Edited with the trial version of
Foxit Advanced PDF Editor
To remove this notice, visit:
www.foxitsoftware.com/shopping

أولاً ..أنواع اللغات :

يمكن تقسيم اللغات المستخدمة في البرمجة الى ثلاثة أنواع :

1 -لغة الآلة
2 -لغة المجمع
3 -اللغات عالية المستوى .

1 -**لغة الآلة** : هي اللغة التي يستطيع الحاسب أن يفهمها مباشرة وهي معرفة من قبل البنية الصلبة للحاسب ,تتألف بشكل عام من سلاسل من الأعداد (مجموعات من الأصفار والواحدات) التي تعطي الأوامر للحاسب من اجل تنفيذ تعليماته الأولية كل تعليمة على حده .

2 -**لغة المجمع** : هي لغة تستخدم مصطلحات قريبة من اللغة الانكليزية للتعبير عن العمليات الأولية للحاسب ,وقد تم تطوير مترجمات البرامج تسمى بالمجمعات assemblers تحويل البرامج من لغة المجمع إلى لغة الآلة .

3 -**اللغات عالية المستوى** : هي اللغات التي ظهرت لتسريع عملية البرمجة وذلك باستخدام تعليمات تقوم بالعديد من المهام الجوهرية مثل لغة C++ .

تعتبر لغة C++ من أشهر اللغات التي تتمتع بطابع القوة والمرونة لإنتاج أسرع برامج وأداء أفضل . وعلى الرغم من وجود العديد من لغات البرمجة الأخرى إلا أنها تفتقر شمولية لغة C++ وقوتها . فاللغة C++ تتميز بقابليتها على معالجة التطبيقات الكبيرة والمعقدة، والقوة في صيانة البرامج المكتوبة بها مما يوفر وقتاً في تصميم البرامج وتطويرها .

ملاحظة :تدعى البرامج التي تقوم بتحويل النصوص من البرامج مكتوبة بلغات عالية المستوى الى لغة الآلة بالمترجمات

ثانياً.. كيفية كتابة برنامج بلغة (C ++):

الشكل العام للبرنامج هو :

```
#include<iostream.h>   ⟶  المكتبات الحاوية على العمليات الخاصة بالدخل والخرج .
main ( )               ⟶  التابع الرئيسي الذي يبدأ من عنده التنفيذ
{                      ⟶  القوس للدلالة على بداية البرنامج
هنا يتم تعريف البيانات ومن ثم يتم كتابة البرنامج
return 0 ;
}                      ⟶  القوس للدلالة على نهاية البرنامج
```

1 - طباعة نص مؤلف من سطر :

```
// First Program                    // كل الكتابات التي تلي هذه الإشارة ( // ) تسمى تعليق لا يتم تنفيذه
  #include <iostream.h>             //  توجيه ما قبل الترجمة حيث يتم ضم محتوى الملف الرأسي ذو ( .h )
                                    الامتداد الحاوي على العمليات الخاصة بالدخل و الخرج لنص البرنامج

main( )                             //  التابع الرئيسي الذي يبدأ من عنده التنفيذ
{                                   //  بداية البرنامج
   cout << " welcome to c++ "  ;    // تعليمة الطباعة
    return 0 ;                      // إحدى طرق الخروج من التابع
}                                   //نهاية البرنامج
```

Edited with the trial version of
Foxit Advanced PDF Editor
To remove this notice, visit:
www.foxitsoftware.com/shopping

2 – برنامج جمع عددين صحيحين:

```
#include <iostream.h>
 main( )
{
    int x1 , x2 , x3 ;                    //   تعريف المتحولات
    cout << " enter first number "  ;  // تعليمة الطباعة
    cin  >> x1   ;                        // تعليمة قراءة متحول
    cout << " enter second number "  ;
    cin  >> x2   ;
    x3=x1+x2   ;                          // x3 إجراء عملية الجمع والإسناد إلى المتحول الجديد
    cout  << "sum is " << x3   ;          // تعليمة الطباعة المتعددة
    return 0 ;
}
```

ثالثاً.. المتحولات (المتغيرات) Variables:

كل اسم من أسماء المتحولات مثل .... , x3 , x2 ,x1 يتم وضعه في الذاكرة ويعرف باسم name ونمط type وحجم size وقيمة value .

