفات الـ class كاملة ويعرفها يصير له [object] وينسخ على

توقيت انشاء [Global] P1 [1]
main نفذ [2] [Constructor] P2 ثم P3

Note
P2 مع تكرار الـ [؛] لازم يروح [line34 بدايته من line33 نهاية بـ

[أول واحد انشئ آخر واحد يتضرر]
توقيت الحذف P2 [1]
P3 [2] [Destructor]
P1 [3]

الممسوحة ضوئيا بـ CamScanner

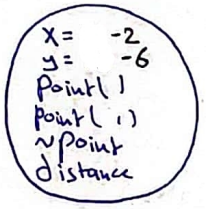
ooo تتبع الكود لنطلع [output] ooo

1 [line 30] ⇐ [Global obj] ← انفذه أول واحد ← نستدعي الـ [non-param. Constructor] ⇐ [line 8]
يحطي قيم لـ x, y (1) ونطلع اول جملة طباعة الخاصة فيا Constructor

P1

→ Non paramerterized Constructor intitialized X to 1 and y to 1

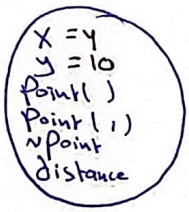
2 نروح عند main لنفذ الكود [line 37] انشئ obj اسمه [P3] من class اسمه [Point] ← نستدعي [Parameter Constructor] لأنه مررت قيم ⇐ [line 12]
← نحطي x لـ (9) و y لـ (4) ونطبع جملة الطباعة الخاصة منه



P3

→ Paramerterized Constructor initialized X to -2 and y to -6

3 [line 38] ⇐ استدعينا ← Fun [test] ← نروح عليه وانفذه ⇐ [line 31]
الـ Fun. يطلب منه انفذ object اسمه P2 من class اسمه [Point]
مررت قيم ⇐ [Parameter Constructor] ⇐ [line 12]
← اعطي x لـ (9) و y لـ (6) ونطبع جملة الطباعة الخاصة منه



P2

→ Paramerterized Constructor initialized X to 4 and y to 10

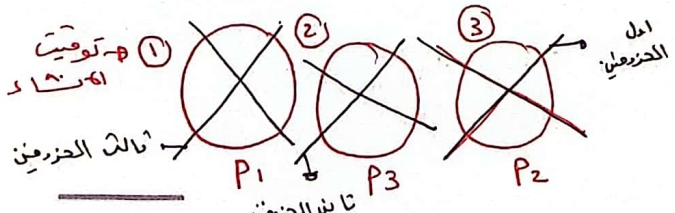
← اول ما اخلص من Fun بحذف P2 ⇐ [line 23] ⇐ نستدعي [destructor] كصوي كل جملة طباعة

→ The value of X is : 4 the value of y is : 10

← خلصت تنفيذ البرنامج وضل عندنا P1, P3 مبدأنا بـ P1 هو آخر واحد بحذفه ونروح احذف P3 ثم P1
استدعى الـ destructor للـ P3 ثم لـ P1

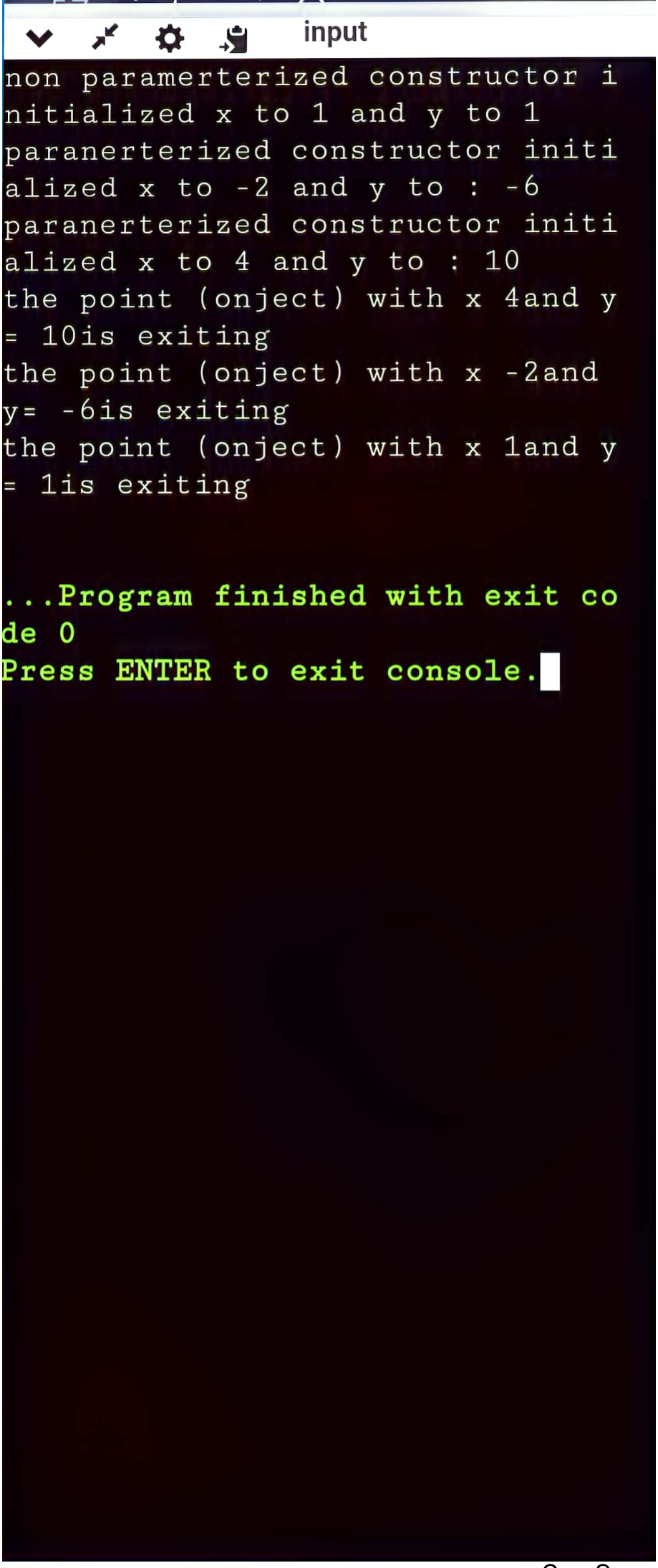
→ The value of X is : -2 the value of y is: -6

→ The value of X is : 1 the value of y is: 1



Note | أول ما تشوف في البرنامج بقول destructor وهم جزء يحفظ ترتيب الـ object [توفيت ثاني اول] ويرجع للـ destructor وتشوف هو طالب منك في الكود تتبعنا [جملة طباعة]

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

```
/************************************************
// VARIABLES SCOPE: FILE, FUNCTION AND CLASS SCOPE
************************************************/
1.      #include <iostream>
2.      using namespace std;

3.      int x = 100;

4.      class test{

5.      public:
6.      test(int a,int b,int c):x(a),y(b),z(c)
7.      {}
8.      void function()
9.  {
10.      int x = 44;
11.      cout<<"x = "<<x<<"\t for the function scope \n";
12.      cout<<"x = "<<test::x<<"\t for the class scope \n";
13.      cout<<"x = "<<::x<<"\t for the file scope \n";

14.      }

15.      private:
16.      int x,y,z;

17.      };
18.      int main()
19.      {

20.      test t_object(10,20,30);

21.      t_object.function();

22.      cout<<"x = "<<x<<"\t local to the main  \n";
23.      cout<<"x = "<<::x<<"\t global \n";

24.      cout<<"It is incorrect to use test::x since this syntax could be used inside
    a member function"<<endl;
25.      cout<<"It is incorrect to use t_object.x since x is private"<<endl;


26.      return 0;
27.      }
```


الممسوحة ضوئيا بـ CamScanner

Set 1 : 4

* في c++ الـ [Scope] لبعض الـ [Block] بعرف الـ [Block] باستخدام [ ] [ ]
← يقدر اعمل نوعين من الـ Scope: Class, Function

* علاقة الـ [Varible] في Block :
الـ Varible ممكن يكون داخل الـ [Scope] او خارجه وبيسمى إذا كان خارجه [Global]

[Global] int x; x=1;

test
{
int x=2;
}

* عدد الـ [X] في code = 3
1. في الخارج [Global]
2. Local [داخل الـ test او main]
3. داخل اي Fun. اخرى

main
int x=50;

الفرق بينهم كانه الـ Global تنشئ من أول ما الوصل عندما لبداية البرنامج
الـ [X] داخل الـ test اول ما الوصل الـ [ ] ح تنتهي

Function
X

ح تشوف الـ (1) لأنه [X] الباقين محجوبين عنها .

* أماكن تعريف المتغير قبل الـ [test block] لغير الـ [test block] داخل الـ [main] برا الـ [main]

في البرنامج في [line 3] ⇐ X=60 (Global) / [line 10] ⇐ X=44 / [line 16] ⇐ private [X]

test
int x=2
Function
x=44

لجوهنا إذا كنت داخل الـ Class + برا الـ Function
لجوهنا إذا كنت داخل الـ Class + داخل الـ Function

private [X] ← الـ [X] اكاصة من Class test
متابقدر اعمل obj من الـ Class test
[1] لما اعمل obj
[2] ما راح الـ private أعامي [X] لانه اذا ضلت موجودة ح تحجب من اي مكان آخر

[line 11] : لما يطلب [X] بجيب الـ local اذا أمكن موجودة [لانه الاقرب لاني]
والـ local الـ x = 44

[line 12] : لما يطلب [test::X] ⇐ [اسم Class :: Varible] بجيب القيمة اكاصة في Class التابعة لها

[line 13] : لما يطلب [::X] بقصد انه يبه الـ Global لـ x = 100

In General : دائماً بدور على [local Value] إذا كان فيه قيمته بتظهر أعامي وإذا ما كان في بيروح على obj اذا كان فيه
اذا استخدمت المتغير وهو مش موجود لا local ولا Global ح يعطي error!

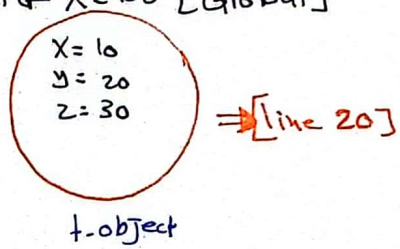
* خلاصة الكود : عدد شبابه الـ Varible في الاسم بـ الـ code تبقى ، لكن ممنوع
شبابه في نفس الـ Scope لانهم داخل الـ Scope الواحد تكون الامور واضحة

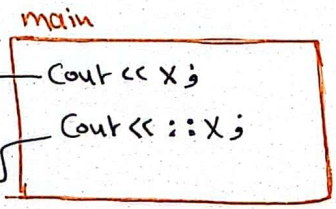
الممسوحة ضوئيا بـ CamScanner

* لو حطينا في main قبل [line 20] جملة [int x = 20;]
  - output الخاص في [line 22] ح يطبع الـ 20
  - output الخاص في [line 23] ح يطبع الـ 100 (لأنه :: هذا الـ local اذا ما في ح تكون error !) ← معناها خذ قيمة x الـ Global إذا ما في

* [line 22] من خلال وجود الـ x مثلاً آخذ ؟!
  1. x = 100 ✓ موجود [لقيراحيه متطول x لوحده او كتابة x ::]
  2. x = 44 ✗ مش موجود [راحت معاني صيغة تجيبها لأنها الـ Fun داخل الـ object] [لأنها كون داخل الـ Fun الـ obj معين]
  3. x = 10 ✓ موجودة [داخل الـ obj لأنه بعمله عالشي لاجيبها من خلال [object name . x] تكون [Global]

[Global] x = 100 ← اول اشي بنحجز في memory لهين بعدين عن main





main داخل الـ local ح تجيب القيمة القريبة منها

Global [x] ← بجيبها قيمة الـ

* cout << t_object.x; ← عشان اطبع الـ [x] داخل الـ obj لازم

Public الجملة ما ح تربط لأنها اصلها الـ private إذا

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لنطلع الـ [output] ooo

1) لا يوجد [Global object] هي البداية من main [line 20] بحكيلي اعمل object اسمه [t_object] من الـ Class اسمه test هي اول ما انشئ الـ object لستدعيلي الـ Constructor وصرت فيه هي [Parameter Constructor] ⇐ [line 6]
اعطي X لـ 9 و y لـ b و z لـ c صار يوجد شيء آخر اعمله من Constructor !؟
← لا يوجد لأنه صار فقط بعمل initialization للقيم بدون عمل طباعة او شيء آخر [؟؟]

2) [line 21] ⇐ انشئت الـ object باسم عائلته لـ Function [line 8]
X=44 [local] ← محجوبة عن غيرها الـ Function (بنهاية بـ تختفي الـ X)

→ X = 44 For the Function Scope [local] الأقرب لي
→ X = 10 For the Class Scope [object] class اكاهة من
→ X = 100 For the file Scope [Global] [::]

3) [line 22] صارت طباعة طالبة قيمة X الأقرب X لـ main في Global لأنه لا يوجد local
→ X = 100 local to the main

4) [line 23] صارت طباعة طالبة قيمة X الـ Global من [::]
→ X = 100 global

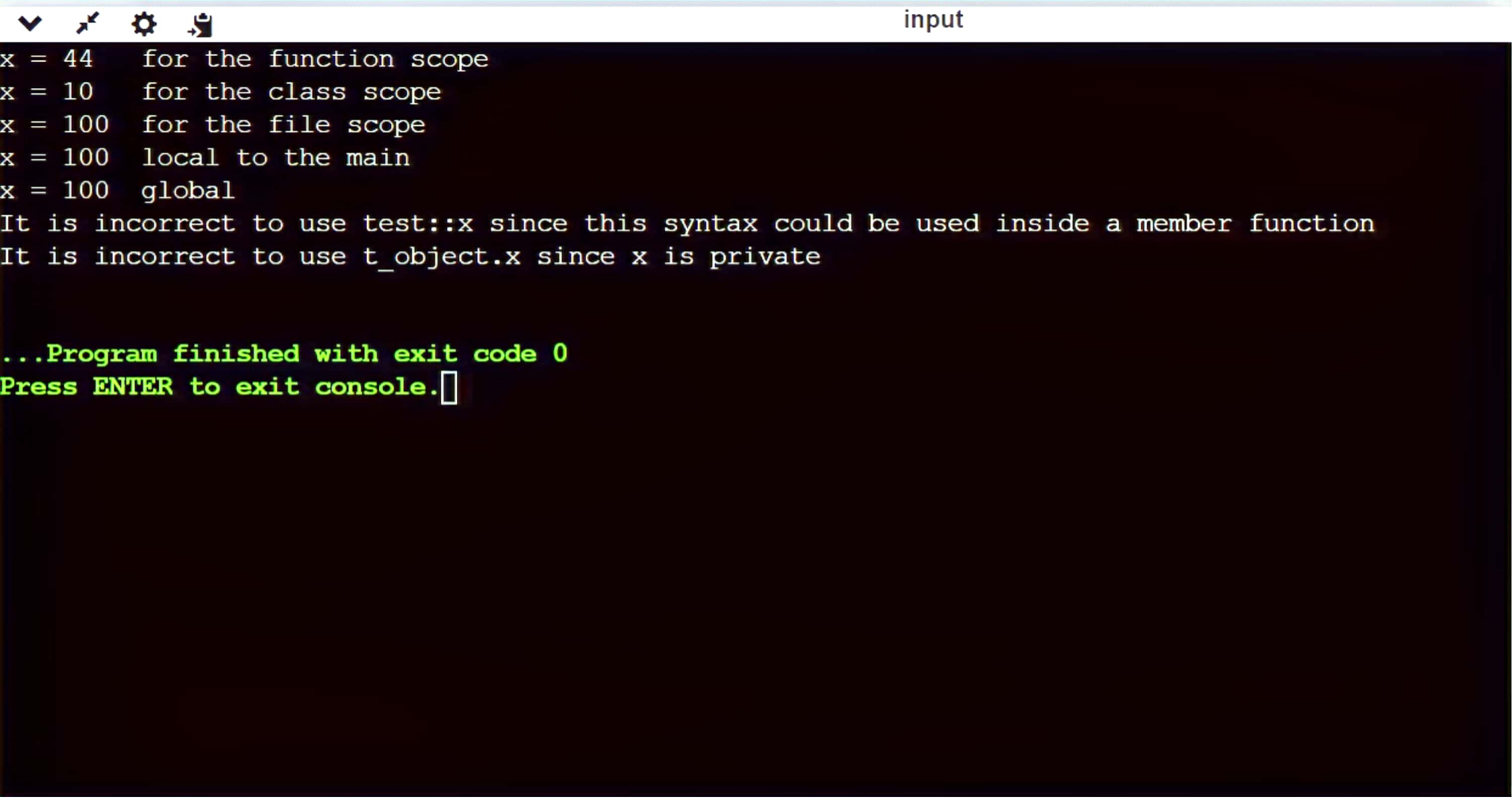
5) [line 25 + line 24] ⇐ مجرد عملة طباعة تظبرعا في بين الـ [" "]

→ It is incorrect to use test::x since this syntax could be used inside a member function

→ It is incorrect to use t_object.x since x is private

بالنهاية لا يوجد في الكود [Destructor] هي ما بيهمنا لوقت الانشاء و حذفه.

