الفصل الثالث
جمل القرار
Conditional Statements

1.6 مقدمة

أن تنفيذ البرنامج بلغة C++ يكون من الأعلى إلى الأسفل لكن بعض الأحيان نحتاج المترجم أن يقفز بعض سطور البرنامج أذا تحقق شرط معين. وقد نحتاج أن نكرر عملية من عمليات الإدخال أو الإخراج أو الحساب عددا من المرات، وقد نحتاج أن نبني شبكة من توزيع الأوامر على عدد من سطور البرنامج، حسب مقتضيات بعض الأحوال، وحسبما تقتضية طبيعة المسألة في هذه الحالات نحتاج ان نتعلم : أساليب الشرط، وأساليب التكرار، وكيفية التحكم في مسار البرنامج، وتعد أساليب الشرط، والتكرار بمثابة القلب في جسم لغات البرمجة وبدونها لا يمكن تنظيم أي برنامج.وتوفر لغة C++ للمبرمج عددا من الأساليب والدوال الفعالة ، المتعلقة بهذا الشأن وتمتاز هذه الأساليب بأنها أساليب بنائية أو بنيوية structured أي يمكن تنظيم عمليات التحكم والتكرار فيها بأسلوب ذاتي من بداية العمليات وحتى نهايتها دون تدخل من المبرمج أثناء هذه العمليات للأشراف على التوجية والتخطيط لكل خطوة من خطوات البرنامج ويعرف بعض المختصين في البرمجة البرمجة البنيوية : أنها البرمجة التي لا تستعمل جملة الانتقال GOTO لتوجيه البرنامج في كل خطوه ومع أن لغة C++ توفر استخدام هذه العبارة لكنها لا تستخدم إلا للضرورة.

2.6 الجمل الشرطية

1.2.6 جملة الشرط (أذا if Statement)

تأخذ هذه الجملة الشكل العام التالي:

```
If (condition) statement;
```

حيث أن:

condition : يمثل الشرط ويجب أن يكتب بإحدى العبارات المنطقية

statement: وتمثل العبارة التي ينفذها الحاسب في حالة تحقق الشرط

مثال :

```
if (x>=10)  y=x+5;
if (Re<=2300) cout<<"Flow is laminar\n";
```

ويمكن جمع أكثر من شرط في جمله واحده على النحو التالي:

```
if ((height>180) && (weight<80))  cout<<"OK";
```

if ((height>180) || (weight<80)) cout<<"OK";

2.2.6 جملة الشرط (أذا وإلا if..else Statement)

ممكن أن تستخدم عبارة (else) مع عبارة (if) وتكتب كما بالصيغة التالية:

```
if (condition) statement1;
else statment2;
```

حيث أن:

condition : يمثل الشرط ويجب أن يكتب بإحدى العبارات المنطقية

statement1: وتمثل العبارة التي ينفذها الحاسب في حالة تحقق الشرط

statement2 : وتمثل العبارة التي ينفذها الحاسب في حالة عدم تحقق الشرط

- من الصور الأخرى التي تكتب بها عبارة الشرط if هي على النحو التالي:

```
1) if (condition)
{
.
. Set of statements
.
}

2) if (condition)
{
.
. Set of statements
.
}
else
{
.
. Set of statements
.
}
```

الصيغة رقم 1 تستخدم لما يطلب من البرنامج تنفيذ أكثر من امر بعد تحقق الشرط فبهذه الحالة جميع العبارات التي تأتي بعد الشرط يجب ان تحصر بين القوسين الكبيرين { }.
إما بالصيغة رقم 2 أذا كان الشرط متحقق فستنفذ مجموعة العبارات التي تلي الشرط إما أذا كان الشرط غير متحقق فتنفذ العبارات التي تلي كلمة (else).

مثال : برنامج لدرجات الطلاب ويكون المطلوب

إذا كانت درجات الطالب ما بين 90 – 100 يطبع A

إذا كانت درجات الطالب ما بين 80 – 89 يطبع B

إذا كانت درجات الطالب ما بين 70 – 79 يطبع C

إذا كانت درجات الطالب ما بين 60 – 69 يطبع D

إذا كانت درجات الطالب ما بين 50 – 59 يطبع E

إذا كانت درجات الطالب ما بين 0 – 49 يطبع F

الحل :

```
#include <iostream.h>
main()
{       int mark;
        char grade;
        cin>>mark;
        if ((mark>=90) && (mark<=100)) grade='A';
        if ((mark>=80) && (mark<=89)) grade='B';
        if ((mark>=70) && (mark<=79)) grade='C';
        if ((mark>=60) && (mark<=69)) grade='D';
        if ((mark>=50) && (mark<=59)) grade='E';
        if ((mark>=0) && (mark<=49)) grade='F';
        cout<<grade<<"\n";
        return 0;}
```