وبالتالي فإن المتحول x1 يملك الاسم x1 والنمط int والحجم 4بايت والقيمة هي حسب القيمة المقروءة .

5	**x1**
10	**x2**
15	**x3**

مواضع المتحولات في الذاكرة

مع ذكر الاسم والقيمة

أنواع المتحولات :

- المتحول المحرفي *char*
- المتحولات الصحيحة *short int , int , long int*
- المتحولات الحقيقية *float , double , long double*
- المتحول المنطقي *bool*

الثابت (ويضاف لنوع المتحول *const*

رابعاً .. قواعد تسمية المتحولات :

بشكل عام يتكون اسم المتحول من مجموعة من الحروف (A ----> Z ,a ----> z) والأرقام ويخضع اسم المتحول للقواعد التالية :

- لا يمكن أن يبدأ برقم .
- لا يحتوي على سلسلة فارغة .
- يجب أن لا يكون من الأسماء المحجوزة في لغة C++ مثل:

true,false,break,puplic,private,static,new,……..,etc

يكتب تعريف المتحول وفق الصيغة :

Data-Type name = initial-value ;

نوع المعطيات اسم المتحول قيمة ابتدائية

إن ذكر اسم المتحول ونوع المعطيات الذي ينتمي إليه هو أمر اجباري في كل تعريف لأي متحول ,أما اسناد قيمة ابتدائية له عند تعريفه فهو أمر اختياري يمكن عدم ذكره في عملية التصريح عن المتحول .

ملاحظة :من الجدير بالذكر أن لغة C++ تفرق بين الحروف الأبجدية الصغيرة والكبيرة

Edited with the trial version of Foxit Advanced PDF Editor
To remove this notice, visit: www.foxitsoftware.com/shopping

يبين الجدول التالي أنواع المتحولات ومجالاتها وحجمها :

نوع المتحول	الحجم	المجال
char	1 byte	-128 to 127
int	4 bytes	-2147483648 to 2147483647
short int	2 bytes	-32768 to 32767
long int	4 bytes	-2147483648 to 2147483647
bool	1 byte	True or false
float	4 bytes	-38+E3.45 to 38-E3.4
double	8 bytes	-308+E1.7 to 308-E1.7
long double	8 bytes	-308+E1.7 to 308-E1.7

خامساً.. المعاملات في لغة C++ :

A. المعاملات الحسابية الأحادية :

المعامل	الاستخدام	الوصف
++	s++	زيادة قيمة المتحول s بمقدار 1, استخدام قيمة s قبل زيادته
++	++s	زيادة قيمة المتحول s بمقدار 1, استخدام قيمة s بعد زيادته
- -	s- -	انقاص قيمة المتحول s بمقدار 1, استخدام قيمة s قبل انقاصه
- -	- -s	انقاص قيمة المتحول s بمقدار 1, استخدام قيمة s بعد انقاصه

B. المعاملات الحسابية الثنائية :

العملية	الاستخدام	الوصف
+	x+y	جمع المتحول x مع y
-	x-y	طرح المتحول yمن x
*	x*y	ضرب المتحول x بالمتحول y
/	x/y	قسمة المتحول x على المتحول y
%	x%y	باقي قسمة المتحول x على المتحول y

C. المعاملات العلائقية :

تستخدم المعاملات العلائقية لمقارنة قيمتين وتحديد العلاقة بينهما والتي تكون النتيجة إما True أو False (يتم تمثيل True بالقيمة 1 و False بالقيمة 0)

