



## قسم هندسة تقنيات الحاسوب

مادة البرمجة (C++)

المرحلة الاولى

2018-2017

المحاضرة الثالثة

م.م غيداء احمد علي

مثال: اكتب برنامج بلغة C++ على شكل E الذي يتكون من حرف x:

```
# include<iostream.h>

void main ()
{
    cout<<"xxxxxxxx" <<endl;
    cout<<"x "      << endl;
    cout<<"x "      << endl;
    cout<<"xxxxxxxx" <<endl;
    cout<<"x "      << endl;
    cout<<"x "      << endl;
    cout<<"xxxxxxxx" <<endl;
}
```

مثال: اكتب برنامج يقوم بحساب مساحة الدائرة بدلالة نصف القطر:

```
# include <iostream.h>

void main ()
{ float R ,area ;
float pi =3.14;
cout<<"enter radius of circle";
cin>>R;
area=pi*R*R;
cout <<area;
}
```

**إضافة التعليقات:** التعليقات مهمة جدا بجانب الأسطر البرمجية حتى في ما بعد إذا أردنا فهم البرنامج الذي كتبناه في وقت مضى مجرد نقرأ التعليقات بجانب الأسطر نفهم ما يعني الكود في C++ نضع التعليقات بعد هذه العلامة //

```
#include<iostream.h> // declaration of Lab we will use it
```

## الأسبقيات وطرق معاملتها:

- يبدأ تنفيذ العمليات من اليسار إلى اليمين
- يقارن كل عمليتين معا العملية التي لها أسبقية أعلى تنفذ أولا
- إذا تساوت اسبقيتان يبدأ بالتنفيذ من اليسار إلى اليمين
- ما بين الأقواس ينفذ أولا لأنه أعلى أسبقية من غيره ( وما في داخل ما بين الأقواس إي إذا كان أكثر من عملية بين الأقواس تعامل حسب الأسبقية الذي أسبقيته أعلى ينفذ أولا)

في جداول الرموز (في المحاضرة الثانية ) وضعنا أمام كل رمز الأسبقية الخاصة به على سبيل المثال إذا جاءت عملية ضرب وعملية جمع في تعبير واحد فيكون للضرب أسبقية على الجمع لأن أسبقيته الضرب هي (3) وأسبقية الجمع هي (4) إي الضرب أعلى أسبقية (الرقم الأقل أعلى أسبقية من جدول الاسبقيات) وينفذ قبله وهذا هو من أصول عمل المترجم لذلك يجب فهم الأسبقيات حتى لا تخطئ في طريقة تحليل التعبير الرياضي لأي مسألة. في المثال التالي نشاهد الأسبقيات: ( a-b/d)

### مثال 1:

يبدأ المترجم تنفيذ العمليات من اليسار إلى اليمين

→

$$\text{Reslt} = a - c / d$$

يبدأ بمقارنة كل عمليتين رياضيتين معا وأيها له أسبقية أعلى تنفذ أولا

إذا كان (a=5,c=10,d=2) فيكون تسلسل تنفيذ الخطوات

1.  $\text{Reslt} = 5 - 10 / 2$

عند مقارنة عملية القسمة و الطرح فوجد انه القسمة له أسبقية أعلى من الطرح لذلك ستنفذ القسمة أولا  
فيقسم (10/2=5)

2.  $\text{Reslt} = 5 - 5$

تنفذ عملية الطرح بشكل اعتيادي لأنها أخر عملية رياضية متبقية قبل المساواة ويكون الناتج هو صفر  
وأخر عملية ستنفذ هي المساواة فتصبح قيمة

( Reslt=0)

لو لاحظنا كيف تغلبت عملية القسمة على عملية الطرح في المثال وربما نحن كنا نريد أن تنفذ عملية الطرح أولا لكن المترجم نفذ حسب الاسبقية لذلك يجب مراعاة التعبير والاسبقيات واستخدام الاقواس للتخلص من الاسبقية الاعلى لان الاقواس اعلى اسبقية من الجميع.

**ملاحظة مهمة:** عند مقارنة عمليتان ويجد أن الاسبقيتان متساويتان سينفذ من اليسار إلى اليمين أول عملية تقع في اليسار تنفذ أولاً.  
**مثال 2:**

$$y = \frac{5 + A}{D} - \frac{B}{C}$$

ونحن نريد أن نقصد في تعبيرنا أن (5,A) يجمعون أولاً ثم تقسم النتيجة علي (d) لكن الواقع غير حسب الأسبقيات أن القسمة لها أسبقية على الجمع لذلك سننجز القسمة أولاً أي يقسم (5/D) وتجمع النتيجة مع (A) ويكون الناتج غير صائب.....!

لاحظ تسلسل العمليات بدون أقواس وحسب التعبير ( Y= 5+A /D - B/C )

$$y=5+A/d-b/c$$



يبدأ بمقارنة كل عمليتين رياضيتين معا وأيها له أسبقية أعلى تنفذ أولاً

إذا كان ( c=2,A=4,b=4,d=4 ) فيكون تسلسل تنفيذ الخطوات

1.

$$y=5 + 4 / 4 - 4 / 2$$



عند مقارنة عملية الجمع والقسمة فوجد انه القسمة له أسبقية أعلى من الجمع لذلك سنقسم أولاً فيقسم (4/4=1)

2.

$$y=5 + 1 - 4 / 2$$



عند مقارنة عملية الجمع مع الطرح وجد أن الاسبقيتان متساويتان لذلك سننجز من اليسار إلى اليمين ومن اليسار أول عملية تقع هي الجمع لذلك سينفذ الجمع أولاً (5+1=6)