يمكن حل المثال أيضا باستخدام عبارة if..else :

```
#include <iostream.h>
main()
{
	int mark;
	char grade;
	cin>>mark;
	if (mark>=90) grade='A';
	else
	if (mark>=80) grade='B';
	else
	if (mark>=70) grade='C';
	else
	if (mark>=60) grade='D';
	else
	if (mark>=50) grade='E';
	else
	grade='F';
	cout<<grade<<endl;
	return 0;
}
```

ملاحظة: لا تستخدم (if) بعد (else) عندما يكون هناك احتمال واحد متبقي (كما بالمثال أعلاه) ، وتستخدم بعد (else) أذا كان هنا أكثر من احتمال واحد ويجب اختيار احدهما لأن استخدامها مع وجود احتمال واحد يعتبر غير منطقي بالرغم أن البرنامج ممكن أن ينجز العمل.

مثال : أكتب برنامج عن عمل أشارة المرور.

```
#include <iostream>
#include <string>
using std::string;
using std::cout;
using std::cin;
main( )
{
string color;
        cin>>color;
        if (color=="red")  cout<<"stop\n";
        else
        if (color=="yellow") cout<<"wait\n";
        else
        if (color=="green")   cout<<"pass\n";
        return 0;
}
```

مثال : أكتب برنامج لحساب جذور المعادلة $ax^2+bx+c=0$؟

```
#include <iostream.h>
#include <math.h>
main()
{
float a,b,c,x1,x2,r;
cin>>a>>b>>c;
r=b*b-4*a*c;
if (r>0)
{
```

```
        x1=(-b-sqrt(r))/(2*a);
        x2=(-b+sqrt(r))/(2*a);
        cout<<"root1="<<x1<<"\n";
        cout<<"root2="<<x2<<"\n";
}
else
cout<<"Error square root with negative value \n";
return 0;
}
```

3.2.6 إذا المركبة Compound if

من الممكن أن تستخدم (if) بشكل متداخل مع (if OR else) أخرى، وبهذا الحالة تسمى مركبة (أي ممكن أن يكون بعد الشرط الذي بعد (if) عبارة (if) أخرى وممكن أيضا بعد عبارة (else) وممكن أن تكون أكثر من عبارة (if) واحدة . فمثلا تريد أن تفحص نوعية رمز معين ووفقا لذلك تقرر ما هو الأجراء الواجب أتباعه وكما يأتي:

```
if (condition1)
  { if (condition2)
        Statment1;
else
     { if (condition3)
        Statment2;
      else
        Statment3;
      }
  }
else
Statment4;
```

مثال:

```
if ((ch >= ‘0’) && (ch <= ‘9’))
Kind = digit;
else {
if ((ch >= ‘A’) &&( ch <= ‘Z’))
Kind = upperletter;
else {
if ((ch >= ‘a’) && (ch <= ‘z’))
Kind = lowerletter;
else
Kind = special;
}
```

4.2.6 جملة التوزيع switch statement

الصيغة العامة لهذه الجملة هي :

```
switch (var)
{
case value 1:statement 1;break;
case value 2:statement 2;break;
.
.
.
case value n:statement n;break;
default;last statement;
}
```

ملاحظات:

1) المتغير الذي يأتي بعد عبارة (switch) يجب أن يعرف كمتغير صحيح (int) أو رمزي (char) فقط.

2) value 1, value 2……value n هذه تمثل قيم المتغير الذي ذكر بالعبارة أعلاه.

3) statement 1,statement 2 …….statement n تمثل العبارات المراد تنفيذها لما تكون قيمة المتغير value 1,value 2…….value n على التوالي.

4) last statement تمثل العبارة التي يتم تنفيذها عند عدم تنفيذ أي من الحالات السابقة أعلاه.

5) أذا لم تضع الأمر (break) فان التنفيذ سيستمر للعبارة اللاحقة وهكذا إذا كان المبرمج يقصد ذلك وفي هذه الحالة يتم وضع ملاحظة.

6) لاحظ أن صيغة (switch…case) دائما تحتاج إلى أقواس بداية ونهاية { }.

7) يفضل استخدام صيغة (switch…case) في البرامج التي تحتاج إلى ثلاث عبارات (if) متتالية أو أكثر.