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

```
/************************************
// ARRAY OF OBJECTS + POINTER TO OBJECT
************************************/
1. #include <iostream>
2. #include <string>

3. using namespace std;


4. class student {
5.        string name;
6.        long id;
7.        int year;
8.        char *major;

9. public:
10.       void set_attributes(string na,long no, int ye=2014, char *maj="CE")
11.       {
12.       name=na;
13.       id = no;
14.       if(ye ==2014||ye==2013||ye==2012||ye==2011)
                        year = ye;
15.       else
                        year = 2014;
16.       major = maj;

17.       }
18.       void display_attributes()
19.       {
20.       cout<<"hello"<<endl;
21.       cout<<"iam "<<name<<"   a student with ID : "<<id<<endl;
22.       cout<<"Enrolled HU in : "<<year<<"and my major is
     :"<<major<<endl<<endl<<endl;
23.       }

24.       };

25.       int main()
26.       {
27.       student students[3];

28.       students[0].set_attributes("ahmad",20140301);

29.       student *pointer_to_student = &students[1];

30.       pointer_to_student->set_attributes("Ali",20140303,2003);

31.       (pointer_to_student+1)->set_attributes("Bayan",20130304,2013,"EE");

32.       for(int i=-1;i<2;i++)
33.       (pointer_to_student+i)->display_attributes();
34.       return 0;
35.       }
```

- Objects with one data members could be treated specially by initializing them using the initializer list
- A default constructor is created for every object in the array of objects, but the constructors have null body

الممسوحة ضوئيا بـ CamScanner

Set 1 : 5

* نفعل Array عناصره [object] و Pointer يؤشر على [object]

[line 10] : الـ Function تاخذ (4) من الـ Calling لازم امرر قيم للي ما عندهم Value

مثل الـ String والـ long [ ]

عندك 3 خيارات لتمرير القيم:
- اول قيمتين امرر [وخذ الباقي من default المعهود]
- امرر (3) قيم [الباقية اخليها default] [CE]
- امرر (4) قيم كاملين [ح يحذف الـ default]
(إذا عنده قيمة وجوده وعدم وجوده واحد)

[line 27] : اعمل (3) [Array] الشيء بينهم كلهم إلهم نفس قيمة الاسم
* ختلف إذا تبقول على [obj] معين ما تقصد كل [obj] غيره

* عندنا شاد الـ array لتستدعي الـ [Constructor] بعدد عناصر الـ array مثل المثال تبعنا (3) :
لتستدعي (3) مرات ، نفس الشيء الـ [destructor] ، بحال كانت في جملة طباعة للـ Constructor والـ destructor
كل وحدة (3) مرات طباعة [الفرق إنه بس يطهر مثي destructor آخر واحد الشيء اول واحد يطهر]

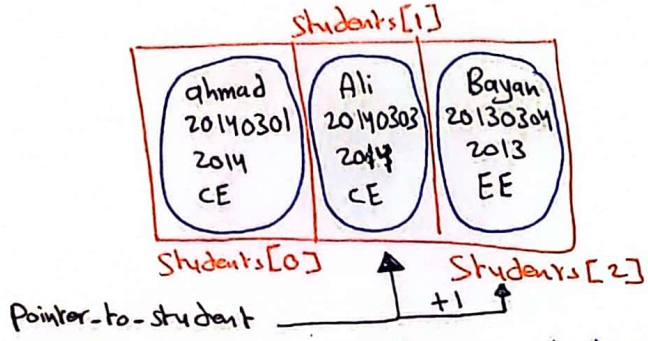
[line 29] : عرفنا Pointer [تذكير ، لو بدي أعرف Pointer ⇐ [النوع * اسم Pointer]]
(اسم الـ Pointer * الـ نكواك اسم)

[line 30] : [->] بتشير الـ [.] متى بحط كل وحدة فيهم ؟!
↳ إذا بلي كتبته obj بحط [.]
إذا بلي كتبته Pointer بحط [->]
↳ باقي اللي بصدد امين الاب والابن [استرجاع]

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لنطبع الـ [Output] ooo

1) يروح [Global] هو البداية من [main] line 27] وانشئ Array عناصرها عبارة عن Students عددهم [3] من Student class يسمى object
← لازم يروح Constructor هو ما استدعيه



⇒ Array تم إنشاء

2) [line 28] ⇐ [Set-attributes] Fun. استدعي الـ [Students[0]] array ⇐ [line 10] لـ اول عنصر
بالدالة مررلي سنتين هو اول سنتين حسب التقرير و آخر سنتين (default)
← عبيت أول عنصر بقيم التقرير

3) [line 29] ⇐ انشأنا pointer اسمه [Pointer to student] نوعه من class [Student] بأشر على ثاني عنصر من [Students[1]] array
← وقفت pointer على ثاني عنصر في array

4) [line 30] ⇐ استدعيت الـ [Set-attributes] Fun الـ pointer اللي حالياً بأشر عليه
← الـ pointer بأشر على العنصر الأول وبمرر 1 فيتم هو آخر سنة default وليش صحيح السنة 2014 مش 2003
زي ما نفيت من التقرير؟! لأنه لما نروح على Fun بترجع الـ scope بدأ من [12 - 16] بنعبي الاسم + Id عادي
اما الـ year إيها IF و else الـ 2003 من else هو year=2014

5) [line 31] ⇐ زيد الـ pointer بـ (1) اللي كان على العنصر الثاني رح يصير على العنصر الثالث لهيك استدعيته

6) [line 32] ⇐ For loop من [-1] لـ [1] استدعي الـ pointer اللي بأشر عليه قيمة [i] لـ Fun [Set-attributes] ← مرر على الـ pointer وحسب القيم [هي default أو تم تغييرها] [display attributes]

[Pointer-to-student+i] ⇐ i=-1

Fun لـ obj واستدعي الـ [Ahmad] اللي pointer to student+(-1); ← zero
[display attributes]

i=0 ← Pointer-to-student+(0); ← one
Fun لـ obj واستدعي الـ [Ali] اللي بأشر عليه
[display attributes]

i=1 ← Pointer-to-student+(1); ← two
Fun لـ obj واستدعي الـ [Bayan] اللي بأشر عليه
[display attributes]

* ليس ليفت الـ [obj] لـ Fun كل اللي رح تعمله رح تطبع لي الطباعة بناء على قيم الـ obj فقط

الممسوحة ضوئيا بـ CamScanner

```
hello
iam ahmad a student with ID : 20140301
Enrolled HU in : 2014and my major is :CE


hello
iam Ali a student with ID : 20140303
Enrolled HU in : 2014and my major is :CE


hello
iam Bayan a student with ID : 20130304
Enrolled HU in : 2013and my major is :EE




...Program finished with exit code 0
Press ENTER to exit console.
```

الممسوحة ضوئيا بـ CamScanner

```
/*****************************************************************************
// PASSING OBJECTS BY VALUE AND BY REFERENCE + RETURNING OBJECTS FROM FUNCTIONS
*****************************************************************************/
1. #include <iostream>
2. using namespace std;

3. class date {

4. public :
5.       date();
6.       date(int , int ,int );
7.       void print(){      cout<<d<<"\t"<<m<<"\t"<<y<<endl; }
8.       int d,m,y;

9. };

10.       date::date(int a,int b,int c)
11.       {
12.       d=a;  m=b;  y=c;
13.       }

14.       void fun (date x,date &y,date *z)
15.       {
16.       x.d+=2; x.m+=2; x.y+=2;

17.       y.d+=2; y.m+=2; y.y+=2;

18.       z->d+=9; z->m+=9; z->y+=9;
19.       }
20.       date special_function(date &ob)
21.       {
22.       ob.d++;
23.       ob.m++;
24.       ob.y++;

25.       return ob;
26.       }
27.       int main()
28.       {
29.       date a(10,10,10);
30.       date b(20,20,20);
31.       date c(30,30,30);
32.       date *d=&c; // d is not an object, it is a pointer to object c

33.       fun (a,b,d);
34.       a.print();
35.       b.print();
36.       c.print();
37.       d->print();

38.       date e(2,4,6);
39.       e.print();
40.       date f = special_function(e);
41.       f.print();

42.       return 0;
43.       }
```


الممسوحة ضوئيا بـ CamScanner

Set 2 : 1

* نقدر نمرر الـ object لـ Function بطريقتين — [Cut, past] by reference / [copy, past] by Value

← اي [object] ببعته [by Value] بمعنى اني باخد نسخة من الـ object المجوت وعشان يوضعني [Copy Constructor] الـ Calling

← اي اشي ببعت [by Value] انت اخزت الـ object استنسخت منه نسخة وبعتت النسخة لـ Function [التصديق ما بتصير الاصل]

← اي [object] ببعته [by reference] يعني باخد اشارة الـ address المخزن في [array] او ببعت الـ object نفسه ما بعمل استدعاء لـ [Copy Constructor]
[اشارة الـ address [$] تنسخ الـ Copy]

← [line 14] : data x , data $y , data *z

data x ← Calling by Value ← ما بتحتاج الـ Copy Constructor

data $y , data *z ← Calling by Reference ← ما بتحتاج الـ Copy Constructor

[إشارة الـ address [$] + اشارة الـ pointer [*] + الـ array ← [by reference]

* ممكن اصنف الـ Value والـ reference اللي بترجع مؤشر الـ Fun [line 14] Calling كان لسلم من [object, object, pointer] بس ما اتخلص برجع void [لا شيء]

← [line 16] : الزيادة على [x] للارجع على [main] بعد انتهاء الـ Fun ما بغير اي اشي عن [Calling by Value]

← [line 17 + line 18] : الزيادة على [y, z] للارجع على [main] بعد انتهاء الـ Fun (كل التغيرات اللي صارت بـ obj كانه بيعته نفسه وبرجع لنفسه لانسخة عنه)

← [line 20] : الـ data special Function
data special Function (data $ob)
ماني اشارة $ يعني ترجع [by Value]
لو [void] ولا شيء [error!] ولا نمارح يرجع اشي ومارح استخدمها في [main]
لو حصلتي اشارة $ ما بحتاج copy
Calling [Copy Constructor] لنفسه يعني ما بدي الـ [ob] انا بيعت الـ address [$]
لو حذفت اشارة [$] رح تغير [by Value] يعني رح تغير [Calling Copy Constructor]

data special Function (data $ob) الـ Fun بتوقع مني [Object by reference] برجع [Object by Value]
يعني رح اعتبرها [object] تشتغل عليه لنفسه إذا غيرت اشي رح يظل على طول بالنهاية الـ [object] يلي برجع [by Value]
الفرق بين ترجع رح اعطيها مكان الـ Calling : الترجع بدون by Value [Calling Copy Constructor]

الممسوحة ضوئيا بـ CamScanner

[line [29+30+31]] ← عملت object (3) = b, a, c : [object]

[line 32] ← عملت pointer اسمه [d] نوعه من class اسمه [date] بأشكل object = d

[line 33] ← الـ Fun تاخد object (x,y,z) الـ x,y,z ليها [Calling by Value] هي داخل الـ main

لما انتهت الـ Fun الـ obj فيها رجع الأول شوفنا الـ Fun رجعلي على [obj] الصورة الأصلية ما بتضاب

[الـ obj القديم اسمه [a] / الـ obj الجديد اسمه [x] ← بدلنا على [x] ما صبنا الـ [a]]

الـ x,y,z رجعوا ليصيرلهم [Calling by reference] هي الـ obj ماخدين منه صورة ثانية وحطه نفسه ونبعته لـ Fun

اشتغل عليه وجبلي اياه نفسه هي الـ obj = b الـ obj رجع ليطلعوا لقسم الـ Fun.

[الـ obj = b طلع لـ Fun حار اسمه b وخزل نفسه بالتالي اي تعديل على (y) رح يصير على اللي نبعت (b) لهيك الشيء لـ [a]]

[line 34] ← الـ [a] نفسه ما تغير بقي محفوظ من main / الـ [x] راح

[line 35] ← رجع اسمه (b) بطل اسمه (y) ليس هيك موجود داخل memory

* الـ (x) طارت الـ (x,y,z) فضلت التعديل على (x) راح معها طارت

[line 37] ← الـ pointer رجع اسمه d = وبطل

[line 38] ← الـ (e) نفسه لـ (object) مع قيمته بطلع لفوق [by reference] هي هيك [Calling Copy Constructor]

[line 40] ← اعمل object جديد اسمه (F) خليه يوخذ هيك من اللي برجعه (special function) بالتالي

الـ special function لهد التنفيذ رح ترجع [return obj] بشيله وبحطه هون

← العملية زي كأنها لنسخ هي رح يكون [Copy Constr.] كأنه by value

← لو حطيت [&] هو نفسه رح يصير اسمه (F)

* وقت اللي بيجي [Copy Constructor] ← [line 20] قبل استدعيته في الـ main ?! في [line 40]

لما بقت الـ (e) ترجع لـ ob بأول حالي جاي [Copy Constructor] اول ما عبالاته في اشارة [&] هي الـ (e)

ترجع لـ ob ليس من طريق الـ copy ترجع نفسها هي الـ = لنفس الـ ob ما بتعمل copy

بالمقابل الـ [Fun] ليس تخلص ترجع ob من جديد اسمه F ?! هيك الـ ob نفسه اذا كان في اشارة [&]

اذا كان في اشارة [&] [اذا مش موجودة اشارة address of معناته (ob) اعمل نسخة ثانية خليه يوخذ نفس القيم واسمه F]

الممسوحة ضوئيا بـ CamScanner

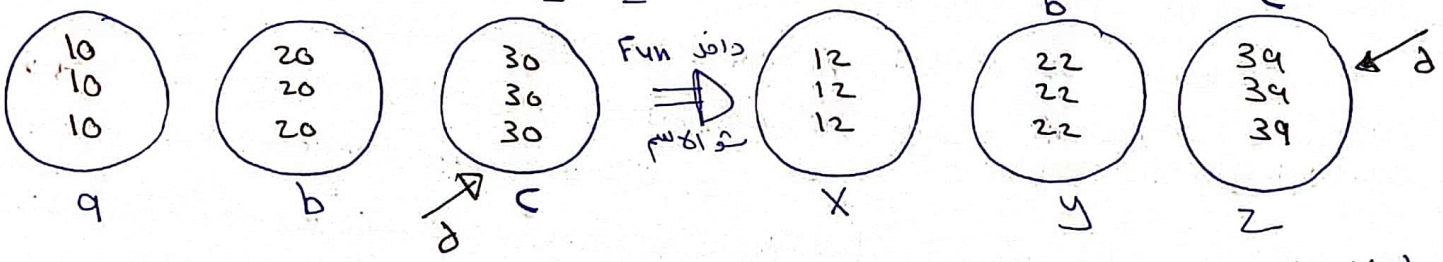
ooo تتبع الكود لنطبع [output] ooo

1) ماضي [Global obj] هو البداية من main [line 29 الى line 31] يبدء نقل date class من object

2) [line 32] date اسمه class من d اسمه pointer نعمل
بالسهم [obj c]

3) [line 33] المتبادل objects الـ (3) اللي عملتم لـ
Fun مع الانتباه شو نوع العبث [a] Value
[b,c] Reference

الـ (a) عوامل عليها كمثال ماخ فيبين في main هو ما تقلب حالك فيها اشتغل على (reference) لانه العبث محفوظ فيها
الـ Fun [line 14] لشغله انه يضيف على الاحداثيات اما لـ c او a



لما نرجع لـ main رح نوخذ معنا الـ z و y بلين عا نفهم الـ b,c اما الـ X ماخ تروح رح نوخذ الـ (a)

4) [line 34] المتبادل [obj a] لـ Fun[print] ⇐ [line 7] رح يطبع الـ output

→ 10 10 10

5) [line 35] المتبادل [obj b] لـ Fun[print] ⇐ [line 7] (b نفسه y)

→ 22 22 22

6) [line 36] المتبادل [obj c] لـ Fun[print] ⇐ [line 7] (c نفسه z)

→ 39 39 39

7) [line 37] المتبادل pointer (بلبابا سهم d) لـ Fun[print] ⇐ [line 7]

→ 39 39 39

8) [line 38] اعمل obj جديد اسمه e من class اسمه date واعطيه القيم

9) [line 39] المتبادل obj (e) لـ Fun[print] ⇐ [line 7]

e: 2 4 6 → 2 4 6

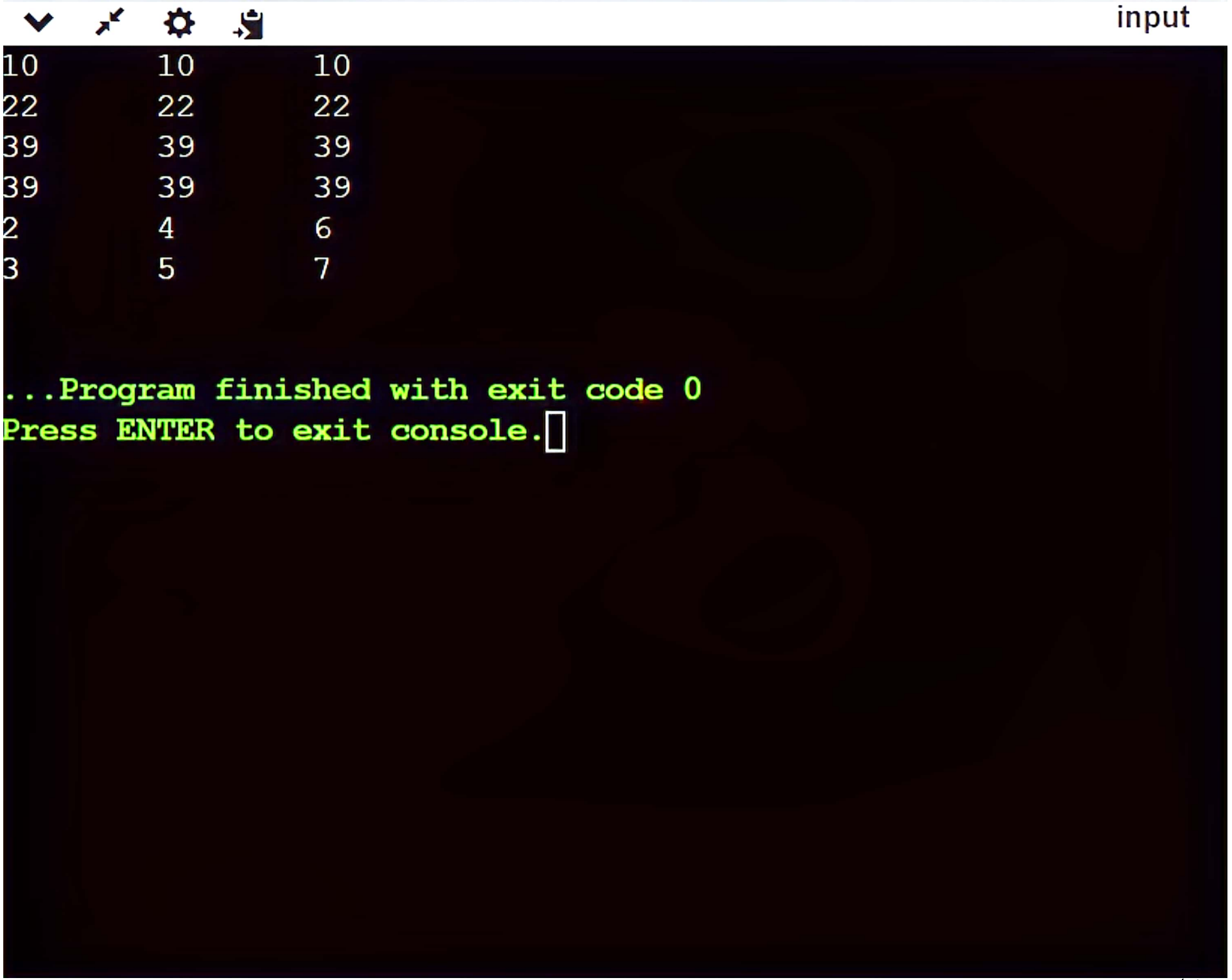
10) [line 40] المتبادل الـ (obj e) لـ Fun [line 20] ورجعه واعطيه لـ obj جديد اسمه F

11) [line 41] المتبادل الـ obj F لـ Fun[print] ⇐ [line 7]

e: 2 4 6 ⇒ ob=F: 3 5 7

→ 3 5 7

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

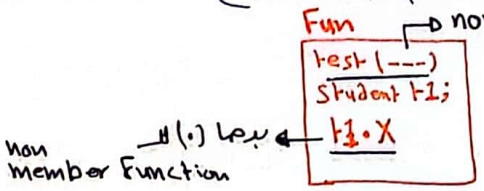
```
/*****************************************
// FRIEND FUNCTION
*****************************************/
1. #include <iostream>
2. #include <cmath>
3. #include <string>

4. using namespace std;

5. class B;      // forward declaration
6. class A {
7. private:
8. int data;
9. public:
10.      A(): data(12){ }
11.      friend int func(A , B*);   //friend function Declaration
          };

12.      class B {
13.      private:
14.      int data;
15.      public:
16.      B(): data(1){ }
17.      friend int func(A , B *);  //friend function Declaration
18.      };


19.      int func(A d1,B *d2)

20.      {
21.      cout<<"function F has the power to access the private data in both class A and
   B"<<endl;
22.      if(d1.data > d2->data)
23.      return d1.data;
24.      else
25.      return d2->data;
26.
27.      }

28.      int main()
29.      {

30.      A a;
31.      B b;
32.      B *p=&b;
33.      cout<<"Data: "<<func(a,p);
34.      int t;
35.      cin>>t;
36.      return 0;
37.      }
```

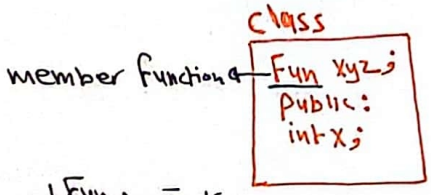
- A friend function may be friend to more than one class

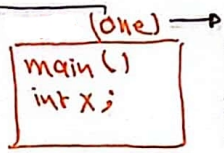
الممسوحة ضوئيا بـ CamScanner

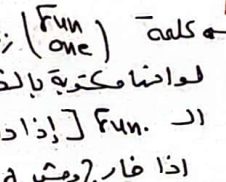
Set 2 : 2

* بعتاج فكرة الـ [Friend] إذا الحذري أكثر من class وبدي واحد
يقدر يشوفهم من برا كلهم [بعرفه خارج الـ class لكنه زي كأنه داخل الـ class كأنه member]









* داخل الـ main مابزبط اكون Function مع الـ Function كل الأخير فوقهما موال main

الـ Function ممكن تكون بـ Block كامل لوحدها + أكون ممكن داخل class [بغير scope داخل scope] Function الـ

يلي ورثانية الـ class ← تسمى non-member Function : مابزبط اعمل Access عن Varible صيغة وتفعيل

تسمى member Function ← : بقدر اعمل Access إذا الـ Varible (Public)

[obj اسم . Varible]

* الـ Fun (xyz) جاية member بتقدر توصل الـ (x) مافرق إذا كانت Public او Private بالله هنا بالداخل إذا بالخارج بتفرق.

