

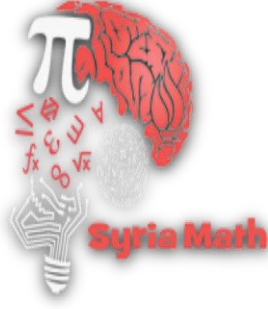
28-10-2018

عملي

◀ دكتور المادة: سمير جعفر

◀ المحاضرة: الثانية

◀ عنوان المحاضرة: السجلات



**المحتوى العلمي :** أهلاً بكم أصدقائي سندرس في هذه المحاضرة :

١- مفهوم الدوال العودية

٢- السجلات

**مراجعة مفهوم الدوال العودية :**

نقول عن هدف ما انه عودي اذا عرف كلياً او جزئياً بدلالة نفسه "أي هو استدعاء الدالة نفسها" بمعنى ان الدالة سوف تستدعي نفسها بنفس كتابة الدالة.

✓ الصيغة العامة للدوال العودية:

```
return – type  function – name (list of parameters){
statement(1);
statement(2);
.
.
function – name(....);
return value;
}
```

مثال: العامل لاي عدد طبيعي معرف بالشكل:

$$n! = n * (n - 1) * ... * 2 * 1$$

$$\Rightarrow f(n) = n! = \begin{cases} 1 & ; n = 0 \\ 1 * 2 * 3 * \dots * n(n-1) & ; n > 0 \end{cases}$$

**السجلات:**

**التعريف:** هو نوع بيانات مركب يحوي على حقول غير متجانسة بالضرورة .

**الشكل العام للسجلات:**

**struct** { اسم النوع

اسم الحقل الأول نوع الحقل الأول ;

اسم الحقل الثاني نوع الحقل الثاني ;

.

اسم الحقل الأخير نوع الحقل الأخير ;};

لانسي الفاصلة المنقوطة في نهاية السجل

**التصريح عن متحول من نوع السجل (النوع الجديد):**

اسم المتحول اسم السجل ;

• للوصول الى الحقل المطلوب في النوع المركب نضع :

اسم الحقل نقطة اسم المتحول ;

**البرنامج الثالث :**

عرف سجل شخص يحوي اسمه ورقمه الوطني ، ثم اكتب برنامج يقوم بإدخال شخص وطباعة رقمه الوطني .

```
#include < iostream.h >
```

```
struct person {
```

```
char name [30];
```

```
long int id;
```

عرفنا السجل بإعطائه اسم **person** ويتألف من حقلين  
(name) للاسم و(id) للرقم الوطني وهو من النوع (long  
(int) لأن الرقم كبير

❖ يمكننا التعريف عن الاسم باستخدام (string name)  
(; هنا استخدمنا النوع (char) وحجزنا ٣٠ حرفاً من  
الذاكرة لاسم العامل بالشكل (char name [30];)

};

```
int main (){
```

```
person p ;
```

التصريح عن متغير من النوع الجديد

```
cin >> p.name >> p.id;
```

```
cout << "name is:" << p.name << endl << "number id is:" << p.id;
```

طباعة الاسم والرقم الوطني

```
return 0 ;
```

}

```
name is: Ali
number id is: 001256890
```

خرج التنفيذ

نتيجة التنفيذ بالشكل:

على فرض الاسم Ali

أعداد فيبوناتشي رياضيا معرفة بالشكل:

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

اكتب دالة لحساب عدد فيبوناتشي باستخدام الطريقة العودية ثم اكتب برنامج يقوم بإدخال عدد فيبوناتشي ☺

الحل:

```
#include <iostream >
```

```
Using name space std;
```

```
int fib (int n){
```

```
if (n == 0) return 0;
```

```
else if (n == 1) return 1;
```

انشأنا دالة عودية أسميناها fib لحساب عدد فيبوناتشي تستقبل وسيط من نوع صحيح وتقوم بحساب عدد فيبوناتشي للقيمة المرسله لها وترجع قيمته.

```
else return (fib(n - 1) + fib(n - 2));}
```

```
int main()
```

```
{ int n;
```

```
do { cin >> n; }while(n < 0);
```

```
int t;
```

```
t = fib(n);
```

```
cout << t;
```

```
return 0;
```

انتهت العاصفة

”لا حول من كلال المحيطين، انك تستطيع....“ *you can*

إعداد: بيان البوشي \*علا الدالاتي\* مرح غريب