3.

$$y=6 - 4 / 2$$



عند مقارنة عملية القسمة مع الطرح وجد أن الأسبقية القسمة أعلى لذلك سننجز القسمة أولاً (4/2=2)

4.

$$y = 6 - 2$$

تنفذ عملية الطرح بشكل اعتيادي لأنها آخر عملية رياضية متبقية قبل المساواة ويكون الناتج هو (4) وأخر عملية ستنفذ هي المساواة فتصبح قيمة

$$y = 4$$

رياضيا لحل هذه المشكلة نضع أقواس حول العمليات التي لها أسبقية أقل ونريدها أن تنفذ أولا وهو بالضبط ما نعمله برمجيا أيضا نستخدم أقواس حول عملية الجمع لذلك سوف ينفذ الجمع ثم يقسم النتيجة على (D) وبما أن الناتج مطروح من عملية قسمة سوف ينفذ القسمة B على C ويطرح النتيجتين وتكون النتيجة صائبة؟

ويكتب برمجيا هكذا

$$Y = (5+A) / D - B / C ;$$

☒ ملاحظة: الأسبقيات بالنسبة للأدوات Bowties والأدوات المنطقية هي نفس طريقة في العمليات الرياضية أيضا الذي له أسبقية أعلى ينفذ أولا

مثال

True && False || True

جد النتيجة.....؟

1. True && False || True

نعم انه and له أسبقية أعلى من or لذلك سينفذ أولا (True And False=False)

2. False || True

الآن عملية or بشكل اعتيادي ( False or True=True ) فالناتج يكون (True)

نفس الطريقة بالنسبة لباقي الأدوات تنفذ حسب الأسبقية

$$(3+2) \&\& (4*2) \ || \ (4\%2)$$

جد النتيجة.....؟

$$(3+2) \&\& (4*2) \ || \ (4\%2)$$

1.

بما انه موجود تعابير رياضية وعمليات منطقية سينفذ العمليات الرياضية التي بين الأقواس أولاً أول عملية سينفذها هي الضرب لان أسبقيتها أعلى من الجمع وباقي القسمة هي  $(4*2)$  وتساوي ثمانية

$$(3+2) \&\& 8 \ || \ (4\%2)$$

2.

سينفذ باقي القسمة لأن أسبقيته أعلى من الجمع وباقي قسمة  $(4\%2)$  هو صفر

$$(3+2) \&\& 8 \ || \ 0$$

3.

سينفذ عملية الجمع  $(2+3)$  يساوي خمسة

$$5 \&\& 8 \ || \ 0$$

4.

الآن يقارن عملية  $(and)$  وعملية  $(or)$  نعم انه  $and$  له أسبقية أعلى من  $or$  لذلك سينفذ أولاً في عمليات المقارنة أي رقم غير الصفر يعتبره واحد سوا كان موجب أو سالب إذن تكون المقارنة بشكل التالي

$$(5 \&\& 8)$$

$$(1 \&\& 1=1)$$

النتيجة هي واحد

$$1 \ || \ 0$$

5.

آخر عملية مقارنة هي  $(or)$  بين الواحد والصفر والنتيجة هي واحد

$$(1 \ || \ 0)=1$$



## دوال الرياضية في مكتبة <math.h>:

تستخدم دوال هذه المكتبة في حل العمليات الرياضية كإيجاد جيب أو جيب تمام أو قيمة مطلقة وغيرها من الدوال لتسهيل عملية حساب هذه القيم وهذا شرح لبعض دوالها

1. **(sin)** تستخدم هذه الدالة لإيجاد جيب الزاوية بالنظام ال (rad) لذلك لإيجاد جيب الزاوية بالنظام (deg) فقط تضرب قيمة الزاوية ب  $\pi/180$

```
الكود (إيجاد جيب الزاوية 90)
Float x;
X=sin(90*(3.14/180));
```

2. **(cos)** تستخدم هذه الدالة لإيجاد جيب تمام الزاوية بالنظام ال (rad) لذلك لإيجاد جيب الزاوية بالنظام (deg) فقط تضرب قيمة الزاوية ب  $\pi/180$

```
الكود (إيجاد جيب تمام الزاوية 90)
Float x;
X=cos(90*(3.14/180));
```

لإيجاد بقية الدوال المثلثية جميعها تكون مشتقة من جيب وجيب تمام. أي إننا نحول أي دالة مثلثية إلى جيب أو جيب تمام حسب تحويلها بالدوال المثلثية. مثلا لإيجاد الضل الزاوية فقط نقسم جيب على جيب تمام الزاوية

```
عملية رياضية لإيجاد ضل 90
Float Tanx;
Tanx =sin (90*(3.14/180) ) / cos(90*(3.14/180));
```

وكذلك بقية الدوال المثلثية بنفس الطريقة

3. **(pow)** تستخدم هذه الدالة لإيجاد قيمة رقم مرفوع إلى أس (مثلا  $3^2=9$ ). طريقة تمثيل هذه الدالة **X=pow(number,hispower);**

حيث أن **number** هو الرقم و **hispower** هو الأس المرفوع له  
مثال: لو كان لدينا (3^9) أي ثلاثة مرفوع لأس تسعة فيكتب برمجيا باستخدام هذه الدالة هكذا

```
الكود
X=pow(3,9);
```

4. **(abs)** هي القيمة المطلقة للرقم

```
الكود
X=abs(-3); // x=3
```

5. **(sqrt)** هي دالة تستخدم لإيجاد جذر الرقم

```
الكود
X=sqrt(25); // x=5
```