← من خلال اول اشي بقل [object] لعين بجيب العنصر عن طريق (اسم Varible . اسم object)
الشرط الثاني لما اعمل حماية الحرية وانزل كل الـ class يلي فيه العنصر يلي حطيت اسمه لعبد الـ (.) بزبط غير جيك مابزبط public

[line 5] : الـ Class B في الـ class A ← السبب إنه لقدام الـ class اعرفه حقاً

[line 7] : بقدر توصلها الـ [Fun.] يلي جزء من class + Function Friend ← يلي بتكونوا بزا ليس بوجودا ضمانين كاعتم جزا

[line 10] : Data(12) ⇐ data=12 ≡

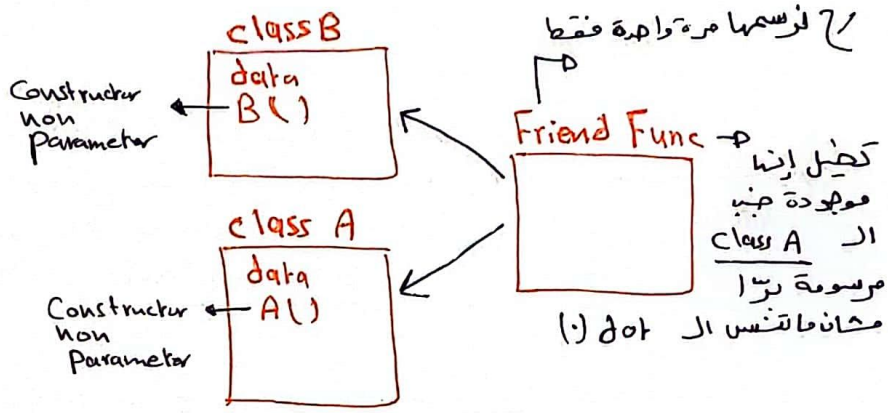
[line 11] : Friend Func ⇐ Friend to class [يعني الـ] بدي الـ body يكون مكتوب برا وهي ليست [member] لكن من ناحية الـ Access member up

الممسوحة ضوئيا بـ CamScanner

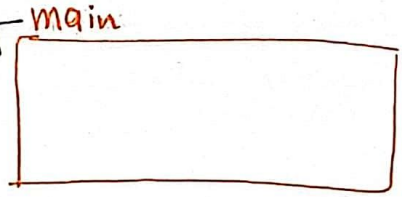
* by Value : التعديل ليس بشكل دائم
* by reference : التعديل لغل موجود حتى مع نهاية الـ [main] أو الـ [Scope]

[line 19] : الـ Fun. Friend لوضع معين [object (A) , pointer (B)]

[line 22] : d1: data ← [object] يسبقها "استدعاء"
d2-> data ← [pointer] يسبقها "استدعاء"
private : data



نرسمها مرة واحدة فقط
تخيل إنها موجودة جنب الـ class A مرسومة برا مشان ما تنعس الـ dot (.)

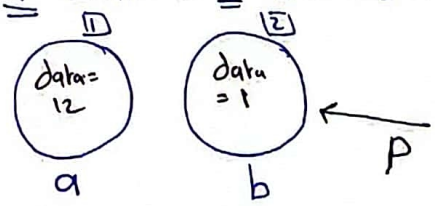


كما بقدر يوصل للـ data الـ [private] data
لكن الـ Friend Func هنه خارجية عن طريق

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لنطلع [output] ooo

1. ⟸ يوم [Global obj] في البداية من main ⟸ [line 30] بره نعمل object اسمه a من class A



2. [line 31] ⟸ بره نعمل object اسمه b من class B
3. [line 32] ⟸ اعمل pointer اسمه p من class اسمه B عليه يأشر على b
4. جملة طباعة واستدعاء الـ Func ومررله p, a ⟸ [line 19]
   الـ object من A وال pointer في B

لما ننشئ الـ [obj] لازم نعرّف الـ Constructor لكل [initilization]

→ Function F has the power to access the private data in both Class A and B 12

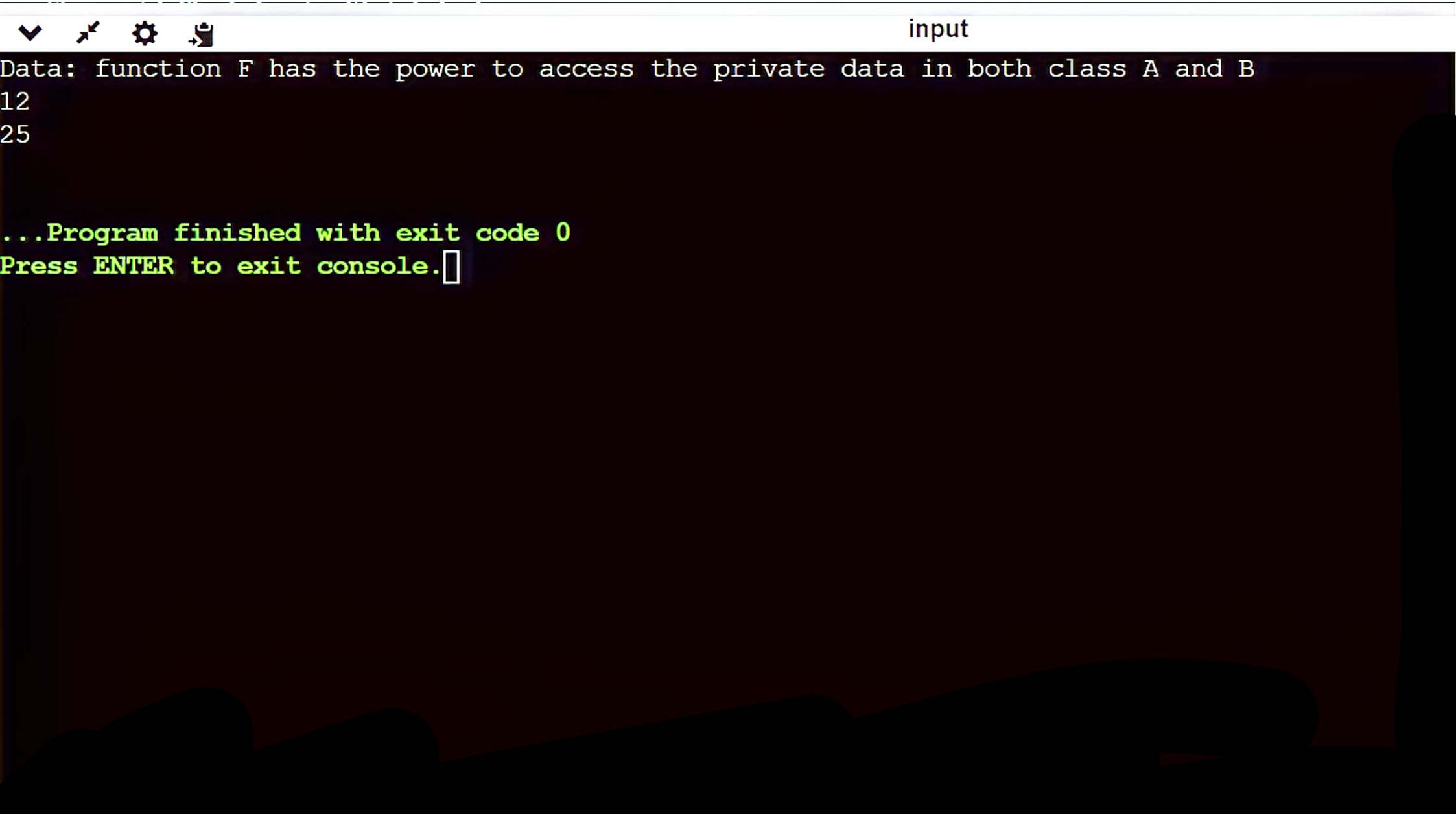
← جملة IF/else اذا صح رجع الأكبر [object] إذا غلط رجع الأقل [pointer]

Ex] object = 12 / pointer = 1

IF (12>1) ⇒ T return 12 ⇒ output

5. [line 35] ⟸ جملة cin [انا مني output فبص دخلة 25]

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

```
/*********************
// COPY CONSTRUCTOR
*********************/
1. #include <iostream>
2. #include <new>
3. #include <cstdlib>
4. using namespace std;
5. class array {
6.       int *p;
7.       int size;
8.       public:
9. array(int sz) {
10.      p = new int[sz];
11.      size = sz;
12.      }
13.      ~array() { delete [] p; }      destructor
14.      array(const array &a);         copy const.

15.      void put(int i, int j)
16.      {
17.      if(i>=0 && i<size)
18.          p[i] = j;}

19.      int get(int i) {    return p[i]; }

20.      };
21.      //Copy Constructor
22.      array::array(const array &a) {
23.      int i;
24.      p = new int[a.size];

25.      for(i=0; i<a.size; i++)
26.           p[i] = a.p[i];
27.      }
28.      int main()
29.      {

30.      array num(10);
31.      int i;
32.      for(i=0; i<10; i++) num.put(i, i);
33.      for(i=9; i>=0; i--) cout << num.get(i);
34.      cout << "\n";
35.      // create another array and initialize with num

36.      array x(num); // invokes copy constructor

37.      for(i=0; i<10; i++)

38.      cout << x.get(i);

39.      return 0;
40.      }
```

- array a(10);

```
array b(10);
b = a; // does not call copy constructor
```

- copy constructor must be included when the class contains pointer to data to ensure correct operation

الممسوحة ضوئيا بـ CamScanner

Set 2 : 3

* Copy Constructor : هو نوع من [Constructor] يمكن الـ [initilize object] بدل [initialization] لنفسه من غيره

لازم نعرفه بالـ Class بنفس الـ Copy لكن داخل الـ [initialization] بعمل

* الـ [Copy Const.] نفسك الطرف الايمن يوضح المعلومات يلي عنده للطرف الايسر

* كيف امرر الـ [Copy Constr.] في الكود وجوده ؟
  1. يوضح [object] مبني عليه حالتين
  2. اشارة [&] Refrence
  3. Constant

اي [Call] بصير عندك (3) شغلات بكون [Copy Const.]

* عشان استدعي [Copy Const.] من خلال الصفحة

New = old

object [حالة] نعمل هيك

obj. نعمل سابقاً

[3] حالات على هالصفحة :

Ex)

```
Class one
{
public :
one (int a) : X(a) { }
one (const one &temp)
{ Cout << " Bye"; }
int X;
:
:
```

وجود الـ (3) ميزات

[Copy Cons.] عوض [Copy Const.] اكتابة لـ

Ex) one t ; ← حالة (1)

:

one tt = t ;

old [تذكير] ← [ما تذكر شي] new

لو عرلت حطيت [int one tt] بوقتها بـ one tt = t ; طابع لعيني الـ [Copy]

Ex) one t = Fun ← حالة (2)

new

* جود استدعاء الـ [Copy Constr.]

1. حالة (1) : واحد جديد [Fresh] بوخذ قيمة من واحد قديم كعرف سابقاً .
2. حالة (2) : new = old القيمة [Return by Value] .
3. حالة (3) : [Calling by Value] ← يعني يكون في Function و إذا استدعيت الـ Function في main لازم غرضها جاهز prototype من الـ object الـ Class one العجة يلي بنمررها main جاهزة قديمة بروح للـ [obj] إجبر .

Ex) ← حالة (3)

function (one L)

new

main()
one xyz;
function (xyz)
old

الممسوحة ضوئيا بـ CamScanner

← [line 9] : Constructor الـ array ماإليها اسم ليس وضعت عليه [pointer] ويأشر على أول [location]
based [zero] الـ

← [line 10] : كلمة [new]

* يلي ماإليها [new] بنحجز في اللغة الثانية بالبرنامج
[وقت الحجز]
x = 6

Ex] main ( )
{
int x = 10;
p = new int (3)
}

أما يلي معها [new]
أنك ما تطلع الشاشة السوداء [بنحجز] داخل mem.
يعني الـ Compiler ماإله دخل فيها
الغرض [وقت الحجز]

إذا بدي اطبع الـ 10 بكتب ⇐ [Cout<< x ;]
إذا بدي اطبع الـ 3 بكتب ⇐ [Cout<< *p ;]
← المتغير (3) ماعليه اسم إذًا ما بقدر استفيد منه لو السطر يلي
بعده عكيتبدي الـ (3) ماح اقدر اجيبه إلا اعطيه مؤشر [pointer]
[delet p] : احذف المؤشر مع يلي بأشر عليه
[بتكون delet الـ pointer مسؤولة إنها destructor]

* عند لتيكون القوس بنضف حجز الـ [x] وإما الـ [3] بتبقى في memory
ليس جيد لأنه إذا الكود خلص لازم كل تعليمة تنضف الـ [3] ليس له اسم ولا بنضف أوتوماتيكيًا
[قبل وضع الـ pointer عليه (p)]

← [line 13] : إشارة الـ [~] ⇐ destructor وجود لسبب وجود الـ [new] بحذف الـ new
[وجودها مع destructor لأنها داخل الـ class]

⚠ كذف الـ array يلي بأشر عليها pointer وضع بداية إشارة الـ [] array ثم الـ pointer ⇐ [p] ⇐ [] p

← [line 14] : اسم الـ class [ob] ⇐ const
إشارة الـ [&]
([ob + ob] ⇐ [ob جديد + ob قديم]) Copy Constructor ⇐ عمليات تنفذ على [لاستخدامه]

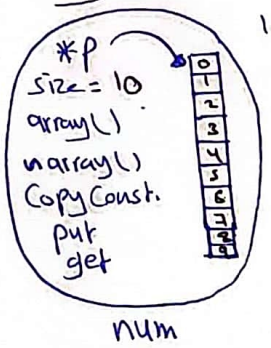
← [line 22] : وجود الـ body الـ Copy خارج الـ class ⇐ بدل التعريف الكامل [استدعاء]
[اسم الـ Fun :: اسم الـ class]

البرنامج لتيتفذ على [1] متأكد من error
[2] بحجز المتغيرات يلي ماحذرها new
[3] ينفذ

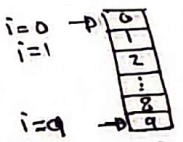
الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لنطلع الـ [output] ooo

1) يوجد [Global obj] في البداية من main [line 30] ⇐ أعمل obj اسمه num من Class اسمه array
اول مانفعل [obj] نستدعي الـ Constructor [line 9] ⇐ نعمل الـ array بـ size = 10



2) [line 32] ⇐ For loop من 0 لـ 9 اعبتها لـ num لـ [put] Fun ⇐ [line 15]
ومررله الـ [i] بكل لفة
← وظيفة الـ Fun في [IF statement] إذا الـ [i] أكبر من zero و [i] أقل من size [10]
عبي في array الـ [i] حسب اللفة القيمة بقيا الـ [i]



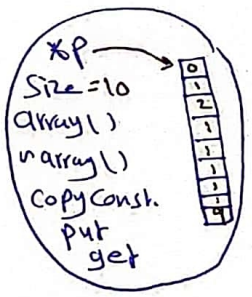
3) [line 33] ⇐ For loop من 9 لـ 0 اطبع المجلة
يلي مجوتة لانسا الـ [get] Fun لـ obj num
مرره الـ [i] حسب كل لفة [line 19]
← الـ Fun وظيفة برجع قيمة الـ index الخاص في اللفة

→ 987654321

4) [line 34] ⇐ new line
5) [line 36] ⇐ اعمل obj جديد اسمه [X] من Class اسمه [array] محتواه الـ obj منسوخة عن [obj num]
الـ line بنكرنا في تعريف [Copy constr.] تحديد للـ جديد .. استدعي [Copy Cons.] ⇐ [line 22]
← وظيفة الـ Fun لمرر الـ num [object] بدل كل (9) مرات (For loop) تتمشي من 0 لـ [size-1 = 10-1 = 9]
وبكل لفة تنقل Copy من (i) الموجودة في num لـ (i) الموجودة في (X) لحد ماتوصل لـ 10 وتطلع من loop

6) [line 37] ⇐ For loop من 0 لـ 9
اعبتها الـ X لـ [get] Fun ⇐ [line 19]
وبكل لفة ارجعلي الـ value لـ index معين (i)

