New
2018

اللجنة الأكاديمية للهندسة المدنية

دفتر

برمجة حاسوب

مروة حلوة

Contact us :

Civilittee HU | لجنة المدني

Civilittee Hashemite

www.civilittee-hu.com

★ Introduction :-

لبدء التعلم في البرمجة (software) يجب التعرف على كيفية انتقال البيانات داخل أجزاء الحاسوب.

★ الوحدات التابعة لحركات البيانات :-

CPU / memory / Hard Disk / Input/Output devices

- CPU ↓ لتنفيذ البيانات
- memory ↓ انتقال
- Hard Disk ↓ تخزين
- Input/Output devices ↓ وحدات الإدخال والإخراج

★ Processing a C++ Program :-

① Editor :- (المحرر)

* تقوم فيه عملية كتابة الـ code والتعديل عليه من زيادة أو نقصان. (في الـ HD).

② Preprocessor :- (المعالج)

* يبدأ عمل المعالج مع (#)
* وكأنه مكتبة كبيرة تحوي معنى الكلمات والاشارات المستخدمة في كتابة البرنامج ليفهم الحاسوب معنى ما يتم كتابته. (في الـ HD)

③ Compiler :- (المترجم)

* يعمل المترجم مهمتين رئيسيتين وهما :- (في الـ HD)
  ① ان يتأكد من عدم وجود أخطاء كتابية في البرنامج
  ② ان يترجم الـ code من لغة الانسان إلى لغة الآلة.

④ linker :- (الرابط)

* يعمل على تحويل الـ code من نص كتابي إلى برنامج تنفيذي (في الـ HD).

Scanned by CamScanner

⑤ Loader :- (المحمّل)

* يعمل على ارسال البرنامج التنفيذي الى الذاكرة الرئيسية (main memory).
(HD → memory)

⑥ Execute :- (التنفيذ)

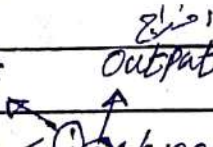
* وتعتبر الخطوة الأخيرة وهي تنفيذ البرنامج وإظهار النتائج النهائية.

★ Program development :-

(معلومات عامة).

* لكتابة برنامج يجب على المبرمج (programmer) ان يفهم جيدًا ماذا يعمل البرنامج (يعني شو بنستفيد منو)
* يجب عليه تحليل البيانات (يعني يعرف كم قيمة رح تدخل مثلاً وكم قيمة رح تخرج من برنامج جمع رقمين رح تدخل قيمتين وتطلع قيمة وحدة وهكذا)
* يجب تحديد الشروط الموضوعة على البيانات (يعني مثلاً لازم الرقم الأول أكبر من ٣ أو يقبل القسمة على كذا وهيك).
* نختار اسرع طريقة لكتابة البرنامج بأقل عدد أسطر اذا أمكن (كلما زيادة عدد الاسطر يعني زيادة سعة البرنامج).
* هاد الحكي بشكل عام عن البرمجة والهدف دائمًا افضل النتائج بأقل تكلفة وأقصر وقت.
* البرمجة هي تحويل الجمل من لغة الانسان الى لغة الآلة.
* جهاز الكمبيوتر عبارة عن (تمثيل) غير قادر على اتخاذ القرار، المبرمج زي ما بيكتب الكمبيوتر بنفذ فلازم نخلي بالنا كيف نتعامل معاه بطريقة صحيحة.

★ Notes :-



* #include <iostream> ①
* ملف البرنامج منبع المدخلات والمخرجات لبداية كتابة الـ code
* يتضمن
* عملية الاستدعاء للمكتبة وبداية معالجة البرنامج

Scanned by CamScanner

② using namespace std; ؛
تستخدم لاستدعاء جزء اصغر من المكتبة لتقليل سعة البرنامج.

③ int main
{
return 0; }
هو عبارة عن الدالة الرئيسية التي تحتوي على البرنامج التنفيذي ، يكتب كل شيء داخل هذه الأقواس لتنفيذه

④ cin, cout
هما لما تريد مثلاً تطبع اشي على شاشة اطرحها ، فمن المنطق نفكر مثلاً print 523 هاي بلغة البشر ، لما أجي انسخها على البرمجية نحكي بدل print ← cout و بتجربها بإشارة أصغر من (<<) مرتين ، بحيث بكتب وأي بدي اياه بين (" ") وآخر اشي نحط (;) فاصلة منقوطة . بالنسبة لـ cin ، بنستخدمها لما بدنا المستخدم (user) يعطينا قيمة لمتغير نستخدمها داخل البرنامج . الـ cin بنستخدم بدها اليمين (>>)

⑤ طبيعة عمل البرنامج كالآتي ، عبارة عن تعريف متغيرات واعطاء قيم و ادخال هذه المتغيرات في عمليات حسابية (÷ ، - ، +) أو منطقية (< ، > ، ==) ، واعطاء النواتج المطلوبة (كل هاد حكي عشان نفهم و الامثلة بتوضح معنا) .

