

تعريف المتغيرات :

عند كتابة أي برنامج بلغة ++C، نحتاج لتخزين المعلومات الواردة للبرنامج في ذاكرة الحاسوب تحت عناوين يطلق عليها أسماء المتغيرات، وبما أن أنواع المعلومات المراد تخزينها تكون عادة مختلفة مثل القيم الحقيقية أو الصحيحة أو الرمزية فإننا نحتاج أن نعلم المترجم في بداية البرنامج عن أنواع المتغيرات التي نريد استخدامها فمثلاً :-

الكلمات integer1 , integer2 , sum هي أسماء لمتغيرات عبارة عن أعداد صحيحة (النوع int) وهو أحد أنواع البيانات المتوفرة في ++C .
يمكن تعريف المتغيرات في أي مكان في البرنامج لكن يجب تعريفها قبل استعمالها، يمكن تعريف المتغيرات التي تنتمي إلى نفس النوع في سطر واحد.

تسمية المتغير:

يتم تعريف المتغير بذكر الاسم ونوع البيانات التي يمكن أن يحملها هذا المتغير من أي سلسلة تحتوي على أحرف Letters أو أرقام Digits أو خطأً تحتياً Under
(_)score، على أن لا يبدأ اسم المتغير برقم. ومن الجدير بالذكر أن لغة ++C تفرق بين الحروف الأبجدية الصغيرة والكبيرة، فمثلاً الأسماء integer1 , Integer1 تعامل كمتغيرات مختلفة.

الدخول من لوحة المفاتيح :-

`cin >> integer1`

هذه العبارة تخزن الرقم الذي يكتبه المستخدم من لوحة المفاتيح في متغير يدعي

integer1. يمثل الكائن cin- والذي يلفظ كـ in C- لوحة

المفتاح، ويأخذ عامل الحصول `get from (>>)` الأشياء الموضوعه على يساره ويضعها في المتغير الموجود على يمينه، عند تنفيذ هذه العبارة ينتظر البرنامج أن يكتب المستخدم رقماً من النوع `integer` ويضغط على مفتاح `Enter`، يتم تعيين القيمة التي أدخلها المستخدم إلى المتغير `integer1`.

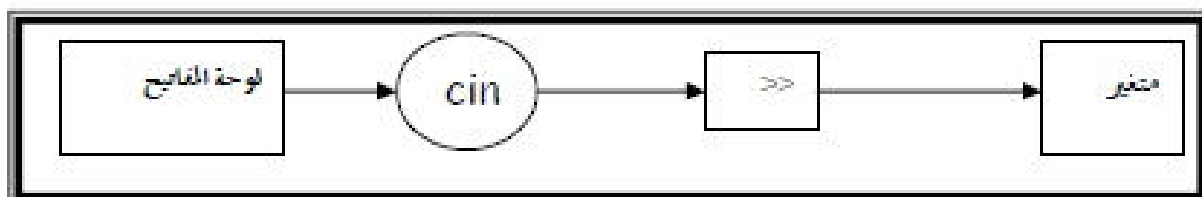
يمكن استعمال عامل الحصول عدة مرات في نفس العبارة:

```
cin >> integer1 >> integer2
```

يضغط المستخدم هنا `Enter`، أو مفتاح المسافة `Space`، أو مفتاح `Tab`

بعد كل قيمة، قبل أن يكتب القيمة التالية، ولكنه من الأفضل عادة إدخال قيمة واحدة في كل مرة لتجنب الخطأ.

الشكل (٢-١) يوضح الدخل بواسطة `cin`.



شكل (٢-١) يوضح الدخل بواسطة ++C

المناور endl:

العبارة:

```
cout << "sum = " << sum << endl
```

تطبع النص `sum =` متبوعاً بقيمة `sum`، نلاحظ أننا استخدمنا `endl` وهو

وسيلة أخرى في ++C للانتقال إلى سطر جديد، ويسمى مناور `manipulator` و `endl`

اختصاراً لـ `end line`، وهو يعمل تماماً كما يعمل تتابع الهروب `\n`.

العوامل الحسابية (Math Operators) :

لقد استعملنا عامل الجمع (+) لجمع integer1 إلى integer2، تتضمن C++ العوامل الحسابية الأربعة الاعتيادية بالإضافة إلى عامل خامس كما مبين في الجدول التالي:

العامل	الوظيفة	التعبير الجبري	التعبير في C++
+	جمع	B+h	B+h
-	طرح	B-h	B-h
*	ضرب	Bh	B*h
/	قسمة	B/h,	B/h
%	الباقى	B mod h	B%h

العوامل الأربعة الأولى تنجز أعمالاً مألوفاً لدينا، أما عامل الباقي % المسمى أيضاً المعامل modulus، يتم استعماله لحساب باقي القسمة لعدد صحيح على عدد آخر، لذلك فالتعبير 3%20 يساوى 2. تسمى هذه العوامل الحسابية بالعوامل الثنائية لأنها تعمل على قيمتين.

يمكن استعمال أكثر من عامل في تعبير رياضي واحد، فمثلاً التعبير:

$$C=(f-32)*5/9;$$

يجول درجة الحرارة من مئوية إلى فهرنهايت. (استعملت الأقواس لكي يتم تنفيذ

الطرح أولاً بالرغم من أولويته المتدنية، يشير المصطلح أولوية Precedence إلى ترتيب تنفيذ العوامل، العاملان * و / هما أولوية أعلى من + و -). وهذا ما سنراه لاحقاً بعد أن نتعرف على بقية عوامل C++.