→ 0123456789



=

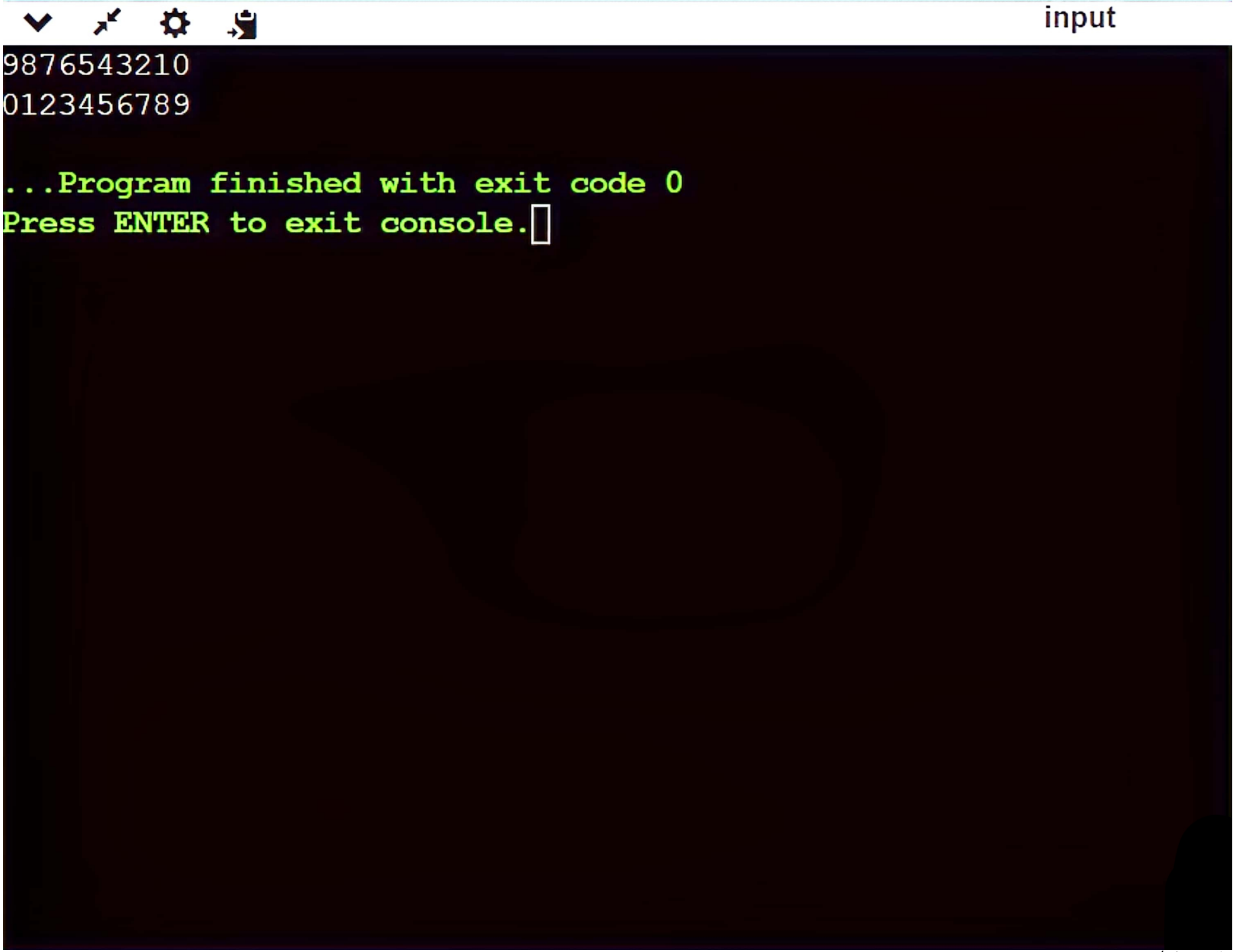
(main اسمه في) num
(line 22 اسمه في الاستدعاء) 9

X
لـ اي اشي private
طالبقر لوصل لإله

array الـ Pointer بالـ
[Public] مشان اكون اسرق
قيمهم

Note ← مشان تكون [line 36] statement صحيحة
[object] X يكون في التعريف public

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

```
class example
{
    int x, y, z;
public:
    example(int a, int b, int c) :x(a), y(b), z(c) { cout << "parameterized constructor\n"; }
    example(const example& left_object)
    {
        x = left_object.z;
        y = left_object.x;
        z = left_object.y;
        cout << "Values have been copied\n";
    }
    int get_x() { return x;}
    int get_y() { return y; }
    int get_z() { return z; }
};
void main()
{
    example one(100, 200, 300);
    example two(1000, 2000, 3000);
    example three = two;
    cout << one.get_x() << "\t" << one.get_y() << "\t" << one.get_z() << "\n";
    cout << two.get_x() << "\t" << two.get_y() << "\t" << two.get_z() << "\n";
    cout << three.get_x() << "\t" << three.get_y() << "\t" << three.get_z() << "\n";
}
```

```
Microsoft Visual Studio Debu
parameterized constructor
parameterized constructor
Values have been copied
100     200     300
1000    2000    3000
3000    1000    2000
```

الممسوحة ضوئيا بـ CamScanner

```
class example
{
    int x, y, z;
public:
    example(int a, int b, int c) :x(a), y(b), z(c) { cout << "parameterized constructor\n"; }
    // example(const example& left_object)
    // {
    //     x = left_object.z;
    //     y = left_object.x;
    //     z = left_object.y;
    //     cout << "Values have been copied\n";
    // }
    int get_x() { return x;}
    int get_y() { return y; }
    int get_z() { return z; }
};
void main()
{
    example one(100, 200, 300);
    example two(1000, 2000, 3000);
    example three = two;
    cout << one.get_x() << "\t" << one.get_y() << "\t" << one.get_z() << "\n";
    cout << two.get_x() << "\t" << two.get_y() << "\t" << two.get_z() << "\n";
    cout << three.get_x() << "\t" << three.get_y() << "\t" << three.get_z() << "\n";
}
```

Comment

```
Microsoft Visual Studio Debu
parameterized constructor
parameterized constructor
100     200     300
1000    2000    3000
1000    2000    3000
```

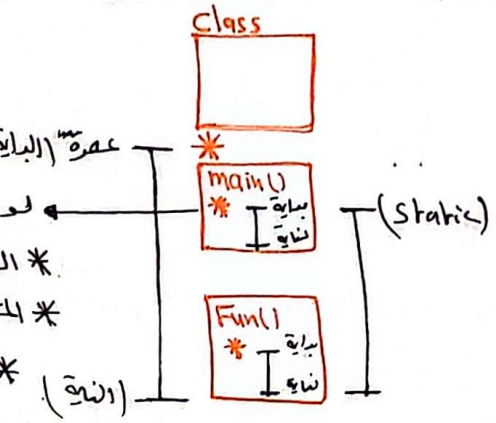
الممسوحة ضوئيا بـ CamScanner

```
/*****************************************
// STATIC vs AUTOMATIC OBJECTS
*****************************************/
#include <iostream>
using namespace std;

int i=1;
class test
{
public :
test(int x){ // parameterized constructor
          object_num = x;
          cout<<"object with ID "<<object_num<<" created"<<endl;
      }
      ~test()
      {
      cout<<"object with ID "<<object_num<<" destroyed"<<endl;
      }
      private:
          int object_num;
      };

      void create_ob()
      {
      static test T2(20); // STATIC LOCAL OBJECT
      test T3(i); // AUTOMATIC LOCAL OBJECT

      }

      test T0(0); // STATIC GLOBAL OBJECT

      int main()
      {
      static test T1(10); // STATIC LOCAL OBJECT
      for(;i<3;i++)
      create_ob();

          return 0;
          }
```


الممسوحة ضوئيا بـ CamScanner

Set 2 : 4

* [Static] ≠ ممكن تحول الـ [local] ليصير عاية [Global]
* لو بدي ارجع [obj] ممكن داخل الـ [main] و خارج الـ [main] اذا عرفنا داخل الـ [main] الـ [Fun] لتبقا فيتهم رح ينضفوا الـ [object] عمره محدود من بداية المجلة المعرفة فيه لنهاية الـ scope



* رح يطول عمره لنهاية البرنامج [كأنه Global] لوضفت (Static) لهاد الـ main رح يضل في main
* الـ [Static] ما بتغير الـ [scope] ≠ يعني إذا مكانة في مكان تالي
* المتغير يلي بستعمله [Static] و ما بزبط استخدمه في مكان تالي
* الـ [Static] ممكن تنضاف لمتغير او [obj] أو Fun.

[line 20] ← : ليس اوصل لعنده اول مرة رح اعمل دايرة الـ [object] و أعمل [inilization] للـ [T2 = 20] واكتب جنبه [Static] و مارح اعرفه ولا رح يضل لنهاية البرنامج

[line 22] ← : بتبيكره الـ [?] ← T2: رح يضل موجود
T3: رح ينضف و مارح يضل موجود [رح اعمل object واعرفه و مارح احط جنبه [Static] مع اسمح لـ [Fun] تضل لنهاية الـ [Fun]]

[line 23] ← : رح يضل موجود لنهاية البرنامج ≠ Global object رح انفذه مرة وحده و مارح ارجعلنده مرة اخرى

* شو لستفيدانه [object] يكون [Static] ؟! إذا بدي اعمل الـ [Fun.] إنما أكثر من مرة Call وبدي اعا شي يلي نحسب اول مرة اعتمر عليه

T2 ← : عمره بداءنا [line 20] لنهاية البرنامج و لوفي [main] كمان T2 ما بنفع و كونه Static ما بتغير الـ scope لكن بتغير العمر و كمان داخل الـ Fun إذا لقدر استخدمه داخل الـ Fun فقط

لو رحت من [Fun] الـ [main] شو رح يصير له ← T2: موجود لكن مارح اقدر استخدمه في main و كونه ليس scope اكاهة فيه
T3: ميش موجود بالتالي مارح اقدر استخدمه

T0 ← : لهيك الكتب [T0.] كونه [Global + Static]

* الـ T2, T3 : القنين [local] لكن واحد ميزته انه بدي اطول عمره فقط و يغير الـ scope اي انه ما لقدر اعمل Access خارج نطاق الـ [Fun] يلي تعرف فيه

الكلام على [Varible, object, على اي شيء] إذا تعرف داخل الـ scope إذا محدد فيه

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لنطبع الـ [output] ooo

1) يوجد [Global object] ⇐ [line 23] اكمل [obj] اسمه من T0 من اسمه test class
هة استدعي الـ [Constructor] ⇐ [line 7] اكمل initialization ثم اطبع: كلمة Count

T0 (Global)

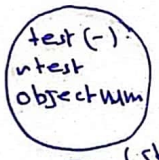
→ object with ID 0 created

2) كما يوجد بزيه Global هة لزوج من main [line 26] و تنشئ obj اسمه من T1
[line 7] Constructor و تستدعي الـ static لونه test class اسمه

T1 (static)

→ object with ID 10 created

3) For loop ⇐ [line 27] من (i=1 الـ i=2) بكل لفة لزوج
obj [T2] لكن لونه static ⇐ [line 18] ⇐ [i=1] و تنشئ Fun [create ob] من [line 18]
[تنشئ مرة واحدة فقط بغض النظر عن عدد مرات الزيادة] أي ان مع انشاء Fun الـ Scope ليبقى موجود.



T2 (static)

→ object with ID 20 created
لنفس اللفة ⇐ تنشئ T3 الـ ⇐ لياوي عدد اللفة اي ان [i=1]

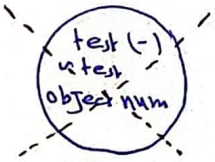
→ object with ID 1 created

T3

مع انعدام الـ Scope ⇐ ينظف T3 ويبقى T2 هة ⇐ تستدعي الـ destructor للـ
[line 13] T3

→ object with ID 1 destroyed

[i=2] ⇐ [line 18] ⇐ T3 تنشئ T3 اما T3 لبرهما موجودة هة ⇐ تستدعي
الـ Constructor للـ T3



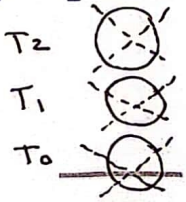
T3

→ object with ID 2 created

طلعنا من scope هة ⇐ T3 تنظف الـ destructor الى ⇐ تستدعي

→ object with ID 2 destroyed

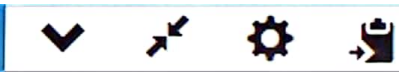
4) خلصنا من main بالة فيا بالكود [destructor] لبهين اعرف الترتيبة مثان ترتيب الحذف باين موجود
عنا T0 ⇐ T0 و T1 و T2 [T0 اول واحد يتم انشاءه هة آخر واحد يحذف ثم T1 ثم T2 هة الحذف في العكس]



→ object with ID 20 destroyed
→ object with ID 10 destroyed
→ object with ID 0 destroyed

الانشاء ←
الحذف →

الممسوحة ضوئيا بـ CamScanner



input

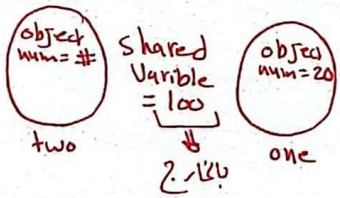
```
object with ID 0 created
object with ID 10 created
object with ID 20 created
object with ID 1 created
object with ID 1 destroyed
object with ID 2 created
object with ID 2 destroyed
object with ID 20 destroyed
object with ID 10 destroyed
object with ID 0 destroyed


...Program finished with exit code 0
Press ENTER to exit console.
```

الممسوحة ضوئيا بـ CamScanner

Set 2 : 5

* Varible ⇐ [data member]
* [Static Fun.] ⇐ بتشتغل على اي اشي [Static] حصراً فقط
Ex] [line 13] بتشتغل فقط مع [line 18] اما (Function) يعزه ما بتشتغل
* الفرقا بين Fun.[line 8] و Fun[line 13] :
Fun[line 13] يلي بحطله static زي كائن بحكيله حماي Fun
Fun.[line 8] بعدن [Static] هي بقدر
[Static] نوعا Fun. هي [Static data member] مشان اكتب
[Static data member] كمان يكون في
[Static] يشتغل على يلي بدل ايا ه
على [Static] و [non static]
نتعامل فقط مع
* [line 7] : member ← [member] تكون Fun. حالك
private و public بالجهتين ال
لوكانت [non-member] بحل عليها انه ما رح يكون قدامها dot (.) إذا private ما بقابل معه إذا public
بقدر تتعامل معه
[line 17] : عدد النسخ منه = عدد ال object ← [initilization بعمل Constructor ال]
[line 18] : بكتبه برا ال object ←

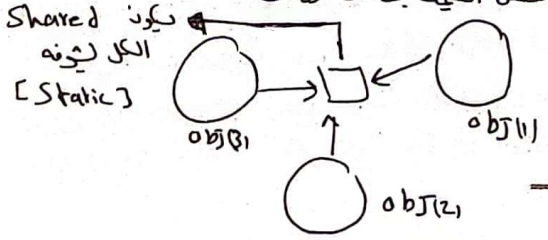
Ex)



* اي شي [static] بنعمل مرة وحدة لو مارة على السطر مية مرة
* ال Shared Varibk بكتبه برا ال object [تابع للكل]
تابع لـ class
لو بدي ازيد ال object num بكتب [one.objectnum++]
ليس static بكانه موجود نسخة ⇐ [اسم ال Varibk . اسم ال object]
لكل متغير انا لبسمي النسخة عن طريق ⬜ . ⬜ ← ال dot
لو كتبت two.Shared Varibk ← [X] غلط ، زي كانني دخلته جوا ال object
يلي لقدر لستخدم ال Shared varibk عا ال [object] ليس بدون ال dot (.)

Shared Varible
not belong object,
belong Class (تابع لـ class)

* متى بحتاج انه [member] متى Shared Varibk اكتبه static !؟
اذا بدي كل ال object يرجعوا النفس القيمة برا ال class



الممسوحة ضوئيا بـ CamScanner

استخدام ال [::]

1 _ :: Value [للاغراض في Global إذا فيه]
2 _ :: member [Class يلي بدك اياه من]

[obj ال member او ال Fun. ال] Class [للـ اسم ال ← يتبع ال member

لتتبع ال obj لكتب dot (.)

[Function member] لها الاستثناء ال [.] مشان [::] صطبة ال [line 28] ←

ال shared لا يتبع لـ [obj.] معين يتبع للكل ؛ لانه ال [dot] لتتخدم مع [obj]

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لطلع الـ [output] ooo

1 يروح Global البداية من main [line 22] نفذ obj اسمه Class من A اسمه

object num=0 ⇐ [line 6] Constructor ABC لتشريح

2 [line 23] ⇐ نفذ obj اسمه Class من AB اسمه ABC لتشريح

objectnum = 1 ⇐ [line 6] Constructor الـ

3 [line 24] ⇐ المبنى obj [A] لـ Fun [increase] ⇐ [line 7]

Shared var = 101 ← يلي بعد نفذ إنه تزيد الـ Shared لعتبار إن ooo

4 [line 25] ⇐ المبنى obj [A] لـ Fun [print] ⇐ [line 8]

→ object 0 with shared varible 101

5 [line 26] ⇐ المبنى obj [AB] لـ Fun [increase] ⇐ [line 7]

Shared var = 102 ← بعد نفذ إنه تزيد الـ Shared لعتبار إن ooo

6 [line 27] ⇐ المبنى obj [AB] لـ Fun [print] ⇐ [line 8]

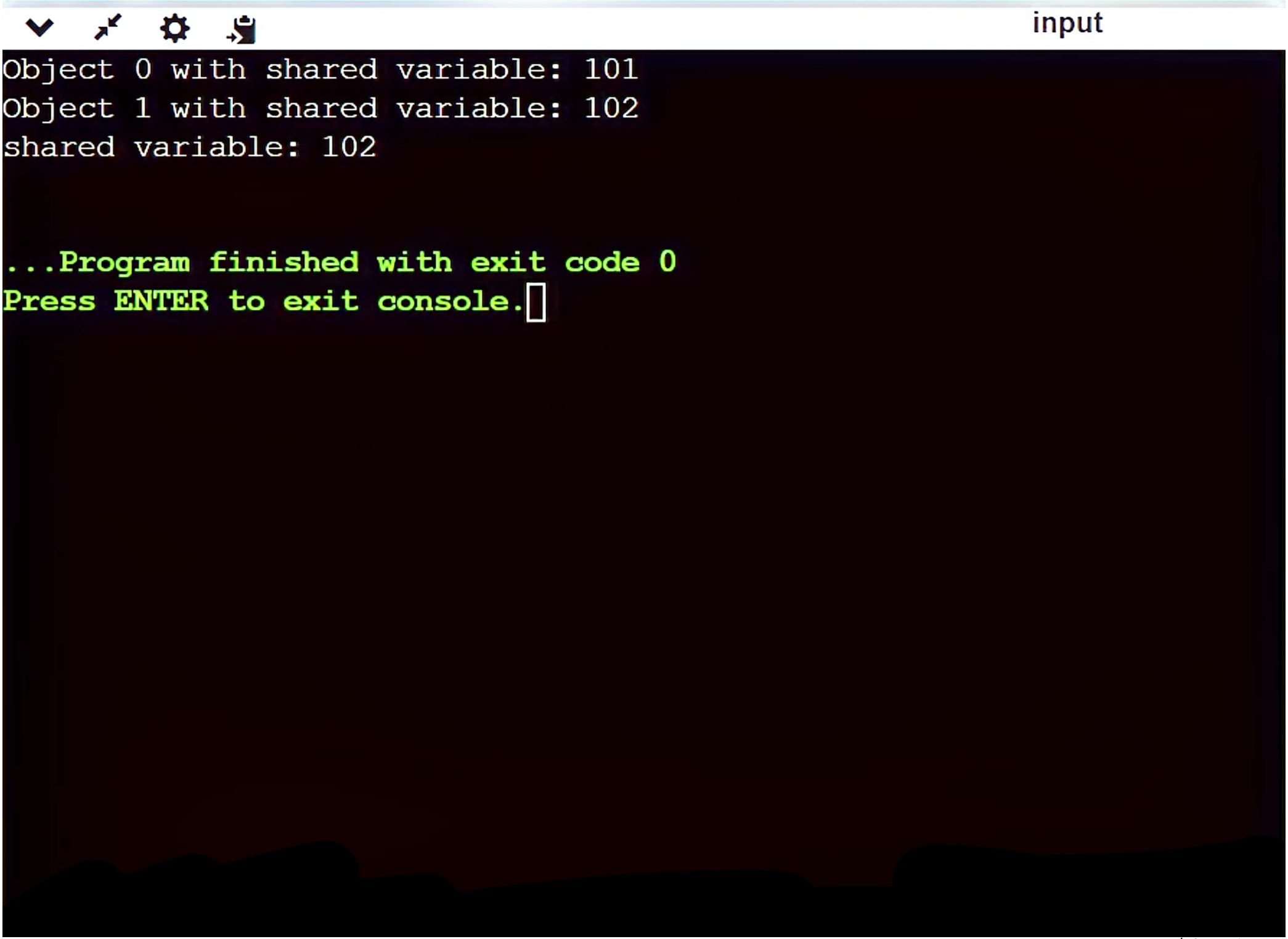
→ object 1 with shared varible 102

7 [line 28] ⇐ استدعى الـ Fun [Static print] من Class اسمه ABC ⇐ [line13]