⑥ نهاية أي جملة بالفاصلة المنقوطة وليس بسطر جديد

⑦ لإنزال السطر جملة سطر
cout<<"ahmad"<<endl;
cout<<"ahmad/n";

Scanned by CamScanner

★ أنواع الـ Errors

تنقسم إلى نوعين

syntax error

* يكون الخطأ أثناء كتابة الـ code
مثل عدم كتابة الفاصلة المنقوطة
int x = 5□ x
أو استخدام كلمة محجوزة كاسم لمتغير.
* لا تعرض نتائج على شاشة الإخراج

Logical error

يكون الخطأ منطقي
يعني مش النتيجة اللي بدي إياها تطلع مثل عدم اعطاء المتغير قيمة و استخدامه في عملية حسابية

int x, y, sum ؛
sum = x + y ؛
↳ Logical error.

★ تعريف المتغيرات :-

① طريقة التعريف :-

* لتعريف متغير نستخدم ما يسمى بـ Data type وهي نوع المتغير الذي سنستخدمه ، صحيح ، عشري وغيره ، المتغير يستخدم أصلاً للاحتفاظ بالقيم داخله وتنقل القيم داخل البرنامج دون مناداتها.
* الشكل الأساسي للتعريف ⇦ datatype variablename ؛
* لا يمكن تعريف متغير مرتين (syntax error)
* الأحرف الكبيرة و الصغيرة تعبر متغيرات مختلفة
int I = 2 ؛
cout << i ؛ → syntax error.

Scanned by CamScanner

② Data Types :-

① int (4 byte) يستخدم لتخزين الارقام الصحيحة

② float (4 byte) يستخدم لتخزين الارقام الصحيحة و العشرية

③ double (8 byte) نفس استخدام float لكن الفرق في السعة

④ bool (1 byte) يخزن فيه (0,1) أو (true, false)

⑤ char (1 byte) يخزن فيه اي قيمة على الكيبورد واحدة فقط

1 byte = 8 bit *

Notes on data types :-

* char تطبع a فقط اذا تم تخزينها من قبل المبرمج بقيمتها

```
char x = 'a';
char y = 97;
cout << x << endl;
cout << y << endl;
```

x = 'a' أو x = 97

a
a

* char تطبع آخر خانة إذا تم

```
char x = '97';
cout << x;
```

7

اعطاء قيمتها من قبل المبرمج.

* char تطبع أول خانة اذا تم ادخال

```
char x;
cin >> x; // 97
cout << x;
```

9

قيمتها من قبل المستخدم.

اي معناها ملاحظة بس وما الها داعي

* اذا ادخلنا متغير من نوع char على

```
char x = a;
cout << x+3;
```

100

عملية حسابية فإنه يتعامل مع قيمته من الاسكي كود (a=97 وتزيد واحد وهكذا b=98)

(A=65 وتزيد واحد وهكذا B=66)

* المتغير من نوع bool عند اعطاءه

```
bool x = True;
```

syntax error.

قيمة true أو false يجب ان تكتب ان أول حرف (small)

* المتغير bool يعتبر اي قيمة = 1 ما عدا false و 0 يعتبرها 0 (تخزينا)

Scanned by CamScanner

* الخطأ في هذا البرنامج ان a غير موضوعة داخل (' ') فاعتبرها البرنامج عبارة عن variable غير معرف ← syntax error.

```
char x = a;
cout << x;
```
↳ syntax error

③ variable شروط كتابة الـ :-

① ان لا يبدأ برقم ⇦ `int 3x=5;` ✗
② ان لا يكون كلمة محجوزة ⇦ `int for=4;` ✗ (الكلمات المحجوزة Small)
`int Float=2;` ✓
③ لا يوجد فراغ (white space) ⇦ `int x y = 2;`
④ لا يحتوي على (++ ÷ * - +) ⇦ `int x++ = 2;`
⑤ لا يحتوي على (--- ※ @ #) ⇦ `int x$ = 5;`
`int x_3 = 4;` ✓ (underscore character)

Note :- // ⇦ تعني comment وليس تنفيذي ← سطر واحد

/* ← ⌐ ← بأكثر من سطر. */

* coding example :-
اكتب برنامج لتعريف ٣ متغيرات وطباعة المجموع والمعدل :-

```
#include <iostream>
using namespace std;
int main()
{ int x, y, z;
  x=3; y=4; z=1;
  int sum = x+y+z;
  double ave = sum/3;
  cout << ave << sum << endl;
  return 0;
}
```

استخدمنا double لان ناتج القسمة لا يساوي عدد صحيح بل هو عشري.