العملية	الاستخدام	الوصف
>	x>y	إعادة True في حال كانت x أكبر تماماً من y وFalse إذا لم يكن xأكبر تماماً من y
>=	x>=y	إعادة True في حال كانت x أكبر أو تساوي y وFalse في الحالة غير ذلك.
<	x<y	إعادة True في حال كانت x أصغر تماماً من y وFalse إذا لم يكن xأصغر تماماً من y
<=	x<=y	إعادة True في حال كانت x أصغر أو تساوي y وFalse في الحالة غير ذلك.
==	x==y	إعادة True في حال كانت x تساوي y وFalse في حال عدم المساواة .
!=	x!=y	إعادة True في حال كانت x لا تساوي y وFalse في حال المساواة .

Edited with the trial version of
Foxit Advanced PDF Editor
To remove this notice, visit:
www.foxitsoftware.com/shopping

D. معاملات الاسناد :

نستخدم معامل الاسناد لإسناد القيمة الموجودة على يمين المعامل الى المتحول الموجود على يسار المعامل .

x=5 أسندنا القيمة 5 الى المتحول x .

ملاحظة يمكننا في لغة C++ إجراء عمليات حسابية مع عمليات الاسناد مثل الجمع والضرب والطرح والقسمة كالتالي :

المعامل	شكل الاستخدام	مساوي لـ
+=	x+=y	x=x+y
-=	x-=y	x=x-y
*=	x*=y	x=x*y
/=	x/=y	x=x/y
%=	x%=y	x=x%y

E. العمليات المنطقية :

يتم استخدام العمليات المنطقية مع العمليات التي يكون لها إحدى قيم True أو False وهذه العمليات هي :

العملية	الرمز
AND	&&
OR	||
NOT	!

✓ **جداول الحقيقة لأدوات الربط المنطقية:**

- **النفي "Not":** لتكن A عبارة منطقية إن تطبيق أداة الربط نفي على هذه العبارة سيعطينا عبارة منطقية جديدة رمزها **!A** .

جدول الحقيقة للنفي:

A	!A
0	1
1	0

- **الوصل "And":** A And B= A&&B

جدول الحقيقة للوصل:

A	B	A&&B
0	0	0
0	1	0
1	0	0
1	1	1

إن العملية A AND B دوماً خاطئة (أي قيمة الحقيقة لها تساوي الصفر) إلا في حالة واحدة فهي صحيحة (أي قيمة الحقيقة لها تساوي الواحد) وذلك في حال كلا العبارتين A,B صحيحتين.

- **الفصل "OR":** A OR B= A||B

جدول الحقيقة للوصل:

A	B	A||B
0	0	0
0	1	1
1	0	1
1	1	1

إن العملية A OR B دوماً صحيحة (أي قيمة الحقيقة لها تساوي الواحد) إلا في حال كلا العبارتين A,B خاطئتين.

Edited with the trial version of
Foxit Advanced PDF Editor
To remove this notice, visit:
www.foxitsoftware.com/shopping

سادساً.. أولوية العمليات الحسابية :

عندما نطبق العمليات الحسابية في العبارات الحسابية فإنها تطبق حسب ترتيب معين تبعاً لقواعد الأولية بين العمليات وهي :

1) ( ) الأقواس :يتم تنفيذ الأقواس الداخلية أولاً أما إذا كان لدينا مجموعة من الأقواس جانب بعضها البعض وعلى نفس المستوى عندها يبدأ الحساب من اليسار إلى اليمين .
2) %,/,*: إذا وجدت على نفس المستوى فإنها تنفذ من اليسار الى اليمين .
3) -,+ : تنفذ في النهاية ,إذا وجدت على نفس المستوى فإنها أيضاً تنفذ من اليسار إلى اليمين .