→ Shared varible 102

يروح [destructor] ooo ما بهمني توضيح الـ [obj]

الممسوحة ضوئيا بـ CamScanner



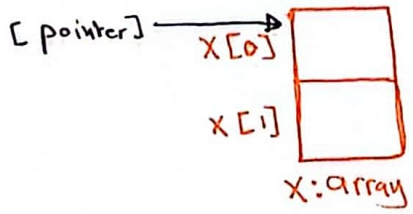
الممسوحة ضوئيا بـ CamScanner

```
/*************************************************************
// THIS POINTER AND FUNCTIONS RETURNING REFERENCE TO OBJECTS
*************************************************************/
#include <iostream>
using namespace std;

class A;
class B;

class B
{
int b;
public:
B(int y=33):b(y) {      }
      B& set_b( int x )
      {
      b=x;
      cout<<"class B \t"<<this -> b<<endl;
      return *this;
      }
      A& set_b( char c )
      {
      b=c;
      cout<<"class B \t"<<this -> b<<endl;;
      return (A&)*this;    }
      };

      class A
      {
      int a;
      public:
      A(int w=11):a(w) { }
      A& set_a(int c)
      {
      a=c;
      cout<<"class A \t"<<a<<endl;
      return *this;
      }
      B& set_a(char c)
      {     a=c;
      cout<<"class A \t"<<a<<endl;
      return (B &)*this;
      }
      };

      int main()
      {
      A a_object;
      B b_object;

      a_object.set_a(1).set_a(2).set_a(3);
      cout<<"**************"<<endl;

      a_object.set_a(10).set_a('2').set_b(3);
      cout<<"**************"<<endl;

      a_object.set_a('a').set_b('b').set_a('c').set_b('d').set_a('e');

      return 0;
      }
```


الممسوحة ضوئيا بـ CamScanner

Set 3 : 1

* لو بدي أعمل [Array]



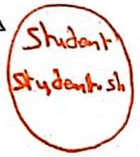
داخل الـ memory الـ [location] عادة انا عملته، أما الـ Compiler بشكل أوتوماتيك بعمل pointer على أول [location] كل أول
← اسمه الـ pointer نفس اسم الـ array وبقدر استخدمه لبس بدون ما تحركه

(X) * ← بترجع قيمة الموقع الأول
(X+1) * ← بترجع قيمة الموقع الثاني

* مين الـ [object] منيه فكرة بتشبه هاي الفكرة بصير الـ [object] ← pointer ليس [this]

← عدد نسخ الـ pointer this واحد فقط
[only one this] بأشير على object إذا رحت على object آخر
ح ينتقل الـ [this] للـ object الآخر

← pointer الـ [this] بوجد لفترة مؤقتة لما برجع تنفذ object داخل الـ Function



[this] بنقل الـ من object 1 للـ object 2

EX في [Student 1] obj. خاصة من Calling
ورجعت للـ main داخل الـ main بوجد pointer this [عنوان مؤقت]
← بيبدأ عنوان لما ترجع على [obj] وبدأ تنفذ Fun. معين

← [this] لقدر اعمل هالحجة سواء كان x public أو private لأنه لما تكون داخل [Function] مصطلح الـ public private ما بزعلهم.
الـ [this] تتخيل إنها داخلاً بتأشير [object] ليستفيد عملك إنما من البرمجة

* لو طلبت Fun. ترجع ← int

```
int Fun()
```

* لو طلبت Fun. ترجع ← Refrence to int

```
int & Fun()
```

(address of) ← لا ضافة إشارة ← int

* لو طلبت Fun. ترجع ← by Refrence + object

```
Student & Fun()
```

int ← اسم الـ class مكان الـ

return ← return this ← كورس تعلم

Student من نوع pointer

لما أعمل Return [دائماً وابدا لازم مالجد Return يكون متناسق مع نوع اسم Fun]

الممسوحة ضوئيا بـ CamScanner

* الاستفادة والاستخدام لـ this : لما تكون الـ Function ترجع [Refrence] لـ [object] مطلع او ممكن لقيدك [Pointer this]

[line 3 + line 4]: بعرف النوع سابقاً، فشان الـ Compiler ما يعترض للبلم [حقاً رح يجي تعريف الـ class]

* السرف والغاية فشان [line 16]

← لو حذفت [line 3+4] راح Compiler وصل لـ [line 16] رح يعترض على [A] لانها مش معرفة قبل فشان هيك عليها error ! عرفناها سابقاً

[line 10]: ترجع RefrenceToB [يعني ترجع [object] نوعه B و [By refarance]

* في main لما أنرن [object] ويجي وقت التنفيذ ويجي تنفذ الـ Fun اسمه Set-b [b.set-b] ترجع لـ هاد الـ scope من الـ class اوتوماتيك رح يعرف [pointer this] object يؤشر على obj كاملة.

[line 14]: رجع مؤشر على B [في تناسق في النوع] مع الـ Fun [B $Set-b]

[line 16]: داخل لـ class الـ B ورجعت [Refrence to A]

[line 20]: this* ($ A) ⇐ Convert to A ورجعها لـ A
↳ بتأثر على B لـ Fun الـ A

* لقد اعمل حركة [line 20] إذا بالضبط بشبهو لبعض تماماً [يعني الـ obj من [ClassB] بتحول لـ obj من [ClassA] ويصير فيه [match type] لقد اعملها اركة]

← كيف بشبهو لبعض ؟! ClassA فيها متغير واحد اسمه A و ClassB فيها متغيرين واحد اسمه B لو شلت حذفت من ClassA رح تختلف بكونه فيه [int a, int z] وفيه class [one int] و class [two int]

[line 31]: رجع مؤشر على A [في تناسب في النوع] مع الـ Fun [A $ set-a]

[line 36]: [Convert A pointer to B pointer to match type of Set a]

* Fun (int x , int $y)

↳ في main بأخذ قيمة الشيء ويعطيه لـ X [Calling by Value] بأخذ [object] بعطيه للمتغير أو بأخذ [object] قيمة واسماً بعطيه للمتغير حسب [Calling by Value OR Refrence]

↳ كل شيء بأخذه بعيد بعطيه لـ (Y) ويصير اشتغل على القيمة نفسها [Calling by Refrence]

[line 43]: خلص الترجع الـ [Fun] أكثر من مرة والـ object name ما في داعي ارجع اكتبه

الممسوحة ضوئيا بـ CamScanner

[line 43] ← توضيح التعبير عن السطر: aobject.seta(1).seta(2).seta(3); ⇐ line

A ⇒ A.seta(2).seta(3)
A ⇒ A.seta(3)
match

[تبسيط المعنى للفهم]

[line 45] ← توضيح التعبير عن السطر: aobject.seta(10).seta('2').setb(3); ⇐ line

A ⇒ A.seta('2').setb(3)
B ⇒ B.setb(3)
match

[line 47] ← توضيح التعبير عن السطر: ⇐ line

aobject.seta('a').setb('b').seta('c').setb('d').seta('e')

A ⇒ A.setb('b').seta('c').setb('d').seta('e')
B ⇒ B.seta('c').setb('d').seta('e')
A ⇒ A.setb('d').seta('e')
B ⇒ B.seta('e')

* لعمل Swap من Class لـ Class تانية
* لتستخدم [Pointer This] إذا حابب اعمل [two object] واعمل بينهم Swap

[line 47] ح تكون بـ (5) اسطر إذا حابب

aobject.seta('a') ← [Pointer This]
bobject.setb('b') ←
:

الـ [this Pointer] مكّنني اعمل [Calling Fun.] موجودين في Class مختلفة وعن طريق Fun الـ Return بترجع [Refrence to object]

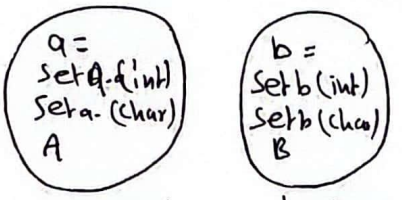
* [two Fun] من النوع setb ← يصلح لأنه واخد بوضح [int] والتاني [Char]
seta ←
[عشان يصلح لازم يختلف عدد أو نوع]

الممسوحة ضوئيا بـ CamScanner

... تتبع الكود لنظهر الـ [output] ...

1) يوجد Global object في البداية من main [line 39] ⇐ ثم ينفذ line 41 obj اسمه a-object

obj اسمه b-object ينفذ line 42 و A اسمه class من B اسمه class



2) [line 43] ⇐ الذهب الى obj [a-object] الى Fun [setA]

[line 27] (3) مرة (2) مرة (1) مرة ثلاث مرات وبكل مرة التقرير يضلف

a=1 العدد الذي | a=2 العدد الذي | a=3 العدد الذي ← التقرير اول مرة (1) بدل كل (1) ← نفوض

[int]

واطبع الحصيلة الوصيدة [line 30]

→ class A 1
→ class A 2
→ class A 3

****

3) [line 44] ⇐ اطبع الـ [*] واعمل [new line]

4) [line 45] ⇐ الذهب الى [a-object] الى Fun [setA نوعه int] ← يرجع A العدد الذي الى [setA نوعه char]

← يرجع B العدد الذي على [setb نوعه int] / line 27 ثم line 33 ثم line 10

→ class A 10 → التقرير كان 10 (int)
→ class A 50 → التقرير كان char (2) ⇒ AscII (2) = 50
→ class B 3 → التقرير كان 3 (int)

****

5) [line 46] ⇐ اطبع الـ [*] واعمل [new line] a_object

6) [line 47] ⇐ الذهب الى obj [a_object] الى setA نوعه char ← يرجع B العدد الذي الى

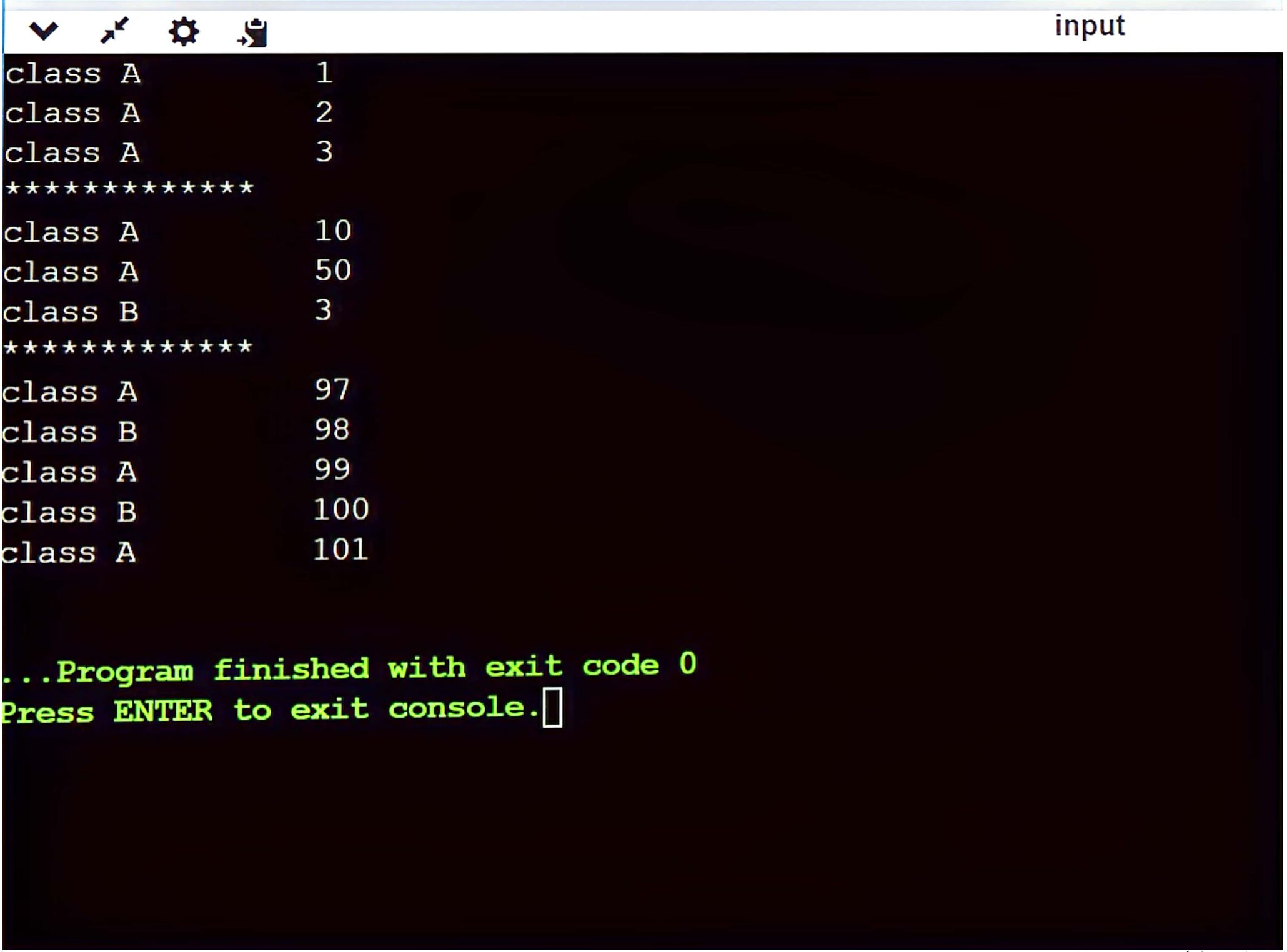
[setA نوعه char] ← ترجع A العدد الذي الى [setb نوعه char] ← ترجع B العدد الذي الى

[setA نوعه char] ← ترجع A العدد الذي الى [setb نوعه char]

line 33 ثم line 16 ثم line 33 ثم line 16 ثم line 33 ...

→ class A 97 → التقرير كان a (char) ⇒ AscII = 97
→ class B 98 → التقرير كان b (char) ⇒ AscII = 98
→ class A 99 → التقرير كان c (char) ⇒ AscII = 99
→ class B 100 → التقرير كان d (char) ⇒ AscII = 100
→ class A 101 → التقرير كان e (char) ⇒ AscII = 101

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

```
/*************************************************
//  COMPOSITION: SYNTAX, CONSTRUCTORS, DESTRUCTORS
*************************************************/
#include <iostream>
#include <new>
#include <cstdlib>
using namespace std;

class CPU
{
    string vender;
    long speed;
    int cores;
public:
    CPU(string v, long s, int c) :vender(v), speed(s), cores(c) {}
    CPU(const CPU& a) { vender = a.vender; speed = a.speed; cores = a.cores; cout <<
   "Copy constructor of CPU class\n"; }
    ~CPU()
    {
        cout << "Destructor for the processor " << vender << "\t" << speed << "\n";
    }
    void set_specs(char* v, long s, int c)
    {
        vender = v;
        speed = s;
        cores = c;
    }
    void get_cpu_spects()
    {
        cout << "CPU specs: " << "\n" << "Vender : " << vender << "\tSpeed : " << speed
   << "\tNumber of cores : " << cores << endl;
        cout << "**************\n";
    }
};
class MEMORY
{
    string technology;
    long capacity;
    bool multiple_slots;
public:
    MEMORY(string t, long c, bool m) :technology(t), capacity(c), multiple_slots(m) {}
    MEMORY(const MEMORY& a) { technology = a.technology; capacity = a.capacity;
   multiple_slots = a.multiple_slots; cout << "Copy constructor of MEMORY class\n"; }
    ~MEMORY()
    { cout << "Destructor for the memory " << technology << "\t" << capacity << "\n";
    }

    void set_specs(char* t, long c, bool m)
```


الممسوحة ضوئيا بـ CamScanner

```
42.     {
43.         technology = t;
44.         capacity = c;
45.         multiple_slots = m;    }
46.     void get_memory_specs()
47.     {
48.         cout << "memory specs: " << "\n" << "Tecgnology : " << technology << "\tCapacity
    : " << capacity << "\tMultiple slots ? : " << (multiple_slots ? "YES" : "NO") << endl;
49.         cout << "**************\n";
50.     }};
51. class IO
52. {
53.     bool mouse;
54.     bool keyboard;
55.     bool screen;
56. public:
57.     IO(bool m, bool k, bool s) :mouse(m), keyboard(k), screen(s) {}
58.     IO(const IO& a) { mouse = a.mouse; keyboard = a.keyboard; screen = a.screen; cout <<
    "Copy constructor of IO class\n"; }
59.     ~IO()
60.     { cout << "Destructor for the io " << endl;    }
61.     void set_specs(bool m, bool k, bool s)
62.     {    mouse = m;
63.         keyboard = k;
64.         screen = s;    }
65.     void get_io_specs()
66.     {
67.         cout << "IO specs: " << "\n" << "Mouse attached ? " << (mouse ? "YES" : "NO") <<
    "\tKeyboard attached ? " << (keyboard ? "YES" : "NO") << "\tScreen attached ? " <<
    (screen ? "YES" : "NO") << endl;
68.         cout << "**************\n";
69.     }
70. };
71. class computer {
72.     long cost;
73.     bool used;
74. public:
75.     IO io;
76.     MEMORY memory;
77.     CPU cpu;
78.     MEMORY* MEMpointer;
79.     computer(long c, CPU& cp, MEMORY& m, bool u, IO& i) :cost(c), cpu(cp), io(i),
    used(u), memory(m) { MEMpointer = &m;cout << "Constructor of container is being invoked
    \n"; }
80.     ~computer() { cout << "Destructor of container is being invoked \n"; }
81. };
82. int main()
83. {
```

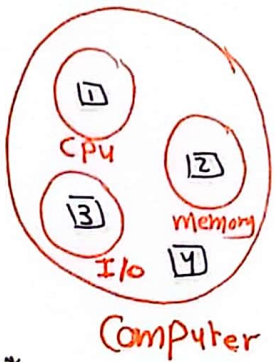

الممسوحة ضوئيا بـ CamScanner

```
84.     CPU c1("Intel", 2400, 4);
85.     CPU c2("AMD", 2100, 2);
86.
87.     MEMORY m1("DDR 2", 4000, true);
88.     MEMORY m2("DDR 2", 2000, false);
89.
90.     IO i1(true, true, true);
91.     IO i2(true, true, false);
92.
93.     computer pc(500, c1, m2, 0, i2);
94.     computer* p = &pc;
95.
96.     p->cpu.get_cpu_spects();
97.     p->MEMpointer->get_memory_specs();//p->memory.get_memory_specs();
98.     pc.io.get_io_specs();
99.
100.          return 0;
101.     }
```


الممسوحة ضوئيا بـ CamScanner

Set 3 : 2

* Composition : التركيب [برنا نركب object من object آخر]



← مثلاً ابي اجمع حاجتين ، لازم اعمل :
1. cpu class
2. memory class
3. I/O class
4. Computer class

كملت الـ [1,2,3] مثلاً اعمل [class 4] وادخل [1,2,3] براخله

* الـ object الكبير [Computer] جهاز Compose الـ obj الصغيرة الداخلة [cpu \ memory \ I/O]

المهم ابني [big class] براخله [small classes]

* من line 6 الى line 29 خاص في [class cpu] ⇐ ليتضمن الـ Constructor لها أعمل واحد صفر (أول مرة) وليتضمن الـ Copy Constru. لها اسحب الصفر واحطه داخل الكبير ليستخدم.