Scanned by CamScanner

ex :-

① `bool x = true;`
`cout << x;` | 1

② `bool x = 100;`
`cout << x;` | 1

③ `bool x = 'a';`
`cout << x;` | 1

④ `bool x = a;`
`cout << x` syntax error

⑤ `bool x = "a";`
`cout << x` | 1

⑥ `double d3;`
`d3 = d3 + 1;`
`cout << d3;` logical error

coding ex :-

Write a program contains the value of the monthly salary of an employee and his salary is 200 $. print his monthly and annual salary with a premium of 20% and a discount of 10%

يطلب البرنامج طباعة الراتب الشهري لموظف يتقاضى 200 دولار في الشهر، ومكافأة السنوية مع عمولات بمقدار 10%، علاوة بمقدار 20% ⟸

```
#include <iostream>
using namespace std;
int main()
{ int x = 200;
  int y = x * 12;
  int z = y * 0.1; int y = y + z;
  cout << x << endl;
  cout << y << endl;
```

+20% - 10% = +10%
الزيادة على الشهر تزيد على السنة

200
2640

x	y	z
200	~~2400~~ 2640	240

Scanned by CamScanner

✦ Casting :-

وهو تغيير قيمة المتغير من الجهاز تلقائيًا وهي عملية فنية

① int a = 100;
bool b = (bool)a;
bool c = bool(a);
cout << b << "\n" << c << endl;
cout << a;

حرية وضع الأقواس

int a	bool b	bool c
100	1	1

1
1
100

② float x = 2.5;
cout << int(x) << endl;
cout << x;

x
2.5

2
2.5

✦ عل casting داخل cout & يغير من قيمتها الأصلية.

③ int a = 7.666;
cout << float(a) << endl;
cout << a;

int a
7

7
7

علة عمل casting في البداية في ما يرجع يعيد القيمة الأصلية

④ float a = 7.666;
cout << int(a) << endl;
cout << a;

float a
7.666

7
7.666.

✦ الجملة المحجوزة لعمل الـ casting :-

Static_cast < datatype > (variable);

⑤ int x = 10;
bool y = static_cast<bool>(x);
cout << x << "\n" << y;

int x	bool y
10	1

10
1

⑥ char x = 97;
char y = 'a';
cout << int(y) << endl << x << "\n";

char x	char y
a	a

97
a

Scanned by CamScanner

★ C++ Arithmatic Operations :-

★ العمليات الحسابية في برمجية C++ :-

① `int x = 3;`
`int y = 2;`
`cout << x+1 << endl;`
`cout << y-x << endl << x << "\n";`

4
-1
3

② `int x=2, y=3, z=4;`
`int result = x+y*z-x;`
`cout<<result;`

12

2+3*4-2
2+12-2 = 12

★ حالات القسمة العادية :-

③ `int a=5;`
`int b=2;`
`cout<<a/b;`

2

① اذا كان ناتج طباعة مبا بشرة :-
★ اذا كان المتغيرين int ⟸ int . int

④ `int a=5;`
`float b=2;`
`cout<<a/b;`

2.5

★ اذا كان احد المتغيرات على الأقل float ⟸ float . float

② اذا تم تخزين ناتج الطباعة داخل متغير ثالث :-

⑤ `int a=5, b=2;`
`float c = a/b;`
`cout<<c;`

2.

★ اذا كان كلا المتغيرين int وخزّن داخل float ⟸ int . int

⑥ `int a=5, b=2;`
`int c = a/b;`
`cout<<c;`

2

★ اذا كان كلا المتغيرين int وخزّن داخل int ⟸ int . int

⑦ `float x=5;`
`int y = 2;`
`float c = x/y;`
`cout<<c;`

2.5

★ اذا كان احد المتغيرات على الاقل float وخزّن داخل float ⟸ float . float

⑧ `float x=5; int y=2;`
`int c = x/y; cout<<c;`

2

★ اذا كان احد المتغيرات float (على الاقل) وخزّن داخل int ⟸ int . int

Scanned by CamScanner

* باقي القسمة وحالاته :-

باقي القسمة أو mod أو % يستخدم لطباعة ما تبقى من الارقام عند قسمة رقمين من نوع int .

⑨ int x=9;
int y=4;
cout<<x%y;

* ما تبقى من قسمة الرقم 9 على الرقم 4 | 1

⑩ int x=4;
int y=9;
cout<<x%y;

* بما أن الرقم الاول أصغر من الرقم الثاني فهي لم تقبل القسمة (ناتج القسمة العادية=0) وباقي القسمة هي الرقم الاول كامل. | 4

⑪ Float x=9;
int y=4;
cout<<x%y; syntax error

* يجب ان يكون كلا الرقمين من نوع int ⇦ syntax error (مهمة جداً جداً).

⑫ int x=2;
Float y=4.5;
cout<<x<<y<<endl;
cout<<x%int(y)<<endl;

* يمكن استخدامها اذا قمنا بالـ casting + علامة cout | 2 4.5 / 2 | لا تؤثر على القيم الأصلية

⑬ int x=2;
x+=2;
cout<<x;
cout<<x+2;
cout<<x;

* X+=2 ⇨ X=X+2
X++ ⇨ X=X+1 | 464

* Post and Pre increment/decrement (+) (−)
(X++, ++X, X--, --X)
Post ⇦
Pre ⇦

* عملية زيادة أو نقصان من قيمة المتغير ⇦ وضعت هذه الفقرة لأن بتكلها جديد وليست صعبة ابداً ⇦
X++ ⇒ X=X+1

* X++ ⇦ اطبع ثم زد
++X ⇦ زد ثم اطبع.