سابعاً.. سلاسل الهروب :

سلسلة الهروب	المعنى
\n	سطر جديد أي وضع المؤشر في بداية السطر التالي
\t	تحريك المؤشر مسافة جدولية أفقية (tab)
\"	تستخدم لطباعة علامة الاقتباس "

```
# include <iostream.h>
main ( )
{
cout <<"welcome to c++ \ n Hello World " ;

    return 0;
}
```

حرف الهروب

والخرج يكون :

```
welcome to c++
Hello World
```

ثامناً.. أمثلة على العمليات :

✓ المعاملات الحسابية الأحادية (معاملات الزيادة والنقصان) :

```
 # include <iostream.h>
main ( )
{ int a=5;
cout<<a<<"\n";
cout<<a++<<"\n" ;
cout<<a<<"\n";
return 0;
}
```

يكون الخرج :5

```
5
6
```

Edited with the trial version of Foxit Advanced PDF Editor
To remove this notice, visit: www.foxitsoftware.com/shopping

```
# include <iostream.h>
main ( )
{ int a=5;
cout<<a<<"\n";
cout<<++a<<"\n" ;
cout<<a<<"\n";
return 0;
}
```

يكون الخرج :5
6
6

✓ <u>أولوية العمليات :</u>

- (5+2)*3+(2*6)/4-1
  =7 * 3 + 12 / 4 – 1
  =21 +3 – 1 =23
- 4 * 8 – 5 + 2 * 6 / 3
  =32 – 5 + 12 / 3
  =32 – 5 + 4=31

<u>المعاملات المنطقية :</u>

- cout<< ((4 > 3) && (3==2)); -----> 1 && 0 ----> 0 النتيجة المطبوعة النهائية
- cout<<(! (4 < 3) || (3==2)); -----> !(0) || 0---> 1 || 0
  وبالتالي النتيجة النهائية المطبوعة 1 .

Edited with the trial version of
Foxit Advanced PDF Editor
To remove this notice, visit:
www.foxitsoftware.com/shopping

تمارين :

1. اكتب برنامج يأخذ كدخل ثلاث أعداد صحيحة من لوحة المفاتيح ثم يطبع مجموعها ومتوسطها وناتج جداؤها.

```
# include <iostream.h>
main( )
{
int a,b,c ;
cin >> a >> b >> c;
cout << " sum is " << a+b+c << " \n" ;
cout << " average is " << (a+b+c)/3 << " \n";
cout << " product is " << a*b*c;
return 0;}
```

2. اكتب برنامج يقرأ نصف قطر دائرة ثم يطبع قيمة قطر الدائرة ,محيطها , مساحتها.

```
# include <iostream.h>
main( )
{
float  r ; // تعريف متحول حقيقي
float const p=3.14; // تعريف متحول حقيقي واسناد قيمة ابتدائية له
cin >> r ;
cout << r * 2 << " \n" ;
cout << 2*p*r << " \n";
cout <<p*r*r;
return 0;}
```

3. اكتب برنامج يطبع شكل مستطيل باستخدام سلاسل الهروب .

```
# include <iostream.h>
main()
{
cout << " ********** \n"<<" * \t" << " * \n" ;
cout << " * \t" <<" * \n";
cout << " * \t" <<" * \n";
cout << " * \t" <<" * \n";
cout << " * \t" <<" * \n";
cout << " ********** \n";
return 0;
}
```

4. اكتب برنامج يقوم بحساب مساحة ومحيط مستطيل (قيم الطول والعرض يتم ادخالها من لوحةالمفاتيح).

```
#include<iostream.h>
main()
{float   area,length,width,cir;     // أسماء للمتحولات
cout<<"Enter the length: ";    //length :الطول
cin>>length;
cout<<"Enter the width: ";     //width :العرض
cin>>width;
area=length*width;             //المساحة
cir= 2*( length+width);        //المحيط
cout<<"\n Area= "<<area<<endl;       // تعني انتقل الى سطر جديد endl
cout<<" circumference= "<<cir<<endl;
return 0;}
```