← [line 18 : Set : حطيت وسيط [public] بعكّين بغير العيتم لوحبيت الـ [Vender/speed/cores]
← [line 24 : نفس الـ set [وسيط public]

* من line 30 الى line 50 خاص في [class memory] ⇐ ليتضمن الـ Constructor لها أعمل [obj] داخل main
دال [Copy Constru.] لها انسخ معلوماته الـ [obj] الكبير ليتضمنه

دال destructor ليتضمنه سواء للصفر لوحده داخل الـ main او اللي داخل الـ [obj] الكبير [ادمره] .

* ليش ليتضمن الـ set ، get ؟! لأنه الـ data member [private] في من الـ main لا! صلاحية اوصلهم بعمل Access عليهم بالتالي بجيب وسيط من [get / set] وحطهم

← الـ [Fun] فيها set مصاوحة تعمل [initialization] للـ [data member]
← الـ [Fun] فيها get برجع تجيب قيمهم

* من line 51 الى line 70 خاص في [class IO]
* من line 71 الى line 81 خاص في [class computer]

← module الـ Computer براخله [another module]

← [line 75 : مصاولة [class computer] فيه object اسمه [IO] جهاز الـ obj من [class IO]
← [line 76 : مصاولة [class computer] فيه object اسمه [memory] جهاز الـ obj من [class MEMORY]
← [line 77 : مصاولة [class computer] فيه object اسمه [cpu] جهاز الـ obj من [class CPU]

الممسوحة ضوئيا بـ CamScanner

[line 79] : الـ order منه لن تبدأ عملية الـ [initilization] بـ long
وإنما من [ob] الصغار الموجودين داخل الـ [ob] الكبير حسب الترتيب الموجود من [line [77+ 76 + 75]
[line] ↑ حسب الترتيب بنفس الـ [Copy Constructor] لتبدأ الـ (cpu) ثم (memory) ثم (io) أول شيء الـ
ليس من line [79] والـ destructor يكون لعكس ترتيب الـ Constructor

[line 78] : [class Computer] داخل الـ [class memory] بأشر على Pointer

* ب [line 79] نعمل [initilization] لـ 5 متغيرات من [line 72 الـ line 77]
والـ [Cost / used] يصريهم [initilization] بالطريقة التقليدية كونهم Varible نوعه [long] يرجع لـ Cost
نوعه [bool] يرجع لـ used
باقي الـ object مابزبط اعملهم [initilization] لقيمة
الغرض انه الـ [Constructor] لتستلم [ob] اسمه من class (IO) وعداد الـ object ليتخدم مكان تقول فيه [initilization]
لـ اذا بالتالي حتة لحاتقل لتتضاف الـ class من obj (IO) مفعول سابقاً لتتضاف [initilization io] لـ Copy Constructor
نفس الشيء memory والـ cpu الـ object الغرض يكون مفعول سابقاً لتتضاف [intilize old] لـ [intilize new object]
[intilize new ob] تستخدم الـ class cpu لـ [old obj] الغرض له cpu والـ [new object memory within class Computer]
(within new object)

[new object cpu within new object Comp.]

← اول تغير ح يصير [1] الوبدي ادخل الـ Cost على انه غرض انه public من main ⇐ [Computer . Cost]
[2] لو كان داخل mem. technology] إذا الـ [public] كلها memory [public] technology
وبحاول اوصلها من main ⇐
[Computer . memory . technology]
↓
ربطية تنكتب بـ [.] لانه الـ Fun. يرجع object او يرجع class

الفرق عن الـ Code السابق إنه خاص فيه this
لـ إنه الـ this يبين كل مراحل صوت Composition

الممسوحة ضوئيا بـ CamScanner

ooo تتبع الكود لنطلع الـ [output] ooo

1 يروح [Global] من البداية main ⇐ [line 84] ⇐ انشيئ obj اسمه c1 من class اسمه cpu
وهو رح استدعي الـ Constru. الكوبي cpu [line 12] ما بطلب مني جزء طباعة فقط اعمل (initilization)
✓ [4 = Cores / 2400 = Speed / Intel = Vender]

2 [line 85] ⇐ اعمل obj اسمه c2 من class اسمه cpu وهو لنستدعي الـ Constructor [line 12]
✓ [2 = Cores / 2100 = Speed / AMD = Vender]

3 [line 87] ⇐ اعمل obj اسمه m1 من class اسمه MEMORY لنستدعي الـ Constructor: [line 36]
✓ [(1) true = multiple slots(m) / 4000 = Capacity / DDR2 = technology]

4 [line 88] ⇐ اعمل obj اسمه m2 من class اسمه MEMORY وهو لنستدعي الـ Constructor [line 36]
✓ [(0) False = multiple slots(m) / 2000 = Capacity / DDR2 = technology]

5 [line 90] ⇐ اعمل obj اسمه i1 من class اسمه IO وهو لنستدعي الـ Constructor [line 57]
✓ [(1) true = Screen / (1) true = Keybord / (1) true = mouse]

6 [line 91] ⇐ اعمل obj اسمه i2 من class اسمه IO وهو لنستدعي الـ Constructor [line 57]
✓ [(0) False = Screen / (1) true = Keyboard / (1) true = mouse]

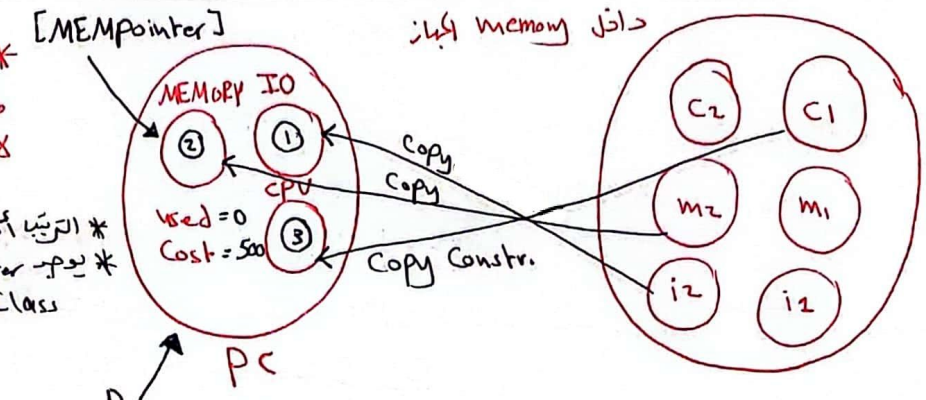
7 [line 93] ⇐ اعمل obj اسمه pc من class اسمه Computer وهو لنستدعي الـ Constructor [لـ memory]
لكن عين الترتيب ؟! حسب lines [75 ثم 76 ثم 77] جدول لنستدعيهم الـ Copy Constructor كلنه من الاول جاهزين

بدي اعمل الـ obj العميقة اكيد جاهزة لـ obj الكبيرة الـ Computer لعمل محتويات [c1 والـ m2 والـ i2]

→ Copy Constructor of IO class
→ Copy Constructor of MEMORY class
→ Copy Constructor of CPU Class
→ Constructor of Container is being invoked

1 ⇐ [line 58] i2 لـ Copy Const.
2 ⇐ [line 37] m2 لـ Copy Const.
3 ⇐ [line 13] c1 لـ Copy Const.
4 ⇐ [line 79] pc لـ Constructor

الممسوحة ضوئيا بـ CamScanner



* من [line 84] لـ [line 91]
طالما طرقنا لفكرة الـ Composition
لانه مارضنا على obj الكبير [Computer] pc

* الترتيب أو ضروري line [75 - 77]
* يوجد pointer اسمه [MEMPointer] بأشر على [MEMORY] Class

[8] [line 94] ⇐ اعمل pointer اسمه [p] من نوع Class اسمه Computer وخليه بأشر على pc
* صار عنا [two pointer] ← خارجي بأشر على الـ obj الكبير [pc] ← داخلي اسمه [MEMPointer] بأشر فقط على [m2 obj] MEMORY الـ Class

[9] [line 96] ⇐ اعتبر الـ pointer لـ Fun. [get cpu specs] الموجودين Class الـ [line 24] Cpu

→ Cpu specs:
Vender: Intel Speed: 2400 Number of Cores: 4
****

[10] [line 97] ⇐ خلي الـ Pointer (P) بأشر على Pointer [MEMPointer] لتستخدم الـ Fun [memory get specs]

→ [line 46]
→ memory specs:
Tecghology: DDR2 Capacity: 2000 Multiple slots ? : No
****

[11] [line 98] ⇐ استخدم الـ pc الـ obj من الـ obj (io) ثم استخدم منه الـ Fun [get io specs]

→ [line 65]
→ IO Specs:
Mouse attached? Yes Keyboard attached ? Yes Screen attached ? No
****

* طريقتين لاوصل لـ get ← [line 96] pointer خارجي لاوصل لـ obj الداخلي ثم الـ Fun
[line 98] لاسم خارجي لاوصل لاسم الداخلي لاوصل لـ Fun

الممسوحة ضوئيا بـ CamScanner

12| لما انا مش عارف بالكود [Destructor] ده ليه بتوضحلنا ان شاء الله obj

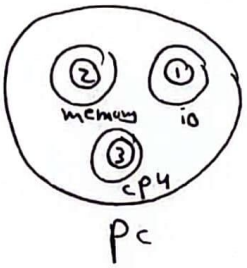
اول obj انشئ آخر obj يتحذف ده c1 اول وحدة انشأت آخر وحدة تتحذف

آخر obj انشئ اول obj يتحذف ده pc اول وحدة ح تتحذف لانه آخر وحدة انشأت

* الـ pc بداخلها obj كيف طريقة حذفهم ؟! نفس الآلية اول واحد انشئ آخر واحد يتحذف

ح نشاد ار2 ح ده الخط 3، 2، 1

ده اول اشي يتحذف الـ pc ثم الـ obj الصغار بداخله والترتيب 1، 2، 3



pc

[line 80] pc destructor 1
[line 14] c1 destructor 2
[line 38] m2 destructor 3
[line 59] i2 destructor 4

→ Destructor of container is being invoked
→ Destructor for the processor Intel 2400
→ Destructor for the memory DDR2 2000
→ Destructor for the io

13 خلصانين pc والـ obj الصغار بداخله باقي الـ obj من [line 84] الـ line [91]

[line 59] i2 destructor 1
[line 59] i2 destructor 2
[line 38] m2 destructor 3
[line 38] m1 destructor 4
[line 14] c2 destructor 5
[line 14] c1 destructor 6

→ Destructor for the io
→ Destructor for the io
→ Destructor for the memory DDR2 2000
→ Destructor for the memory DDR2 4000
→ Destructor for the processor AMD 2100
→ Destructor for the processor Intel 2400

* وهيك بتكون الـ memory فاضية من اي obj مع نهاية البرنامج

الممسوحة ضوئيا بـ CamScanner

input

```
Copy constructor of IO class
Copy constructor of MEMORY class
Copy constructor of CPU class
Constructor of container is being invoked
CPU specs:
Vender : Intel  Speed : 2400    Number of cores : 4
***************
memory specs:
Tecgnology : DDR 2      Capacity : 2000 Multiple slots ? : NO
***************
IO specs:
Mouse attached ? YES    Keyboard attached ? YES Screen attached ? NO
***************
Destructor of container is being invoked
Destructor for the processor Intel      2400
Destructor for the memory DDR 2 2000
Destructor for the io
Destructor for the io
Destructor for the io
Destructor for the memory DDR 2 2000
Destructor for the memory DDR 2 4000
Destructor for the processor AMD        2100
Destructor for the processor Intel      2400


...Program finished with exit code 0
Press ENTER to exit console.
```

الممسوحة ضوئيا بـ CamScanner

Operator Overloading

- Operators are the building blocks for expressions, thus they are essential part in any high level programming language.
- Usually, operators work on the built in data types.

  For example, + operator can add integers, floats, Booleans but can NOT add objects of classes

  Also, > can compare characters, doubles, integers but can NOT compare objects of classes

- How can the programmer extend the capabilities of the operators to operate on objects??
  ⇨ This could be done by operator overloading

- Operator Overloading is a mechanism that helps the programmer to let the operators do operations that do not exist on their original definitions. (like extending the capabilities of + operator to add objects)

- General guidelines and precautions that must be taken into consideration when overloading an operator:

  1. When an operator is overloaded, none of its original meaning is lost , instead the capabilities of the operator is extended.
  2. Operator overloading is done using special function called **operator#**. Where # is the operator that we wish to overload (ex. operator+ can extend to meaning of the + operator)
  3. The **operator#** function can be a- member function b- friend function c- global function that is non-member non-friend function
  4. You can NOT create new operators; rather you can overload the existing ones.
  5. Operator is not overloaded implicitly, if you overload the + and the = operator , this do not mean that the operator += is already overloaded.
  6. When an operator is overloaded, it can work with classes but the **precedence**, **associativity**, **arity** could NOT get changed.
  7. Four operators can NOT be overloaded are: **. .* :: ?:**
  8. = () and [] operators can NOT be overloaded using friend function.
  9. << and >> must overloaded as friend function.

الممسوحة ضوئيا بـ CamScanner

Set 4 :

* operator : عبارة عن [symbol] يلي بعمل عملية الـ [arithmetic] أو [logical] أو [Certain Action] أي أن رمز يحمل بين طياته معنى في البرمجة

* يمكن أن يستخدم الـ operator مع: expression، معين Function، placet of array [ ]

* أنواع data type:
  - [string / bool / char / float / int]: built in
  - [object]: user defined data type

← الـ operator الـ Symbol لتشتغلوا مع [built in data type] هو معناها ماني داعي لكل أي [Action] أما مع [object] مابعرفلوش لتشتغلوا

Ex الـ [+] بتقدر تجمع بين الـ [integers] والـ [floats] والـ [Booleans] لكن ما بتقدر تجمع الـ object الموجودة في Class ونفس الشي لكل العمليات الرياضية [- / * / < / > / ...]

* all of the operator can work on the built in data type , but not work on the user defined data type

من خلال الـ [operator overloading] بجعل الـ operator يقبل يتعامل مع: built in، user defined

* operator overloading : عبارة عن تكنيك يُضاف على الكود في حال كان عندك [object] والهدف منه تساعد المبرمج ليتخدم الـ [operator] مثلاً ليطبقهم على الـ [object]

* General guidelines and precautions that must taken into consideration when overloading an operator :

1. نضيف [operator] معنى ضفناه علشو آلية التعامل مع [user defined data type] بالمقابل ما فقدت قدرتها في التعامل كما سبق [مثل المقارنة]

← أي ان عبارة عن [extension] يعني بتضيف لقدرات الـ [operator] بدون ما تاخد منها

الممسوحة ضوئيا بـ CamScanner

2 يحدث ذلك من خلال [operator #] الـ Function الذي يسمى [operator] والـ [#]
ليتبدل من [operator] بليبل تقله [overloading]
Ex: [operator +] ≡ [+ operator] "تقوم بعملية الجمع بين [object]"

3 مواقع تواجد الـ [operator #] Fun.
1 member function [داخل الـ class]
2 Friend function [برا الـ class بس تابعة لـ class وكمان بداخل الـ class]
3 Global function [تكتب برا ماتكون member او Friend]

4 ما نقدر نعمل operator جديد [مثلاً اشارة الـ @] مش معروفة في [C++] هو ما نقدر استخدمها مع [object] أي ان الـ symbol الموجودة في [C++] مع [built in data type] نفس الشيء تشغيلهم مع [user defined data type]
يعني ألتزم في operator الموجودة والمعروفة معناها

5 بعض الـ [operator] متشابهين في الأداء مثل الـ [+ / ++ / +=] كلهم عبارة عن عملية الجمع
إذا عملت overloading للـ [+] لا يعني ان الـ Compiler عرّف كمان لتشتغل مع الـ [++] والـ [+=]
هو أي operator مش ممكن استخدمها في البرنامج الخاص في class تجرك بس ما احط على اطرافنا [object] لازم تكون معرفها او حاط الـ Function تبعها لتعريف العبارة

6 الـ overloading ما يغير في الأولوية، أي ان في اشياء ما نقدر نغيرها مثل [arity / precedence / associativity] "طريقة التنفيذ واتجاهية التجمعية"

7 الـ [. / .* / :: / ?:] ما نقدر نعملهم overloading

8 الـ [= / () / []] ما يربطوا بكونوا مع Friend [مهم]

9 الـ [>> / <<] إجباري يكونوا مع الـ Friend [مهم]
لـ cout لـ cin

* إذا بدنا الـ class لازم نرجع نعمل [operator overloading] كأنه الـ [Compiler] عرّف لشغلها على الـ class للأول أنا التعين كذا

الممسوحة ضوئيا بـ CamScanner

Ex]

```
class point
{
public:
int x,y,z;
point() { x=1; y=2; z=3; }  ⇐ Constructor non parameter
};

void main()
{
 point one;
 point two;
 point three = one + two;
}
```

two object
[two]x مع [one]x قيمه يجمع بين
[two]y مع [one]y و
[two]z مع [one]z و

لكن لا يعطي خطيا error ! لانه الـ Compiler مابعرف
لحظها اي طارح تقدر تعملها

لو استبدلت الـ [+] بـ أي operator من الـ [- / أو x / أو > / أو <] لا تعطي error !
فنا حتاج الـ Overloading لا تسمح لنا العملية بانه نغير و تسمح لـ operator لتغير
مع الـ classes

لو استبدلت الـ one+two بـ 6+5 لا ترجع الـ error ! [لانه المكتبة مش فيا استخدام الـ operator]
المكتبة فيا [data type] المتغيرات يلي فيهم العملية [المكتبة فيا] الـ class أو الـ object عشان يعرف شحال الـ [operator]

لو بدلت وكتبت
Cout << one; ⇐ لا تعطي error ! لانه الـ [<<] لا يعرف الـ operator
كيف يتم عملية الطباعة [لانه الـ operator مابعرف يتعامل مع object]

Cout << "one"; ⇐ طارح تعطي error ! [لا تقدر تتعامل معها لانها [String] Built in data type]

الممسوحة ضوئيا بـ CamScanner

```
/**********************************
//Operator Overloading : + operator
**********************************/
1. #include <iostream>
2. using std::cout;
3. using std::endl;
4. using std::cin;

5. class class_3d{

6. int x,y,z;
7. public:
8. class_3d(){x=y=z=0;}
9. class_3d(int x1,int y1,int z1):x(x1),y(y1),z(z1){}

10.      class_3d operator+(class_3d operand1); // operator function is a member
   function
11.      void display() {cout<<x<<'\t'<<y<<'\t'<<z<<endl;}
12.      };

13.      class_3d class_3d::operator +(class_3d op1) // call to the overloaded
   operator
14.      {
15.      class_3d temp;
16.      temp.x = x + op1.x;// these statments
17.      temp.y = y + op1.y;// overload the meaning
18.      temp.z = z + op1.z;// of the addition operation
19.      return temp;
20.      }

21.      int main()
22.      {
23.      class_3d object_one(10,10,10);
24.      class_3d object_two(2,4,6);
25.      class_3d object_three(6,0,-2);
26.      class_3d object_four(5,4,3);

27.      class_3d result  =  object_one+object_two;
28.      result.display();

29.      class_3d result1 =  object_one+object_two+object_three+object_four;

30.      result1.display();

31.      return 0;
32.      }
```

Important Notes:

1- You can NOT omit temp. from lines 17,18 and 19 since this will change the meaning of the + operator because it takes two constant objects. That's why we add a temp object.
2- Having cascaded calls to operator+ in line 30 is possible because the operator+ function returns object of type class_3d (the temp object).
3- In line 30, we have 3 calls to operator+ function and a call to copy constructor.
4- You can implement the operator+ function as a function that returns reference (to allow cascaded calls). The alternative solution is shown next.