Scanned by CamScanner

* تصحيح العمليات الحسابية الموجودة في جملة cout & تؤثر على قيمة المتغير ما عدا post and pre تغير على قيمة المتغير .

(14) int x;
x = 2;
x++;
++x;
cout << x++ << "\n";
cout << ++x << endl;

x
2
3
4
5
6

4
6

* ترتيب أولويات العمليات

① ( )
② Post increment / decrement
③ ! → not x = 5 → true !x → 0 $\frac{x}{*0}$
④ Pre increment / decrement
⑤ * / % → ① arithmatic operation العمليات الحسابية
⑥ + -
⑦ > < <= >= → ② relational operation العمليات العلاقية
⑧ == != → يساوي وغير يساوي
⑨ && → and !, &&, || → ③ Logical operation العمليات المنطقية
⑩ || → or
⑪ =

* شروط (&&) and و (||) or

x	y	x && y	x || y
1	1	1	1
1	0	0	1
0	1	0	1
0	0	0	0

Scanned by CamScanner

→ math function

① #include <cmath>
#include <iostream>
using namespace std;
int main()
{ int x=2, z=3, f=2;
bool y=100;
cout << !y << endl;
cout << ++x+2+x << endl;
cout << x+2*5+x++ << endl;
cout << !y*5-z++ << endl;
cout << pow(x,f) << endl; → Power → x^f
cout << z << endl;
return 0; }

0
8
16
-3
16
4

① ++x+2+x → 3+2+3=8 → الـ operand الأول يأخذ الـ pre بالكامل

② x + 2*5 + x++ → 3+10+3=16
(1) (3) (2)
الـ operand الأول يأخذ الـ post بالكامل

x = 11

③ !y*5-z++ → 0*5-3 = -3

④ pow(x,f) $4^2=16$

!y → not
x!=y → مختلف

② bool x=1;
bool y=0;
bool z=0;
cout << !z << endl;
cout << x && y << endl;
cout << x || y << endl;
cout << y && z << endl;

1
0
1
0

Scanned by CamScanner

✳ بعض الرموز التي تتبع cout ومعانيها :-

① "\n" ⇨ endl.
② "\t" ⇨ tap تترك فراغ كبير
③ "\b" ⇨ يرجع المؤشر بمقدار حرف
④ "\r" ⇨ يرجع المؤشر الى بداية السطر
⑤ "\" \"" ⇨ " " تطبع ما داخلها بين
⑥ "\\ \\" ⇨ \ \ تطبع ما داخلها بين

```
① cout << "Jordan \n";
   cout << "Jordan \t first" << endl;
   cout << "Jordan \b \b mn" << endl;
   cout << "Jordan \r amm" << endl;
   cout << "\\ Jordan \\" << endl;
   cout << "\"Jordan\"" << endl;
```

```
Jordan
Joirdan    first
Jordmn
ammdan
\Jordan\
"Jordan
```

✳ مثال يوضح كيفية تقسيم الادخال على عدة متغيرات :-

```
int x; char c; float z;
① cin >>x >>z ; // 34.5
② cin >> z >> c >> x; // 34.5 9734.52
③ cin >>x >>c ; // 97
④ cin >> c >>x ; // 97
⑤ cin >> z >>x; // 34.52
⑥ cin >>c >> x; // 34.52
⑦ cin >>x >>c; // 34.52
⑧ cin >>c ; // 97
⑨ cin >>x >>c >>z ; // 34A22.3
⑩ cin >> z >> x; // 34.252682
⑪ cin >>x >>c >>z ; // 3422.3
```


Scanned by CamScanner

	int x	char c	float z	
①	34	–	0.5	→ يعطي اللي بعده
②	734	9	34.5	من العدد السابق
③	97	wait!	–	→ ينتظر الادخال لانه ناقص
④	7	9	–	
⑤	wait!	–	34.52	
⑥	4	3	–	
⑦	34	0	–	→ 34 (0.52)
⑧	–	9	–	
⑨	34	A	22.3	
⑩	82	-	34.2526	→ يأخذ 6 خانات Float
⑪	3422	0	0.3	

Scanned by CamScanner

★ String :-

* هي عبارة عن data type يعرّف فيها متغير يخزّن داخله العديد من الحروف و الرموز مثل الاسماء .

