

Syria Math

البرمجة و الخوارزميات 2



الدكتور : سمير جعفر

المحاضرة : الأولى

التاريخ : ٢٠١٦/١٠/٤

Web: www.syriamath.net

group: Improve our mathematics



بداية من الأفضل أن نستذكر بعض الأمور المهمة التي مرت معنا في مقرر البرمجة ١ لاسيما في ما يخص الدوال نظراً لأهميتها في مقررنا :

مراجعة عامة عن الدوال في C++ :

الدالة: هي عبارة عن برنامج جزئي، والبرنامج بشكل كامل في C++ يتكون من عدة دوال وقد تناولنا في الفصل الماضي مقدمة بخصوص الدوال أنه عندما يكون لدينا تكرار لتعيين الحساب أكثر من مرة بالبرنامج نستخدم الدوال (مثلًا لدينا بالبرنامج حساب!n.....)

ملاحظة:

الدوال التي تحتاج وسطاء ، حيث أن الوسيط هو بمثابة النافذة التي تتواصل فيها الدالة مع العالم الخارجي (مثلًا : دالة العاملية تحتاج وسيط وهو وسيط واحد عدد صحيح n مثلًا كلما أعطينا قيمة ل n تعطينا الدالة قيمة العاملية له) الدوال التي لا تحتاج معطيات مثل دالة الطباعة (مثلًا : طباعة العبارة "أهلاً بكم")

الشكل العام للدالة :

(.....، اسم الوسيط الأول، نوع الوسيط الأول) اسم الدالة نوع ارجاع الدالة

```
{
; العمليات التي تجريها الدالة
return قيمة
}
```

ملاحظات:

- ١- اسم الدالة مع قائمة الوسطاء تسمى (ترويسة) أو توقيع الدالة .
- ٢- لا يمكن للبرنامج الواحد أن يحوي دالتين من نفس الترويسة .

فوائد تعليمة return :

- ❖ الفائدة الأولى : دلالة على انتهاء تنفيذ الدالة.
- ❖ الفائدة الثانية : أنها تقوم بإرجاع النتيجة (القيمة) الموجودة بعدها وقبل الفاصل المنقوطة للبرنامج الرئيسي..

ترجمة قيمة x \Longrightarrow return x

ترجمة القيمة 0 \Longrightarrow return 0

ترجمة القيمة 15 \Longrightarrow return 15



❖ ملاحظة :

نسمي الدالة بهذا الاسم لأنه تشبه الدالة الرياضية حيث أن القيمة الناتجة هي قيمة وحيدة أي أن الدالة لا يمكن أن ترجع إلا قيمة واحدة فقط ، ولكن في مراحل متقدم سنرى أن هذه القيمة قد تكون بسيطة وقد تكون مركبة ، يمكن أن ترجع الدالة قيمة متحول واحد صحيح أو محرفي أو بولياني، ونستطيع إرجاع متحول مركب (قيمة - مصفوفة) هي نتيجة واحدة ولكنها تحوي أجزاء مختلفة ، لكن الدالة لا تستطيع إرجاع عدد حقيقي وصحيح في نفس الوقت.....

❖ يجب أن نميز :

متحول int x;

دالة int x ();

تنويه: حتى عندما لا يكون هناك وسطاء للدالة يجب وضع قوسين () لنميز أنها دالة..

دالة حساب ال n! :

```
long int fact (int n)
{ long int r = 1;
for(int i = 2; i <= n; i++)
r = r * i;
return r;
}
```

Syria Math

ملاحظات هامة:

- دالة العامل تأخذ عدد طبيعي وترجع عدد طبيعي ولكن لا يوجد عدد طبيعي بالبرمجة لذلك استخدمنا int (صحيح).
- علينا أن نفكر بالحل بطريقة حل رياضية ومن ثم نفكر كيف تنفذ ذلك بتعليمات البرمجة (أعدّ الخوارزمية أولاً).
- بدأنا من $i = 2$ لأن $r = 1$ فليس علينا الضرب بالعدد (1) مرة ثانية ، لكن إذا بدأنا من $i = 1$ فليس خاطئ ولكن سنضيف خطوة بدون داعي .
- دالة العامل معرفة على الأعداد الطبيعية لذلك لا يجب أن يكون الوسيط سالب لأننا عندما ندخل عدد سالب سيكون الناتج (1) وهذا خاطئ إلا عند 0!