الممسوحة ضوئيا بـ CamScanner

Set 4 : 1

* اسم الـ Class [Class 3d]

[line 10] : Class_3d operator + (Class_3d operand)

by value (&) [داخل عملية ارجاع] — by value

موقع الـ Function [member] داخل الـ Class

الجديد في الموضوع أول ما هنشوف مناديك تعرف إنه في عملية جمع بين [object] و [object]

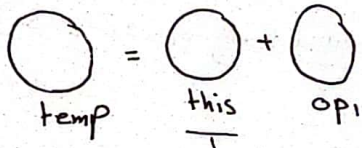
- بما إنه الـ [operator] داخل الـ Class فـ أكيد أحد الأطراف رح تكون من الـ Class نفسه

فكرة الـ [Copy Constructor]

New=old [خلال تنفيذ الـ] Copy Constructor نحنا بنكون داخل الـ new [الشمال]

هو بيسحب معلوماته من old [اليمين]

فكرة الـ [addition]



إذا كانت جاية الـ Fun. جوا من الـ Class بتكون زي كأنك منها

[this] بتكون [object] داخل اي [بتتخيل إنك جواته خلال عملية التنفيذ]

- عشان استقبل الـ [operator +] لازم object على طينه وشماله من الـ Class [obj (operator+) obj]

لكن في [line 10] ليس في [object واحد] وبيا الثاني ؟!

داخلاً وابدا إذا تعرف الـ [Function] يكون [member] الـ object الموجود على شماله [الشمال]

هو يلي بنادي الـ [operator] بالتالي بلزم تمرير الـ object الموجود على اليمين [old]

* الـ [operator +] بتوخذ ارجع بترجع للبرنامج هريلي بقول الـ Compiler ما بشرط عليه اشي ،

المنطقي إذا بدي ارجع [+] obj مع obj الطبيعي يرجع object [يلي برجع من استدعاء الـ Function

هو object من نفس النوع]

[line 13] : (Class_3d op1)

الطرف اليمين برر [old]

الممسوحة ضوئيا بـ CamScanner

← [line 16+17+18]: ما بحتاج لـ [overloading] كانه الـ [X والـ Y بالـ 2] مال [op] عبارة عن [int] دالـ Compiler بعرف يجمع int و int

* $X + op_{10} X$ ← الـ [X] التابع لـ object one [بتتغيل كائن داخل الـ object مباشرة الـ X مابرجعا كلاس قلها class name] كائن بتكون داخل [scop] اغا صغير obj الشمال

الـ [X] التابع لـ object two [اليمين بحاجة لـ Fun name]

← [line 27]:

result = obj one + obj two [كائن موجود من obj one]

استبدالا لـ Fun [operator+] / لو [line 13] مش موجود، كان الـ Compiler طارح بعرف يجمع بين الـ object

* لوكتبت [Copy constructor] طبعت كلمة Hi براضه

← [line 27] ⇐ رح يطبع 3 مرات الـ Hi [3 مرات لـ Calling Copy cons.]

1. نبعت obj two لـ op1 [الـ this] [new لـ]
2. لما النتيجة temp ترجع مكان [obj one + obj two] صارت return by value
3. لما انسخ الجواب لـ result

بالنه [Calling by value] بيصت صورة عنه بالتالي رح يرجع عل [Function] ما خدين صورة عن [obj one و obj two]

ولنجمع ويرجع لـ temp ولو بالغلط غيرت قيم x من [line 16] طارح عكس [obj one أو obj two]

* لو حذفت [line 10] + من [line 13 الـ line 20] رح يعطي عل [+] ÷ [+] [line 27] error!

و الـ [+] بـ [line 29] error! كانه بطل عارف مضلياً شو يعمل فيها الـ [+]

* لو بدي إياه طرح بغير الـ [+] لـ [-]

* عشان الغي [Copy constructor] بحط [by reference] عن [line 13] ⇐ $f_{class-3d}::(..f_{op})$

رح يعرفش الـ object يرجع، وامتى object جديد لـ object قديم رح يعرف [Copy constructor] ليستدعى مرة وحدة

لما يجي لنسخ الطرف اليمين كائن للطرف الشمال

الممسوحة ضوئيا بـ CamScanner

* بـ [line 27] بالرغم من انه two object النظام بيجمع x مع x و y مع y و z مع z و بيروح للناتج من نفس النوع x والـ y والـ z و نفس الشيء الـ [line 29]

result = object one + object two + object three + object four

op1 op1 op1

temp

[obj one و obj two بياخد الباقي] بيسلم ينفذ بالأول [temp يرجع]

temp + obj three

temp + obj four

[line 29] :

3 مرات بيستدعي [operator]

16 + 17 + 18 ح ترجع

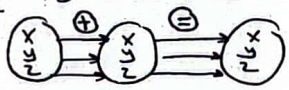
one + two + three x

one + two + three y

one + two + three z

نفس الشيء للباقي يعني temp + obj three

temp + obj four



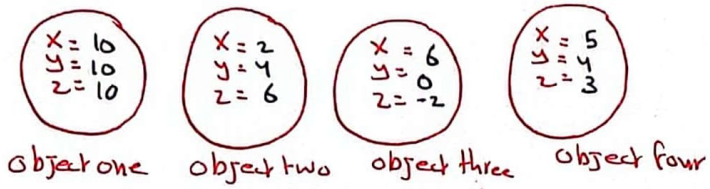
* ممكن أجمع بزيادة مؤشر جديد والـ Compiler ما يعرفش ليس المهم أعرفه كيف يجمع

* لو في [Copy Constr.] بـ [line 29] ح يتعمل 16 مرات

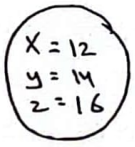
الممسوحة ضوئيا بـ CamScanner

... تتبع الكود لنطلع الـ [output] ...

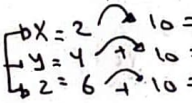
1) يوجد Global object في البداية من main [line 24+25+26] + [line 23] مطلوب منك نقل [4-object] [line 9] object كل مرة إنشاء الـ Constructor لتهيئة الـ [Class-3d] عند استدعاء الـ class اسم معين من نفس النمط
يوجد جمل طباعة لتطبيق فقط ["initialization"]



2) [line 27] ⇐ عبارة عن عملية جمع two object وتخزينها بـ result بالـ [operator+] ⇐ two object
[object one] - [object two] انت براحتك تكون op1 الطرف اليمين هو المرسل [line 13] بدأ امر لـ



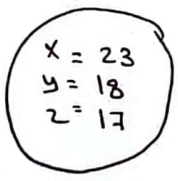
∴ 12 = 2+10, 14 = 4+10, 16 = 6+10 ← temp
الناتج بتخزن في object one

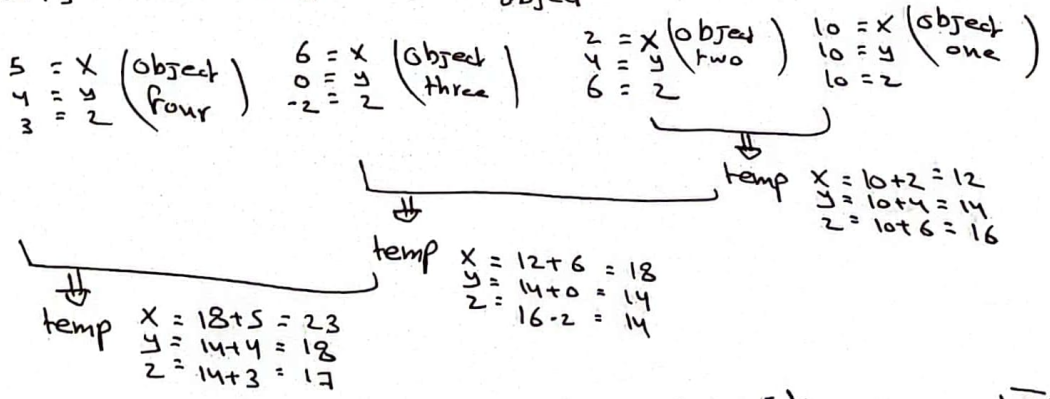


3) [line 28] ⇐ المهمة
[line 11] ⇐ [display] Fun ← object result
عبارة عن عملية طباعة للعنصر

→ 12 14 16

4) [line 29] ⇐ عبارة عن جمع four object وتخزينها بـ result ⇐ [operator+] [line13]

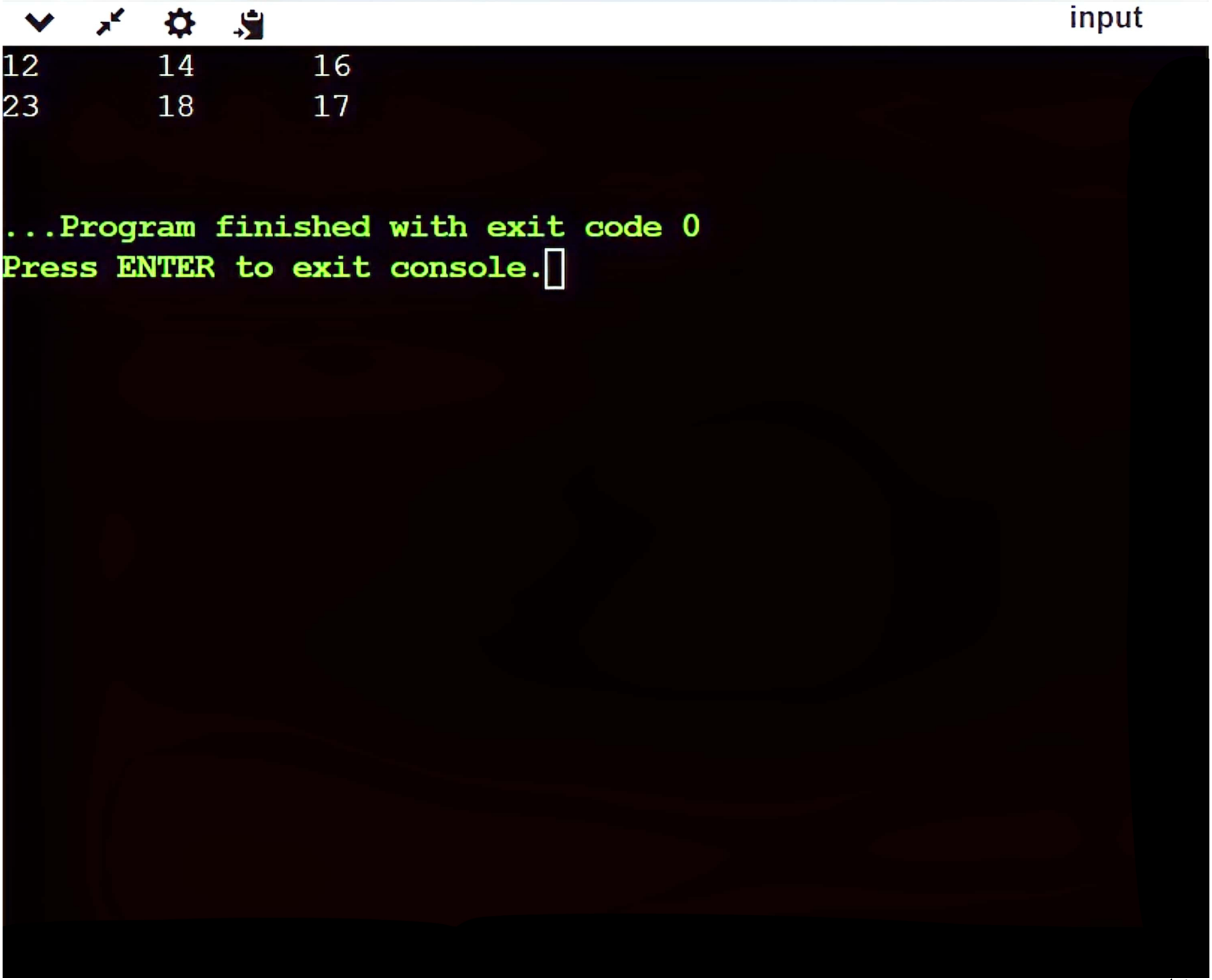




5) [line 30] ⇐ المهمة
[line 11] ⇐ [display] Fun ← object result

→ 23 18 17

الممسوحة ضوئيا بـ CamScanner



الممسوحة ضوئيا بـ CamScanner

```
/*************************************************************
//Operator Overloading : + operator : alternative solution
*************************************************************/
#include <iostream>
using std::cout;
using std::endl;
using std::cin;

class class_3d{

int x,y,z;
public:
class_3d(){x=y=z=0;}
class_3d(int x1,int y1,int z1):x(x1),y(y1),z(z1){}

      class_3d &operator+(class_3d operand1); // operator function is a member
   function
      void display() {cout<<x<<'\t'<<y<<'\t'<<z<<endl;}
      };

      class_3d temp; // must define the object as global to avoid returing
   reference to destroyed object
      class_3d& class_3d::operator +(class_3d op1) // call to the overloaded
   operator
      {

      temp.x = x + op1.x;// this statments
      temp.y = y + op1.y;// overload the meaning
      temp.z = z + op1.z;// of addition operation
      return temp;
      }

      int main()
      {
      class_3d object_one(10,10,10);
      class_3d object_two(2,4,6);
      class_3d object_three(6,0,-2);
      class_3d object_four(5,4,3);

      class_3d result  =  object_one+object_two;
      result.display();

      class_3d result1 =  object_one+object_two+object_three+object_four;

      result1.display();

      return 0;
      }
```

التعديل عن الكود السابق
هو إضافة ال &
بصير " calling by reference "

calling by reference

ال copy constructor
رح يستدعى مرتين ، قبل ٣ مرات
1) مرة لما أبعت لل op1
2) مرة لما أرجع ال result وأخزنها على الشمال

الممسوحة ضوئيا بـ CamScanner

```
/*****************************************
//Operator Overloading : complete Example
*****************************************/
1.  #include <iostream>
2.  #include<iomanip>
3.  using std::cout;
4.  using std::endl;
5.  using std::cin;
6.  using std::istream;
7.  using std::ostream;
8.  using std::setw;

9.  class mobile_num{

10.       int country_code,num_prefix,num_postfix;
11.       public:
12.       mobile_num(){country_code=num_prefix=num_postfix=0;}
13.       mobile_num(int x1,int y1,int
    z1):country_code(x1),num_prefix(y1),num_postfix(z1){}

14.       int operator[](int part); // overloading the [] operator as member function

15.       friend ostream &operator<<( ostream &output, const mobile_num &no ) //
    overloading the extraction operator
16.       {
17.       output << '+' << no.country_code << '-' << no.num_prefix<<'-
    '<<no.num_postfix;
18.       //to allow cascaded printing , we must return an object of ostream
19.       return output;
20.       }

21.       friend istream &operator>>( istream  &input, mobile_num &no ) // overloading
    the insertion operator
22.       {
23.       input.ignore(1);
24.       input>>setw(3)>>no.country_code; // friend function must access the data
    members as object_name.member_name

25.       input.ignore(1);
26.       input>>setw(2)>>no.num_prefix;
27.       input.ignore(1);
28.       input>>setw(3)>>no.num_postfix;

29.       return input;
30.       }

31.       mobile_num& operator=(const mobile_num Left_side)//must overload the =
    operator as member function
32.       {
33.       country_code = Left_side.country_code;
34.       num_prefix = Left_side.num_prefix;
35.       num_postfix = Left_side.num_postfix;

36.       return *this;
37.       }

38.       friend bool operator==(const mobile_num R,const mobile_num L);//as global
    function

39.       }; // End class
```