* يتم تعريفة بمكتبة وهي ⇐ ※ include<string>

* يمكن الادخال باستخدام cin ولكن هنالك مشكلة مع string ، وهي انه اذا تشمل الادخال المساحة (whitespace) فهو يوقف الادخال تلقائياً ، لذا نحن نستخدم اداة الادخال getline ⇐ getline(cin, string variable)

① string x = "ahmad", y;
cin>>y; // hhh hhh
cout<<x;

hhh

② string y;
getline(cin, y); // hhh hhh
cout<<y;

hhh hhh

* هنالك بعض الـ function التي تستخدم مع الـ string لتسهل لنا جمع البيانات فيها مثل .
strvar.Length()
strvar.size()

③ string x = "ahmad";
cout<< x.Length()<<endl;
coub<< x.size()<<endl;

5
5

مثل ما يوضح المثال السابق فإن كلا الدالتين تعملان على عدّ عدد الخانات (يطبع طول النص) .

Scanned by CamScanner

* المقارنة بين المتغيرات من نوع string :-

```
string str1 = "Hello";
string str2 = "Hi";
string str3 = "Air";
string str4 = "Bill";
string str5 = "Big";
```

ⓐ	str1 < str2	✓	Hello	Hi
ⓑ	str1 > "then"	X	Hello	then
ⓒ	str3 < "An"	✓	Air	An
ⓓ	str1 = "hello"	X	Hello	hello
ⓔ	str4 >= "Billy"	X	Bill	Billy

* عند المقارنة بين كلمتين لا يؤثر عدد حروفها ، يؤثر إذا كان الحرف كبير أو صغير A ⇦ 65 ، a ⇦ 97

* يتم المقارنة حرف الحرف صفن الكلمتين وكلما بعد الحرف عن حرف الـ a كان أكبر فـ b أكبر من a و c أكبر من b وهكذا

Scanned by CamScanner

Control Structure

- **selection**: if, switch
- **repetition**: for, while, do while

If :-

* If هي عملية شرطية ، تعتمد حصول الشرط على المتغير
* اذا كان جواب الشرط صحيح تنفذ عمل if واذا كان خطأ تنفذ عمل else .

```
* if (condition)
  { - - - -
    - - -     } body of if
  }
  else // optional
  { - - - -
    - - - }   } body of else
```

* لا يوجد بعد أقواس if فاصلة منقوطة .
* ال else اختيارية ويمكن حذفها
* اذا كانت جملة if أكثر من جملة تنفيذية يشترط وضع أقواس ← syntax
* اذا كانت جملة else أكثر من جملة تنفيذية و لم يتم وضع الأقواس
← يعتبر الجملة الأولى هي فقط جملة else والباقي تتجه بالبرنامج الأصلي .
* عادة نسأل داخل شرط if هل كذا يساوي كذا نضع (==) اما اذا وضعنا واحدة فقط (هاد يعني خلي قيمة x = قيمة معينة أو يعني غير قيمة المتغير و حطها كذا وعادة بيعطي (Logical) كانو عين القيمة وما سأل عن الشرط

Scanned by CamScanner

* جملة if هي جملة شرطية (يعني تؤثر على قيمة المتغير)
* اذا كان داخل أقواس if فقط ال variable يكون جواب الشرط true اذا كان اي عدد غير الصفر وعندما يكون صفر جواب الشرط false .

① int grade = 50 ;
if (grade <= 50)
{ grade + = 10 ;
cout << "grade = " << grade << endl;
cout << "Pass" ;

| grade = 60
| Pass

② int x = 5 ;
if (x < 5)
{ cout << x+1; }
else
{ cout << x-2 ; }

| 3

③ int x = 3 ;
if (x < 5)
{ x + = 2 ;
cout << x ; }
else
cout << x+1 << endl ;
cout << "good" ;

| 5
| good

* اذا ما حطينا اقواس else بتكون أول وحدة فقط والباقي بجري بشكل البرنامج الرئيسي .

④ int x = 2 ;
if (x < 3)
cout << x << endl;
cout << x+2 << endl;
else
cout << x+2 ;
cout << "good" ; ← Synbax error

* لا يجوز الفصل بين if و else بجمل خارجية وهنا اعتبر أول جملة فقط لـ if والثانية ؟ هنا الخطأ ← وبالتالي يعتبر وهمي الأقواس .

Scanned by CamScanner

⑤
```
int x=2;
if (x>=2)
  if(x>5)
    cout<<x;
  else
    cout<<"thank you";
```
thank you

⑥
```
int x=4; float y=2;
if(x%y==0)
cout<<"ok";
else
cout<<"no";
```
* من قواعد الـ mod انه لا يكون float اي هما .
↩ syntax error

⑦
```
int x=1;
if(x=2)
{cout<<x+1;}
else {cout<<x;}
```
3

* عند وضع مساواة واحدة بدلا من logical ← اجعل قيمة المتغير error.

⑧
```
int a=3, b=2;
if(a==b++)
cout<<"value of a:"<<a;
else
cout<<"value of b:"<<b;
```
a	b
3	2
	3

value of b: 3

* بعد if تنفيذية تؤثر على قيمة الـ variables ثم نطبع نتيجة.