مثال : أكتب برنامج عن عمل أشارة المرور باستخدام صيغة (switch…case)؟

```
#include <iostream>
using namespace std;
main( )
{
char color;
cin>>color;
switch(color)
{
case 'r':cout<<"stop";
        break;
case 'y':cout<<"wait";
        break;
case 'g':cout<<"pass";
        break;
}
return 0;}
```

مثال : توفر إحدى الجامعات سكناً لأعضاء هيئة التدريس فيها مقابل أجور مبينة على النحو التالي :

نوع السكن Type	الإيجار بالدينار Rent
1	140
2	70
3	60
4	50

أكتب برنامجاً تحسب فيه صافي الراتب net بعد اقتطاع أجرة السكن rent من الراتب الكلي salary .

الحل:

```
#include <iostream.h>
void main()
{
int type;
float salary,net;
cin>>salary>>type;
switch (type)
{
case 1:net=salary-140;
        break;
case 2:net=salary-70;
        break;
case 3:net=salary-60;
        break;
case 4:net=salary-50;
        break;
}
cout<<"type="<<type<<" net="<<net;}
```

5.2.6 عامل الشرط الثلاثي (?: Conditional Ternary Operator)

عامل الشرط الثلاثي يفحص تعبير، ويعيد قيمة معينة إذا كان ذلك التعبير صح، ويعيد قيمة مختلفة إذا كان ذلك التعبير خطأ ، هذا العامل هو اختصار لعامل (if else) و الصيغة العامة له:

condition ? statement1: statement2

فإذا كان الشرط (condition) صح فان التعبير سينفذ العبارة (statment1) إما إذا كان خطأ فانه سينفذ العبارة (statment2) ممكن أن يرد هذا العامل مع التعابير الحسابية كما بالأمثلة أدناه:

7==5 ? 4: 3 // يعطي (3) لأن (7) لا تساوي (5)

7==5+2 ? 4: 3 // يعطي (4) لأن 7=2+5

5>3 ? a: b // يعطي القيمة a لان (5) أكبر من (3)

a>b ? a: b // يعيد أيهما أكبر a أو b

هذا التعبير الثلاثي يمكن إن نعبر عنه بما يأتي (إذا كان الشرط صحيحا فعليه ستكون النتيجة هي النتيجة الأولى وبخلاف ذلك إي إذا كانت نتيجة الشرط غير صحيحة فستكون النتيجة هي النتيجة الثانية) عادة هذه القيمة المعادة يجب إن تسند إلى متغير.

مثال:

```
include <iostream.h>
void main( )
{
int min,i=10,j=20;
min =(i < j ? i: j);
cout<<min;
}
```

أسئلة حول الجمل الشرطية

س1/ أكتب برنامج يقوم بقراءة قيمة B بحيث تكون (1 أو 2 أو 3 أو 4) ثم يحسب ويطبع قيمة C حسب الأتي

$$C = \begin{cases} B^2 + B^3 : B = 1,2 \\ 3B + 1 : \ B = 3,4 \end{cases}$$

س2/ أكتب برنامج لإيجاد قيمة D حيث أن :

$\mathbf{D} = 3\mathbf{x}^3 - 2\mathbf{x} + 5$ if x>5

$\mathbf{D} = \sqrt{\mathbf{x} - 9}$ if $5 \geq \mathbf{x} \geq 2$

$\mathbf{D} = |\mathbf{x} - 7|$ if x<2

س3/ أكتب برنامج يقوم بإدخال ثلاثة إعداد ويطبع أكبرها.

س4/ أكتب برنامج بلغة C++ لقراءة أطوال أضلاع شكل رباعي أضلاعه هي A,B,C,D على التوالي ثم يطبع عبارة (square) أذا كان الشكل مربع ويطبع (rectangular) أذا كان الشكل مستطيل بينما يطبع عبارة (neglected) إذا لم يكن أي منهما.

س5/ أكتب برنامج بلغة C++ تدخل فيه خمسة أعداد من 1..5 بحيث يطبع العبارات التالية:

Good	أذا كان العدد المدخل 1
Nice	أذا كان العدد المدخل 2
Excellent	أذا كان العدد المدخل 3
Masterful	أذا كان العدد المدخل 4
Incredible	أذا كان العدد المدخل 5
Out of Range	أذا لم يكن العدد المدخل أيا مما سبق

س6/ أكتب برنامج يحسب قيمة X باستخدام صيغة (switch…case) بحيث:

X=A+B if j=1

X=A-B if j=2

X=A*B if j=3

X=A/B if j=4

س7/ أكتب برنامج لاختبار عدد فردي فيما إذا كان يقبل القسمة على 3، 5 بذات الوقت؟

س8/ أكتب برنامج لإدخال عددين ويختبر أمكانية قسمة العدد الأول على العدد الثاني دون باقي؟