الممسوحة ضوئيا بـ CamScanner

```
40.      int mobile_num::operator[](int part)
41.      // must overload the [] operator as member function
42.      // member function access the data members directly without using
   //object_name.member_name
43.      {
44.      if(part > -1 && part < 3) // range checking
45.      {
46.      if(part==0)
47.         return country_code;
48.      else if(part==1)
49.         return num_prefix;
50.      else
51.         return num_postfix;
52.      }
53.      else // for incorrect index
54.      return country_code;
55.      }

56.      bool operator==(const mobile_num Right, const mobile_num Left )// this
   function //needs to access the private date
57.      //therefore we can define it as member or friend function
58.      {
59.      return
   (Right.country_code==Left.country_code&&Right.num_postfix==Left.num_postfix&&Right.
   num_prefix==Left.num_prefix);
60.      }

61.      bool operator!=(const mobile_num R, const mobile_num L)
62.      //this function calls the == operator and does not require to access the
   private data, so it is declared as global function
63.      {
64.      return !(R==L);
65.      }
66.      int main()
67.      {
68.      mobile_num mob1(962,79,6862314);
69.      mobile_num mob2(962,78,6462222);
70.      mobile_num mob3(962,77,6005005);

71.      cout<<mob2<<endl; // calls the overloaded operator<<

72.      mobile_num mob4; // calls the default constructor and initialize the number
   to // +0-0-0
73.      cin>>mob4;// calls the overloaded operator>>

74.      mob1 = mob2 = mob4; // calls the overloaded operator== twise
75.      //first call format mob2.operator+(mob4)
76.      //second call format mob1.operator+(mob2)

77.      cout<<mob4<<"\t"<<mob2<<"\t"<<mob1<<endl;

78.      cout<<mob1[0]<<"\t"<<mob3[1]<<"\t"<<mob4[3]<<endl; //calls the operator[]
   //function 3 times
79.      //first call mob1.operator[0]
80.      //second call mob3.operator[1]
81.      //third call mob4.operator[3]

82.      cout<<(mob1==mob3)<<endl;
83.      cout<<(mob1!=mob2)<<endl;

84. return 0;}
```


الممسوحة ضوئيا بـ CamScanner

Set 4 : 3

* [line 10]: عبارة عن [data member]
* [line 12]: عبارة عن [Constructur non-parameter]
* [line 13]: عبارة عن [Constructor parameter]
* [line 14]: عبارة عن [overloading] الـ [ ] اضافة في [array] كودها موجود في [line 40]
* [line 15]:

```
Friend ostream &operator << (ostream &output, const mobileNum &no);
```

- << : تستخدم في cout , بتخلي إننا unary لكن هي binary , بتكون دائماً في Friend
- output : اسم الشاشة المراد [black screen]
- no : متغير

[mobile num] class من [object] هعمل [prototype] ⇐ استخدم الشاشة المراد لطباعة معلومات الـ

تقسم عملية الطباعة : إعتبر إن الـ output اللي متغير من ostream عبارة عن box فاضي وانت قاعد بتحط المعلومات فيه بالترتيب وانت بتبعته للطباعة

- ostream : الشاشة تستخدم للإخراج

* [line 21]:

```
Friend istream &operator >> (istream &input , mobileNum &no)
```

- >> : تستخدم في cin , binary , بتكون دائماً في Friend
- istream : الشاشة تستخدم للإدخال

* الـ >> ⇐ بتوخد ⇐ من object الشاشة ⇐ وبين اطبع ⇐ هايش اطبع
* الـ << ⇐ منها وبين ⇐ وبين أودّيه ⇐ آخد الـ input
* لازم Friend ⇐ لـ آخد المعلومات من الشاشة واديها لـ object

* لو تغير السؤال في الامتحان يلزم تغير فقط ⇐ اسم الـ input/output ⇐ no
  ⇐ اسم الـ class ⇐ حسب السؤال
* [line 23]: ignore(1) (أهمل) ⇐ input الشاشة اللي جاية طنش اول حرف " اشارة الـ + "
* [line 24]: setw(3) : الرقم داخل setw بمثل اكد اد ايه اللي يلزمني آخده [upto 3digit] خد كدا ≤ 3
  ⇐ لو كان ≤ حروف بأخد (3) لو كان ≤ بأخد (2)
* [line 29]: return input ⇐ هايرجع الـ [object] كامل و يرجع فقط input و كانه زي كاني بحكيلو خد العتيك من الـ [object]

الممسوحة ضوئيا بـ CamScanner

* [line 31]: mobile num &operator= (const mobik num Right side)

[Right side] الـ obj بيرجع لنفس الـ class ، member داخل الـ
[left side] بتتخيل إنك بداخلة بكائن أخر لأن طرف موجود داخل الـ Fundtion

Copy constru. ← الفرق بينهما وبين الـ old(obj) = old(obj) [شرط الشمال جديد] ← ليتضمن لها يكون فيه

obj x
obj y
x = y ✗
لست Copy Constr. كائن x, y كونهم

obj y
obj x = y ✓
Copy Const. كونه متضمن جديد

= ← عبارة عن (binary operation)

مثال تقدر تشتغل لازم لطيها شغلتين
1. object من class [mobile num]
2. object من class [mobile num]

تقل لنقل معلومات من الـ [obj] الايمن إلى الـ [obj] الايسر

لها أعمل ما أراده المفروض يرجع object يلي على الشمال = object يلي على اليمين وكيف عرفنا ؟!

(X=(y=z)) ← تتضمن اليمين الشمال [y=z] والعيبة تنادي [X] يلي يرجع سواء من z او لا هي واحد
z تترجع لـ X

○ = ○ = ○ ← يلي يرجع هو object مافرق ايا واحد بكائنه اذا نقلنا معلومات اليمين الشمال صاروا

line 35 لـ 33 من [Return (&left sid)] ← زي بعض لسيل بقدر اكتبه
ماضي اختلاف بين Right ، left

* [line 36]: [&left side ≡ *this] نفس المعنى

* من line 33 لـ line 35 الطرف الشمال ، مكتوب بدون dot [.] بكائنه ابترحالك داخل الـ object موجود
مش بحاجة لـ [Fullname]

* [line 38]: Friend bool operator == (const mobile num R, const mobil num L)
الطرف الثاني — الطرف الأول
← لتتضمن المقارنة بين شغلتين [binary]
← Friend: [non member, atas member] خارج الـ class هي مجرد فيتين
← مصطلح الـ [non member] كأنك لست بداخل أي object [this] ما يوجد في الـ

* مع [Cut, past] & بدون * [Copy, past] & * this: لفصل الـ object الايمن والمعلق

Note * &left side: & ⇐ (Return by refrance) لو حطيتها الـ [&] ادق ، ما راح تأثر
لـ main [Cut, past] كاجراء ، يلي يفرق انه مكان وجود الـ [&] يمنع استدعاء الـ [Copy Constru.]

الممسوحة ضوئيا بـ CamScanner

* [line 40] : int mobilenum :: operator [ ] (int part)
لـ ح تغير "اختيارية" — لـ ح ترجع داخل الاقواس [ ] unary operator

[اسم class :: member] ← : اذا برّا الـ class وبدك تجيب member لـ class

سؤال في اله متحان : اعمل overloading لـ [ ] و ح تكتب مثل [line40] باستثناء اول كلمة [int]

كانه بتسقل نوع القيمة يلي بدك تجيبها ← char : بوخذ القيمة مبرد من ASCII ويرجع الحرف وليس الرقم
bool ← T : اذا بترجع مغزية
F : اذا القيمة صفر

[return data type] ← هي يلي بقدر اتحكم فيه فقط

* اذا بدك تعمل [operator overloading] مع class تانية و تتذكر انه بتوخذ one part نوعه [int] دائماً
بتغير الكود الـ [int part] ما بتغير نوعه كانه دائماً [index] مالـ index ← int ولـيس شيء آخر

* ايّ operator [آحادية او ثنائية او ثلاثية] ح يكون إما binary او unary و يعني بتوخذ قيمة او قيمتين او ثلاث
+/-/>/< ← binary [بدها قيمتين شان تشتغل]
[ ] ← unary [بتوخذ قيمة وحدة]
++/-- ← unary
F : T ? الشرط ← بتوخذ (3) اقيم لكنها لتستخدم ع [overloading] بسبب نقطة (.) اول set
هي ايّ operator بدنا نستخدم يا آحادية يا ثنائية

* [line 56] : الـ [= =] لما اقارن بين شغلتين اجباري ترجع bool ← T / F

* [line 59] : جميع المتغيرات وردت بعد (.) كانه non member [objname . member name]

[line 59] ← إذا كله [T] المفروض ترجع [T] كانه operator ترجع bool و اذا بترجع الـ bool لـ int ح يعطيني error! [لنقطة في أول set] كان لازم نلتزم في الاولويات ، آليه اكسباب ، اكاهية التجمعية في الناتج

ليش في [line 40] بقدر ابرز int ؟! كانه يلي برجعه دوماً [int] ، إذا array int ترجع int إذا float ترجع float و بالمقابل الـ operator (= =) ما عملوا استخدمه وشفت انه برجع int !
يلي برجعه ناتج مقارنه (T/F) ← bool اجبارية لتحافظ على الصورة العامة لـ operator

الممسوحة ضوئيا بـ CamScanner

* [line 61] : عكس الـ [==] و تنفي جواب الـ [==] يلي طلع هنا

Ex) mob1 = mob2

عبارة عن obj من Class [mobil num] يرجع لـ [line 56] النفس الحالة يرجع T/F

Ex) mob1 != mob3

يرجع لـ [line 61] و [line 64] يرجع لطلعين لفوق لـ [line 56]

كانه مضاما سيد استخدام [operator ==] واعكس الجواب

اياا نرح اعرف في [operator !=] عل جواب [operator ==] والعكس

* لو بدل تعمل [operator +] المجموع يكون الـ [prototype]

1. تحدد هي
   - non member [خارج الـ Class] يعطيه two parameter
   - member [داخل الـ Class] يعطيه one parameter

بنكار (+) و بنكار

Sol mobile num & operator + (mobilenum R , mobil num L)
Return R OR L

1. مع او بدون الـ [&] ما بتفرق غير في استخدام الـ [Copy Constr.]
2. يرجع object [اما R او L ، المفروض الجواب في L]

الممسوحة ضوئيا بـ CamScanner

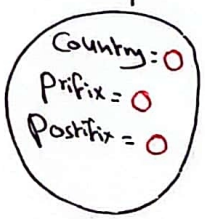
ooo تتبع الكود لنطلع الـ [output] ooo

1) يوجد Global object في البداية من main [line 68 - line 70] نقل [mobile num] class من object(3) وهي مخلقة في الـ Constr. param. [line 13] لتشغيل

2) [line 71] ⇐ يوجد طباعة object في [operator <<] لتشغيل [line 15] نقلت mob2 لأن عبارة عن جمل طباعة

→ +962-78-6462222

3) [line 72] ⇐ يوجد نقل [obj moby] من [mobile num] class لكن ما مر على [Constru. non param.] لتشغيل تم لكن [line 12] ⇐ يعمل initilization لكل [data member] بـ [zero]



4) [line 73] ⇐ يوجد نقل [obj moby] لـ [operator >>] [line 21] ⇐

→ +962
→ -55
→ -5486195

⇐ يعمل الـ + والـ - والـ . من ignore(1) كتبنا عدد خانة الدار

ويخزنهم بـ (obj moby)

5) [line 74] ⇐ يوجد الـ (=) يبعتي obj من ⇐ لتشغيل الـ [operator =] [line 31] ⇐ لنبدأ من الشمال للصين يعني [moby = mob2] ثم الفيفة نلغيها بـ mob1 و mob2 بنهاية الدالة ⇐ لنوصل moby بأنه mob2، mob1 ⇐ صاروا moby

6) [line 77] ⇐ اطبع الـ object في [operator <<] لتشغيل [line 15] ⇐ [moby mob2 mob1]

→ +962-55-5486195 +962-55-5486195 +962-55-5486195

7) [line 78] ⇐ لتشغيل الـ index في [operator []] [line 40] ⇐ لنفس الدوال أثر الدالة

mob1 ⇐ index 0
mob3 ⇐ index 1
moby ⇐ index 3

نختبر الشرط if(part>1 && part<3)

mob1 ⇐ [Country] ⇐ index 0 ⇐ يرجع
mob3 ⇐ [prifix] ⇐ index 1 ⇐ يرجع
moby ⇐ [Country] ⇐ index 3 ⇐ يرجع

else ⇐ جملة ⇐ لأنه ما تحقق الشرط

الـ mob3 ما تعدلت على obj اما الباقين تم التعديل عليه بأخذ آخر تعديل تعدلت

→ 962 77 962

962 ⇐ Country moby
77 ⇐ prifix mob3
962 ⇐ Country mob1

الممسوحة ضوئيا بـ CamScanner

8 [line 82] ⇐ لتنفيذ الـ [operator ==] دالة بحيث [two object] بحيث mob1 ، mob3
الـ [line 56] ، عالية بينهم ⇐ في اول حالة في شرط واحد اقل بعطي 0 (bool) zero
mob1 ، mob3 مختلفين ∴ 0 (غير متساويين)

→ 0

9 [line 83] ⇐ لتنفيذ الـ [operator !=] دالة بحيث [two object] بحيث mob1 ، mob2
اول اشي لـ [operator ==] وبنفي جوابه دالة (!=)
الـ mob1 ، mob2 متشابهين ∴ متساويين 1 ، 1 != 0 ∴ 0

→ 0

عالية هامين [destructor] ∴ مايهتم بحذف الـ object وتوقيتهم

الممسوحة ضوئيا بـ CamScanner

input

```
+962-78-6462222
+962
55
5486195
+962-55-5486195 +962-55-5486195 +962-55-5486195
962     77      962
0
0


...Program finished with exit code 0
Press ENTER to exit console.
```

الممسوحة ضوئيا بـ CamScanner

Inheritance

Base Class Data Member	Inheritance access control	Is inherited data from base Accessible inside the derived class?	Is inherited data from base Accessible inside the derived class?
Public	Public	✓	✓
Protected		✓	X
Private		X	X
Public	Protected	✓	X
Protected		✓	X
Private		X	X
Public	Private	✓	X
Protected		✓	X
Private		X	X

الممسوحة ضوئيا بـ CamScanner

Set 5

* Inheritance : التوريث ؛ بدي أورث خصائص واحد لواحد تاني من [Code useability]
الكود يلي نكتبه للأب رح يستفيد منه الابن ؛ يعني ما تعيد حالك وتكتب الشغلة أكثر من مرة

(1) Is inherited data from base Accessible inside the derived class

* من خارج الأب و الابن بدي أشوف متغيرات الابن
* وصف متغيرات الابن لما انتقي عليه من برا سواء [main] أو [Funs]
→ تنزلت public وكانت public : من الخارج public
→ كانت عند الأب protected اتضمنت في public ؛ ما بقدر من الخارج أشوفها
؛ تنزل protected : ما بقدر من الخارج أشوفها
→ كانت private حتى ممكن تتعرض للتعديل
→ وقت عند الأب public لما تنزلت عند الابن اتضمنت ب protected : ما بقدر من الخارج أشوفها
→ private في Inheritance بالأب شو ما كان إطلاقها تنزل private

(2) Is inherited data from base Accessible inside the derived Class

* بتفرج على الأب من الابن
* وصف متغيرات الأب لما يجي من الابن
→ الشي يلي بكون عند الأب Public or protected
ابنه بقدر يشوفه ويعدل عليه Access ؛ واذا private لو كان عند ابن ما بقدر يشوفه

(3) Inheritance access Control

* وضع المعلومات يلي تنزلت عند الابن لخارج الابن [من خلال اعرف شو عاملت المعلومة وعشان تم آخر المعلومة]

(4) Base Class Data member

* معلومات الأب العلوي
→ public : بتشاف للابن ولغيره
→ protected : ما داخل السلسلة فقط [المصطلح لفيبك فقط لما يكون عندك سلسلة]
protected [لمصانة للسلسلة public
[Fun /main] لخارج السلسلة private و
→ private : ما بشاف غير من الأب فقط
[لا الابن ولا غيره لقدر يشوف فقط الـ class]

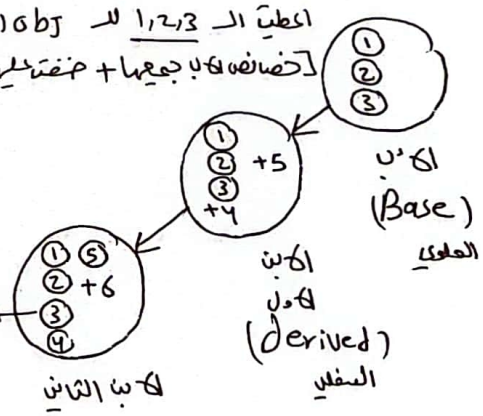
* كيف بعمل [Ex] Public :] من خلال protected او private او public
* الـ ['inheritance] ك code بـ يعراضعب في التبع ؛ لانه بدل
كتفرج على الأب الشي كان وظيفه للابن اشي وظيفه وتعالجهم من Funs أو main

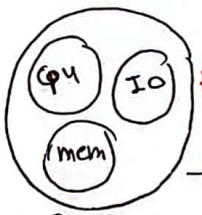
* لتجنب اكثر علاقة بين Inheritance / Composition

Composition
[has-a]
شكل الـ object كان دائرة كبيرة
Computer وجواتها دائر قد الـ
[ما يوجد توارث لكن يوجد تركيب]
Part has a part
Ex Computer has a mem.
Computer

Inheritance
[is-a]
شكل الـ inheritance خصائص موجودة
عند احدى الـ object او الـ class
حاب انت تعطيها لواحد تاني

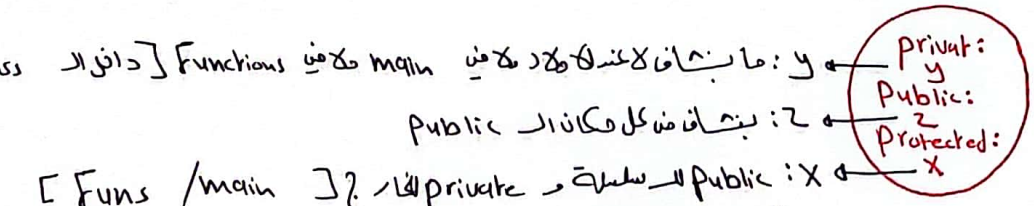
اعطيت الـ 1,2,3 للـ obj الثاني
[خصائص الأب جميعها + خصائصها]





→ داخل الـ class الكبيرة بعمل class للـ extension الغير
[بكتب الـ obj الصغار لبين obj الكبير]

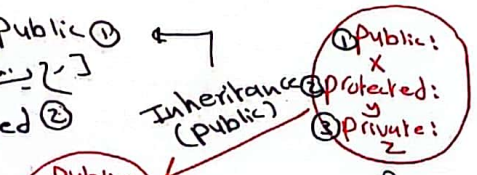
الممسوحة ضوئيا بـ CamScanner



y : ما بتشافش لا جوه ولا برا مكانش في main ولا في Functions [دا لي الـ class فقط]

z : بتشاف من كل مكان الـ public

X : الـ public بسلسلة و الـ private لا ؟ [Funs /main]
لا مكدش يقدروا يشوفوهما



Inheritance (Public)

① Public مع Public متساويين في الـ level هو بنزل public [بتشاف من main]

② Protected مع Protected ← public اقوى هو بنزل protected [بتشاف من الخارج class جوه الـ private خارج الـ]

③ Private مع public الـ private اقوى هو بنزل private [ما بتشافش لتحت الأب]

اخصائص ما بتنزل زي ما هي بتتعرض لعملية فلترة [عبروا كلهم من خلال كلمة public موجودة من الطريق]

⇓

بتسأل حالك مين الأقوى ؟!

protected اقوى من public بكذه مقيد أكتر من public

private اقوى من public هو protected بكذه مقيد أكتر من public والـ protected

الممسوحة ضوئيا بـ CamScanner