⑨
```
int x=1;
cout<<x+1;
int y=2;
cout<<y++;
cout<<y;
```
x	y
1	2
	3

2 2 3

⑩
```
int x=1;
if(x)
cout<<"ok";
else cout<<"no";
```
ok

x
1 → true

* متى ما كان المتغير true يكون الجواب true أي false.

Scanned by CamScanner

(11)
```
int x; cin >> x;
if (x == 5)
cout << static_cast <double> (5+8);
else if (x == 2)
cout << static_cast <double> (15/2);
else if (x == 1)
cout << static_cast <int> (7.8 + static_cast < double> (15)/2);
else if (x == 3)
cout << static_cast < double> (15/2);
else if (x != 7)
cout << static_cast < int (7.8 + static_cast <double> (15/2));
```

cin x	out Put:
5	13.0
2	7.5
3	7.0
1	15
4	14
7	Print nothing.

(12)
```
int x; x= 3;
if( x>5);
cout << x;
```
3

* لانه if تنتهي مع الفاصلة المنقوطة و يعتبر بأنها الخط من البرنامج الرئيسي

(13)
```
int x; x=3;
if(x>5)
cout<<x;
else
cout<<+x; ⇝ syntax error
```

* لانه أنهى if مع ; فاعتبر else شيء غير معرّف بعد if .

Scanned by CamScanner

* coding ex :-

① Write a program to input 3 numbers and print the max value.

```
#include <iostream>
using namespace std;
int main( )
{ double x, y, z;
cin >>x >>y >> z;
double max = x;
if (y>max)
max = y;
if (z>max)
max =z;
cout << max;
return 0; }
```

هناك الكثير من الطرق لحل هذا السؤال و غيره وهذه احد الطرق فقط .

② Write a program that ask a user to enter a grade and print the result as :-

90 - 100 → A
80 - 89 → B
70 - 79 → C
50 - 69 → D
<50 → F

```
#include <iostream>
using namespace std;
int main ( )
{ cout << "Enter your grade" <<endl;
int x;
cin >>x;
```


Scanned by CamScanner

```
if (x>=90 && x<=100)
cout<<x<<endl<<"A"<<endl;
if(x>=80 && x<=89)
cout<<x<<endl<<"B"<<endl;
if(x>=70 && x<=79)
cout<<x<<endl<<"C"<<endl;
if(x>=50 && x<=69)
cout<<x<<endl<<"D"<<endl;
if(x<50)
cout<<x<<endl<<"F"<<endl; return 0; }
```

else cout<<"fail"; ← في حالة لم يتحقق أي شرط

③ Write a program that ask user to enter a value and print if this value even or odd.

```
#include<iostream>
using namespace std;
int main( )
{ cout<<"Enter the value"<<endl;
  int x; cin>>x;
  if(x%2==0)
  cout<<"even";
  else
  cout<<"odd";
```


Scanned by CamScanner

* هناك طريقة لكتابة if بسطر واحد وعمل نموذجين

① condition ? C++ sentence : C++ sentence ;

② cout<< (condition ? some thing : some thing) ;

* لازم مع هذا النظام وجود else وإلا يعطي syntax error
* من عيوبها انها تأخذ جملة تنفيذية واحدة

grade >= 60 ? cout<< "Pass" : cout<< "fail" ;
condition if body of if else body of else

بعض الفروق بين النموذجين ① و ②

* الاول يمكن عمل جميع انواع الجمل اما الثانية فقط cout
* يمكن كتابة هذين النموذجين بجملة if العادية وليس العكس

* int x; x=10 ;
x >= 11 ? x++ : cout<< x++;
cout<< x;

x
10
11

* Constant (const)

* تقوم const بتثبيت قيمة المتغير وعدم تغييرها عند تنفيذ البرنامج .
* وعند تغيير قيمتك داخل البرنامج ← syntax error

const int x ; x=4; → syntax error *

Scanned by CamScanner

Switch :-

* الـ switch يطلب ادخال قيمة أو تعريف قيمة من قبل الـ user كي يبحث عن الـ value الذي يطابق قيمة ويبدأ البرنامج حتى أول break .
* اذا انتهت فاصلة منقوطة بعد أقواس switch ⟵ Syntax error
* الـ break ينهي البرنامج ويخرج منه تمامًا .
* عندما يجد الرقم قيمته ، يقوم بتنفيذ باقي الـ cases دون النظر باكي قيمتها .
* الـ switch لا تأخذ الا int او char .
* اذا كان داخل switch (x++) صحيح انها تزيد القيمة بعد الخروج من الجملة ، وهنا تنفذ الـ case الذي يحمل القيمة الأصلية لانها القيمة الجديدة داخل الأقواس .

```
switch( expresion)
{ case value 1 : ---- ; break;
  case value 2 : ----- ; break;
     :
  case value n : ---- ; break;
  default :
  ---- }
```

① `int x=1 ; cin>>x ;`
`switch( x)`
`{ case 1 : cout<< x++ ; break;`
`case 2 : cout<< ++x ;`
`case 3 : cout<< x ;`
`default :`
`cout<< x-- ; }`

int x
cin → ~~1~~ 2

1

int x
cin → ~~2~~ 3 2

3343

int x
~~5~~ 4

5

Scanned by CamScanner

② bool x = true;
switch (x)
{ case 1 : cout << x++; break;
case 2 : cout << ++x; break;
case true : cout << ++x; break;
case 97 : cout << x; break;
case 'a' : cout << x--; break;
}

{ 1 == true
{ 97 == 'a'

لا يجوز ان يكون أكثر من case نفس القيمة.