الخوارزمية التكرارية : أي طريقة حل تعتمد على الحلقات التكرارية نسميها طريقة تكرارية أو خوارزمية تكرارية.

الخوارزمية العودية (الدوال العودية) recursive : نقول عن هدف ما إنه عودي إذا عُرف كلياً أو جزئياً بدلالة نفسه.

مثال:

- ١- الصفر هو عدد طبيعي وكل عدد يلي عدد طبيعي هو عدد طبيعي (عرّف العدد الطبيعي بدلالة العدد الطبيعي نفسه "كلياً")
- ٢- العامل $n! = n(n - 1)!$ (عرّف جزئياً بدلالة نفسه)

$$n! = \begin{cases} 1 & : n = 0 \\ n(n - 1)! & : n > 0 \end{cases}$$

دالة العامل بطريقة عودية :

```
long int fact (int n){
if (n == 0) return 1;
else return (n * fact(n - 1));
}
```

ملاحظة : (if (n == 0) return 1;) : نسميها حالة توقف استدعاء التابع لنفسه..

Syria Math شرح عمل الدالة:

لنأخذ العدد ٤ كمثال:



- العمق العودي هو عدد مرات استدعاء التابع لنفسه (recursive dept.)

ملاحظة: بمثالنا السابق عدد مرات استدعاء الدالة لنفسه هو ٤.

الطريقة العودية : تستهلك ذاكرة كبيرة لو كانت الاعداد ضخمة مثل (المليار .مليون....الخ)

لذلك البعض ينظرون لها على أنها ثغرة برمجية ولا تستخدمها بعض اللغات البرمجية لكنها جيدة إذا استخدمت بذكاء.



- خاصية إمكانية تسمية أكثر من دالة للاسم نفسه على أن نغير في قائمة الوسطاء من حيث العدد أو النوع أو الترتيب تسمى التحميل الزائد للدوال .
بمعنى أنه يمكن أن نعرف دالتين لهما نفس الاسم على أن نغير في قائمة الوسطاء و هذا التغيير إما بترتيب الوسطاء أو أنواعهم أو عددهم.

1. int f (int x)
2. int f (double x)
3. double f (int x)

فهنا في الدوال السابقة يجوز تعريف الدالتان ١ و ٢ بنفس البرنامج وأيضا ٢ و ٣ لكن لا يجوز كتابة ١ و ٣ لان لديهم نفس الترويسة تماما.

- لا يمكن كتابة دالتين لهما الترويسة نفسها حتى لو كان لهما نوعي إرجاع مختلفين .
- يبرهن أيضا أن كل طريقة عودية ممكن ان تحل بطريقة تكرارية.

وظيفة :

اكتب دالة عودية تقوم بحساب أعدادا فيبوناتشي (fibonacci) .

$$fibonacci(n) = \begin{cases} 0 & : n = 0 \\ 1 & : n = 1 \\ fibonacci(n - 1) + fibonacci(n - 2) & : n > 1 \end{cases}$$

الحل:

```
#include<iostream.h>
Long int fibo (int n){
if (n==0) return 0 ;
else if (n==1) return 1 ;
else return( fibo(n-1)+fibo(n-2));
}
int main (){
تعليمات البرنامج
return 0 ;
}
```

Syria Math

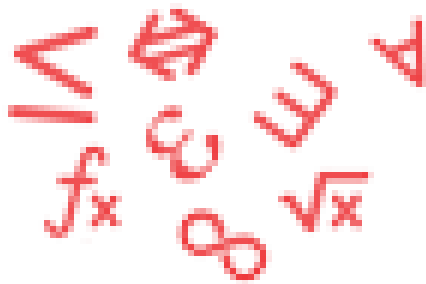


إلّكم لمحة عن أعداد فيوناتشي :

في الرياضيات متتالية فيوناتشي أو أعداد فيوناتشي (fibonacci number) نسبة إلى ليونارد فيوناتشي ، هي الأعداد التي توجد في المتتالية التالية :
 (0,1,1,2,3,5,8,13,21,34,55,89, , ...)
 بتعريفها فإن أول أعداد فيوناتشي هما العددين (1,0) ويكون كل عدد هو ناتج مجموع العددين السابقين له ...

"انتهت المحاضرة"

إعداد : محمد فليون & محمد فهمي قاضي



Syria Math