

Syria Math

البرمجة و الخوارزميات 2



الدكتور : سمير جعفر

المحاضرة : الثانية

التاريخ : ٢٠١٦/١٠/١١

Web: www.syriamath.net

group: Improve our mathematics



في المحاضرة السابقة قمنا بمراجعة لموضوع الدوال في C++ ، ومنه دخلنا إلى مفهوم جديد هو التحميل الزائد للدوال ، ونأخذ فكرة عن الدوال العودية..

التحميل الزائد للدوال : هو مفهوم خاص باللغات المتقدمة مثل C++ وجافا ، وهو عبارة عن إمكانية كتابة أكثر من دالة (**تعريف أكثر من دالة**) لها الاسم نفسه في البرنامج نفسه ، حتى تتمكن من القيام بذلك يجب ان تكون الدوال مختلفة عن بعضها البعض بالترويسة (**بقائمة الوسطاء أو الاسم**) وبالتالي حتى نكتب دالتين أو أكثر بنفس الاسم بالبرنامج نفسه يجب أن نغير إما بنوع الوسطاء أو عدد الوسطاء أو ترتيبها إذا كانت من أنواع مختلفة ..

تمرين:

اكتب دالة عودية لحساب جداء عددين طبيعيين موجبين

$$a * b = a * (b - 1) + a$$

الحل:

```
int mult (int a , int b) {
```

```
    if (a == 1) return b;
```

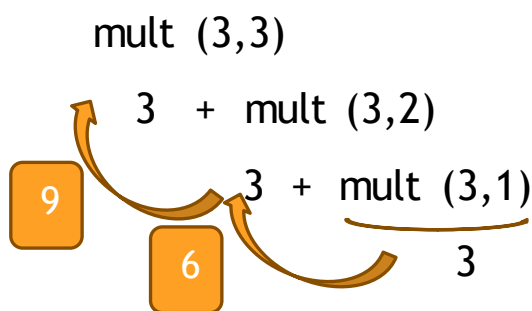
```
    else if (b == 1) return a;
```

```
    else return (a + mult(a, b - 1));
```

```
}
```

Syria Math

شرح تنفيذي:



ملاحظة: mult هي عبارة عن متحول مثل x

وظيفة:



١. أكتب دالة عودية تقوم بحساب قسمة عددين

```
#include <iostream>
int func( int x, int y)
{
if(x>y){

if(y==0)return 0 ;
else return (1+func(x-y,y));}
else if (y>x)
if (x==0) return 0 ;
else return (1+func(y-x,x));
else return 1;}
int main(){
int x,y;
cin>>x>>y;
cout<<func(x,y);
return 0;
}
```

٢. أكتب دالة لحساب الجداء الداخلي لشعاعين بشكل العودي

المؤشرات (pointers) :

ليس لجميع اللغات مفهوم المؤشرات لأن بعض المصممين قرروا أن المؤشرات موضوع صعب ويؤدي بالأشخاص لأخطاء بعمل البرنامج واستخدام الذاكرة ، أما الحقيقة فهو موضوع أساسي بالنسبة لكل المبرمجين لذلك أيضاً هنالك الكثير من اللغات بقيت محافظة عليه ، بدأ الموضوع من لغة الباسكال ، لغة ال C ، لغة ال C++ ..

- سنعود إلى تعريف متحول عادي، للتصريح عن متحول من النوع الصحيح :

`int x;`

يحجز خانة في الذاكرة

`x = 5;`

اسندنا ل x القيمة 5

المؤشر :

هو متحول ، وقيم هذا المتحول هي عناوين ذاكرة (أعداد ست عشرية) .

- الذاكرة هي عبارة عن خانة ثنائية وكل خانة من خانات الذاكرة لها رقم وهذه الخانة مرقمة حسب نظام الست عشري لأن عدد خاناتها كبيرة جداً ، الرقم المخصص لكل خانة نسميه عنوان هذه الخانة..



هيكلية المؤشر :

يمكن التصريح عن مؤشر يؤشر على نوع معين كما يلي :

; اسم المؤشر * النوع المؤشر عليه

`int * p;`

P هو مؤشر يؤشر على متحول من النوع الصحيح حصراً

اي لا يمكنه التأشير على متحول من نوع آخر .

تنويه : اسم المؤشر يعني بمتحول من نوع المؤشر..

مثال : نريد تعريف مؤشر على المتحول من النوع الصحيح :

`int * p;`

تنويه : P هو متحول يؤشر على متحول من النوع الصحيح أو اختصاراً هو مؤشر على متحول

من النوع الصحيح ..

حيث : قيم المتحول P هي عبارة عن عناوين خانات ذاكرة تتسع لمتحول من النوع الصحيح

`int x = 5 ;`

`int * p ;`

كيف سنجعل p تؤشر على x أو ما هو عنوان هذه الخانة في الذاكرة ؟

ذلك عن طريق معامل ((هذا المعامل خاص بإعطاء عنوان الخانة المحجوزة ل x)) ويسمى
مُعامل العنوان.(ونرمز له $\&$)

`p = &x;`

P يؤشر على x

أي اذا وضع امام أي متحول يعطي العنوان أو المكان المحجوز لهذا المتحول في الذاكرة .

$\&x \Leftarrow$ رقم ست عشري يمثل العنوان أو الخانة المحجوزة ل x في الذاكرة .

من الأخطاء الشائعة :

`double * p;`

`int y;`

`p = &y;`



يجب أن يكون المتحول Y من نفس النوع الذي يدل عليه المؤشر .

كيف يمكن الوصول إلى محتويات الخانة من خلال عنوانها ؟؟

معامل المحتوى:

يستخدم مع المؤشرات للحصول على القيمة الموجودة في المكان المؤشر عليه . (نرمز له بـ *)

```
int * p;
```

```
int x;
```

```
p = &x;
```

```
*p = 5;
```

سيطبع 5 \rightsquigarrow `cout << x;`

حيث $*p = 5$ هو محتوى المكان المؤشر عليه P



"انتهت المحاضرة" Syria Math