→ syntax error

③ int x = 1;
switch (x++)
{ case 1 : cout << x; break;
case 2 : cout << x+5;
}

x	2
2	

* بعد خروج التعليمة من حالة switch يزيد بمقدار واحد ولكن يتعامل الـ case القيمة قبل الزيادة (x++).

④ int x = 1;
switch(x)[;] → الخطأ. بسبب الفاصلة.
{ case 1 : cout << x++;
case 2 : cout << ++x;
→ syntax error.

الفرق بين البلاش سلاش والقسمة ؟

✓ "\n" ✓ 5/2 ✗

Scanned by CamScanner

for :-

* دالة for تعمل على تكرار تنفيذ الجمل التي تحتوي عليها لعدد معين.

مثال ① ② ④

```
for(int i=1 ; i<=2 ; i++)
{
  ③
}
⑤ cout << --- ;
```

* ترتيب الخطوات عند التنفيذ ←

① for(int i=0 ; i<=2 ; i++)
{ cout << i << endl;
  cout << "hello" << endl;
}
cout << "ok";

i	
0	0
1	hello
2	1
3	hello
	2
	hello
	ok

② int i=0;
for(; i<=3 ; i++)
cout << i << endl;
cout << i+2;

0	* يمكن كتابة اكثر من اجزاء دالة for فوقها.
1	
2	* عند عدم وضع ; ; تكون الجملة الأولى هي دالة for.
3	
6	

* عند الخروج من for فإن قيمة المتغير تزيد على القيمة الـ decline بمقدار مرة واحدة لتنفيذ الجمل التي تلي for.

③ for(int i=0 ; ; i++)
cout << i;
cout << "ok";

0	وهنا لا يوجد شرط للتوقف عندها.
1	infinite Loop
2	
⋮	
⋮	

Scanned by CamScanner

④ for (int i=0; i<3;)
{ cout << i; }

0	infinite loop
0	
⋮	

لأن الرقم لا يزداد فيتكرر الـ loop دون توقف

⑤ for (int i=0; i<3;)
{ i++;
cout << i; }

1	1
~~0~~	2
~~2~~	3
~~3~~	

* اختلفت ان الزيادة في قيمة i قبل الطباعة.

⑥ for (int x=1; x<20; x*=5)
cout << x;

x	
1	1
~~5~~	5
25	

coding ex :-

① Write a program that print the numbers between(1-10) and print if it even or odd.

```
#include <iostream>
using namespace std;
int main ( )
{ for ( int x=1 ; x<=10 ; x++)
  if ( x%2 == 0 )
   cout << x << "even" << endl;
  else
  cout << x << " odd " << endl;
  return 0; } .
```

② write a program to enter 5 numbers and print if it positive or negative.

Scanned by CamScanner

```
#include<iostream>
using namespace std;
int main()
{ int y;
 for(int x=1; x<=5; x++)
 { cin>>y;
  if(x>=0)
  cout<<x<<"Positive"<<endl;
  else
  cout<<x<<"negative"<<endl;
  return 0; }
```

② write a program its output is →

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

```
for(int i=4; i>=1; i--)
{ for(int x=1, x<=i; x++)
  cout<<"*";
  cout<<endl;
}
```

④ write a program that find 5! → 5*4*3*2*1 (مضروب الرقم)

```
int x, y=1;
cin>>x;
for(i=1; i<=x; i++)
y=y*i
cout<<y;
```

⑤ write a program that find sum of numbers (1-10)

```
int x; int sum=0; cin>>x;
for(int i=1; i<=x; i++)
sum=sum+i
cout<<sum;
```


Scanned by CamScanner

* عند السؤال عن طباعة عدة صفوف وأعمدة كتكرار
طباعة ⊙ بحوم مراراً على ٣ صفوف مثالية نتبع الآتي :-
- نستخدم rested for وهي وجود لفة for داخل لفة for اخرى
حيث تمثل الجملة الأولى "الكبيرة" عدد الصفوف وتمثل الثانية عدد الاعمدة
- اذا كان عدد الاعمدة ثابتة في كل الصفوف يفرض الـ condition رقماً ثابتاً (لـ For الثانية)
- اذا كان عدد الاعمدة غير ثابتة في جميع الصفوف يفرض الـ condition بالنسبة للمتغير الأول (لـ For الثانية)
- ارجع للامثلين السابقين للتطبيق.