العوامل العلائقية :

هنالك ستة عوامل علائقية مبينة في الجدول أدناه:

الرمز	المعنى	مثال
==	يساوى	$a==b$
!=	لا يساوى	$a!=b$
>	أكبر من	$a>b$
<	أصغر من	$a<b$
>=	أكبر من أو يساوى	$a>=b$
<=	أصغر من أو يساوى	$a<=b$

تكون التعبيرات المبينة في عمود المثال صحيحة أو خطأ وفقاً لقيم المتغيرين a و

b.

فلنفرض مثلاً أن:

□ a يساوى 9

□ b يساوى 10.

التعبير $a==b$ خطأ.

التعبير $a!=b$ صحيح وكذلك التعبيرين $a<b$ و $a<=b$ ،

والتعبيرين $a>b$ و $a>=b$ خطأ..

الملخص:

- ◆ تبدأ التعليقات في ++C والتي تتكون من سطر واحد بشرطة مزدوجة (//).
- ◆ تبدأ التعليقات في ++C والتي تمتد لعدة أسطر بالرمز /* وتنتهي بالرمز */.
- ◆ السطر `#include<iostream.h>` يسمى "مرشد المهيئ" وهو عبارة عن تعليمة للمصرف أن يضمن الملف `iostream.h` في البرنامج والذي يجب تضمينه في أي برنامج يقوم بعمليات إدخال وإخراج.
- ◆ يبدأ تنفيذ برنامج ++C من الدالة `main()`.
- ◆ المتغيرات في ++C يجب الإعلان عنها قبل استعمالها.
- ◆ يتم تعريف المتغيرات في ++C بذكر اسمها ونوع بياناتها وتكون الاسم من أي سلسلة تحتوي على أحرف أو أرقام أو خطأً تحتياً (_) على أن لا يبدأ اسم المتغير برقم.
- ◆ ++C حساسة تجاه الأحرف ونعني بذلك أنها تفرق بين الحروف الأبجدية الصغيرة (small) والكبيرة (capital).
- ◆ يرتبط كائن الخرج `cout` مع الشاشة وهو يستخدم في إخراج البيانات.

بنيات التحكم Control Structures

بنيات التحكم الشرطية :

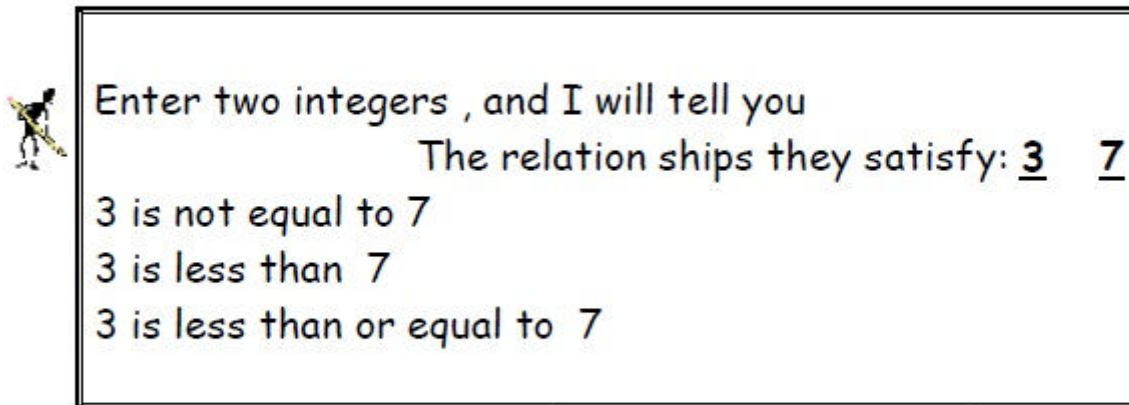
1- العبارة If

أسهل طريقة لاتخاذ قرار في C++ هي بواسطة العبارة If

```
//Program 2-1:  
#include <iostream.h>  
main ( )  
{  
int num1 , num2;  
cout << " Enter two integers, and I will tell you\n"  
<<" the relation ships they satisfy: ";  
cin >> num1>> num2;  
if (num1== num2)  
cout << num1 << " is equal to " << num2 << endl;  
if (num1!= num2)  
cout << num1 << " is not equal to " << num2 << endl;  
if (num1< num2)  
cout << num1 << " is less than " << num2 << endl;  
if (num1> num2)  
cout << num1 << " is greater than " << num2 << endl;  
if (num1<= num2)  
cout << num1 << " is less than or equal to " << num2  
<< endl;  
if (num1>= num2)
```

```
cout << num1 << " is greater than or equal to " <<  
num2<< endl;  
return 0;  
}
```

الخرج من البرنامج بافتراض أن المستخدم قد أدخل الأرقام $num1 = 3$ ، $num2 = 7$



تتألف العبارة `if` من الكلمة الأساسية `if`، يليها تعبير اختبار بين قوسين، ويتألف جسم القرار الذي يلي ذلك إما من عبارة واحدة، أو من عدة عبارات تحيطها أقواس حاصرة `{ }`