for with char :-

تذكير ببعض قواعد char :-

```
char x = 97;
cout<<x;
```
a

```
char x = 'a';
cout<<x;
```
a

```
char x = '97';
cout<<x;
```
7

```
char x;
cin>>x; // 97
```
9

①
```
for( char x='a'; x<=101; x++)
cout<<x<<"\t"<<int(x)<<endl;
```

x	
a → 97	a 97
b → 98	b 98
↓	c 99
	D 100
	E 101

②
```
char x = 'a';
x++;
cout<<x;
cout<<x+1;
```

x
a
b

b
99

* اذا قمنا بعملية حسابية على char ولم نخزنها نتعامل مع قيمتها

Scanned by CamScanner

* for (int i=1 ; i<=3 ; i++)
{ if (i%2 == 0)
break;
cout<<i;
}
cout<<"ok";

i	1
i%2	ok

* وجود break داخل if داخل for يعني اطبع من كل for . (for داخل if)

* for (int x=1 ; x<10 ; x++)
{ if(x%2 == 0)
continue; ← C++ keyword.
cout<<x<<endl;
}

برنامج لطباعة الارقام الفردية . for لا يرجع لمتغير ما لم تكمل دورة ارجع لـ for (if بمجرد ما يتحقق شرط if تأتي لتكملة ما تبقى من متغيرات عليه)

* for(char e = 'a' ; e<='d' ; e++)
{ if(e=='c')
continue
cout<<e<<endl; }

a
b
d

Scanned by CamScanner

While :-

```
while (expresion)
{
  // body of while
}
```

* نفس for بس الاقواس مفرقة بالمعلومات .
* حالة (while-do) تنفذ الجمل التنفيذية قبل اختبار الشرط
  فيتم تنفيذ do مرة واحدة على الأقل .
* في حالة (while - do) اذا لم تحمل ال do التنفيذية أقواس
  ⟵ syntax error
* اذا كان المتغير bool فإن الناتج يكون إما 1 أو 0
* في حالة (while - do) بعد أقواس ال while توضع فاصلة منقوطة
  اما في حالة while العادية اذا وضع لها بشرط فاصلة منقوطة معناها
  عمل لف حتى يصير الشرط غلط .

①
```
int x=5;
while (x<=7)
{ cout<<x;
  x++; }
```

x
5
6
7
8

567

②
```
int x=5;
while ( x-- >=1)
{ cout<<x<<endl;
  cout<<"ok"<<endl; }
cout<<"done<<endl;
```

x
5
4
3
2
1

4
ok
3
ok
2
ok
1
ok
0
ok
done

Scanned by CamScanner

→ do-while

③ int x=1;
do
{ x++;
cout<<x; }
while (x<=5); → semicolon

2
3
4
5
6

④ int x=5; int f=3;
do {
x+=f;
f--;
cout<<x<<endl; }
while (f);

x	f
5	3
8	2
10	10

8
10
11

⑤ bool x=1;
while(x<=5)
{ x++;
cout<<x<<endl; }

1
1
1
1
⋮

Infinite Loop

⑥ int x=1;
while (x<10)
{ if (x%2==0)
continue;
x++;
cout<<x<<endl; }

x
1
2

infinite Loop

لما توصل 2 بتعمل continue وبترجع تعمل 2 تاني وهكذا.

Scanned by CamScanner

rand :-

* هي عبارة عن امر طباعة رقم عشوائي بين 0 , 32767
  ↳ rand.max

* نستخدم `#include <C Std Lib>`

```
for(int i=0; i<10; i++)
    cout<< rand( );
```

10
5
2
⋮

* لاستخدام ال rand داخل البرنامج يجب حصر ال Range لتقليل عدد الاحتمالات :-

offset + rand( ) % scaling factor.
↳ = = ليست باقي قسمة
أول اشي نحدد اعلى وأقل قيمة

* min = offset
* max = scaling factor + min - 1
* % تساوي باقي القسمة حتى لو كان عدد ثاني . مثلاً

= = Scaling factor

4 أرقام

① `for(int i=2; i<=5; i++)`
`cout << 1 + rand( ) % 6;`

① 153426 ② 1153 ③ 115 ④ 1761

[1,6] ← Range ال

min = offset = 1 (أقل قيمة)
max = scaling factor + min - 1 = 6 + 1 - 1 = 6 (اعلى قيمة)

② بما انه 4 ارقام ← أقل قيمة و 6 اعلى قيمة ← الجواب

Scanned by CamScanner

② int x=1 ;
while (x<3)
{ cout<< (2 +rand ( ) % 6);
x++ ;
}

① 78 ② 77 ③ 473 ④ 783

min = 2
max = 6+2-1 = 7 ← Range [2,7] ← while دورتين

الجواب ② .

لا تحسبن المجد تمرًا أنت آكله
لن تبلغ المجد حتى تلعق الصبرا

مع تمنياتي بالتوفيق والنجاح
مروه حلوه
هندسة حاسوب

Scanned by CamScanner