



مقدمة في البرمجة

محاضرات العملي لمقرر

مقدمة في البرمجة

ITGS122

الدرس العملي رقم (5)

اعداد الأستاذاة : هبة الهويجي



مقدمة في البرمجة

• الدوال Functions

الدوال التي استخدمناها في الفصول السابقة مثل (scanf or printf) دوال مبنية في لغة C وهي دوال عامة يستطيع أي مبرمج استخدامها. من مزايا لغة C المرونة في الاستخدام ولذلك يمكن إنشاء دوال مثل الدوال القياسية الموجودة في صلب اللغة لتؤدي وظائف مختلفة أو متشابهة والدالة عبارة عن برنامج صغير (أو مجموعة تعليمات تؤدي غرض معين) يخصص لهذا البرنامج اسم ويتم استدعائه داخل الدالة الرئيسية (main) بهذا الاسم.

ان الغاية من استخدام الدوال هو لعدم تكرار التعليمات داخل البرنامج حيث يتم إنشاء الدالة مرة واحدة ثم يتم استدعائها أكثر من مرة عند الحاجة إليها.

باستخدام الدوال يصبح البرنامج أكثر وضوحًا حيث يأخذ البرنامج الشكل التركيبي فيصبح بالشكل الآتي:

```
#include<filename.h>
```

```
Functions declarations,
```

```
void main ()
```

```
{
```

```
Function1-calling ();
```

```
Function2-calling ();
```

```
.....
```

```
}
```

```
Function1-defination()
```

```
{
```

```
.....
```

```
}
```

```
Function2-defination ()
```

```
{
```

```
.....
```

```
}
```

تعريف الدالة

استدعاء الدالة

هيكلية عمل الدالة

مقدمة في البرمجة

• أنواع الدوال :Functions types

هناك أنواع أخرى من الدوال ونوضحها فيما يلي:

- دوال تعيد قيمة صحيحة .int function
- دوال تعيد قيمة حقيقية .float function
- دوال تعيد عبارة حرفية .string function
- دوال تعيد حرف واحد .char function
- دوال لا تعيد أي قيمة .void function

دوال تعيد قيمة من نوع structure وتسمى struct function

Program returns the sum of two integer numbers

```
1 #include<stdio.h>
2 main()
3 { sum(1,10);
4   sum(15,6);
5   sum(100,25);
6 }
7 void sum(int a ,int b)
8 {printf ("\n%d",a+b);
9 }
10
```

Output



```
11
21
125
```

برنامج لتربيع الرقم المدخل بواسطة الدالة square



مقدمة في البرمجة

```
1 #include <stdio.h>
2 int square(int x)
3 {return (x*x) ;
4 }
5 main ( )
6 {int z ;
7 printf(" Enter value :") ;
8 scanf("%d" ,&z) ;
9 printf ("\n The Square of %d is %d", z, square(z) ) ;
10 }
11 |
12
13
```

Output

```
Enter value :4
The Square of 4 is 16
```

Swap two numbers using the function



مقدمة في البرمجة

```
1 #include<stdio.h>
2
3 void swap(int *,int *);
4 int main()
5 {
6
7     int n1,n2;
8     printf("\n\n Function : swap two numbers using function :\n");
9     printf("-----\n");
10    printf("Input 1st number : ");
11    scanf("%d",&n1);
12    printf("Input 2nd number : ");
13    scanf("%d",&n2);
14
15    printf("Before swapping: n1 = %d, n2 = %d ",n1,n2);
16    //pass the address of both variables to the function.
17    swap(&n1,&n2);
18
19    printf("\n\nAfter swapping: n1 = %d, n2 = %d \n\n",n1,n2);
20    return 0;
21 }
22
23 void swap(int *p,int *q)
24 {
25     //p=&n1 so p store the address of n1, so *p store the value of n1
26     //q=&n2 so q store the address of n2, so *q store the value of n2
27
28     int tmp;
29     tmp = *p; // tmp store the value of n1
30     *p=*q;    // *p store the value of *q that is value of n2
31     *q=tmp;   // *q store the value of tmp that is the value of n1
32 }
```

Output

```
Function : swap two numbers using function :
-----
Input 1st number : 2
Input 2nd number : 4
Before swapping: n1 = 2, n2 = 4
After swapping: n1 = 4, n2 = 2
```

مقدمة في البرمجة

برنامج لإيجاد القيمة المطلقة للرقم المدخل بواسطة الدالة fabs

```
1 #include <stdio.h>
2 float fabs(float x)
3 {
4     if(x >= 0) return (x) ;
5     else
6         return(-x) ;
7 }
8 main ()
9 {
10    float a;
11    scanf("%f", &a) ;
12    printf("\n Absolute value of %f is %f" ,a , fabs(a) ) ;
13 }
```

Output

-1

Absolute value of -1.000000 is 1.000000

الدوال الجاهزة

توجد بلغة سي دوال رياضية جاهزة لا نحتاج ان نبرمجها وهي موجودة داخل مكتبة لغة سي داخل ملف math.h كما في الجدول التالي :

جدول : الدوال الجاهزة

الدالة	الغرض منها	نوع المتغير
abs(x)	القيمة المطلقة للعدد الصحيح	int
fabs(x)	القيمة المطلقة للعدد الحقيقي	double
sqrt(x)	الجذر التربيعي	double
pow(x,y)	الرفع بالقوة	double
sin(x)	جيب الزاوية	double
cos(x)	جيب تمام الزاوية	double
tan(x)	ظل الزاوية	double

مقدمة في البرمجة

double	معكوس جيب الزاوية	asin(x)
double	معكوس جيب تمام الزاوية	acos(x)
double	ظل الزاوية معكوس	atan(x)
double	جيب القطع الزائد	sinh(x)
double	جيب تمام القطع الزائد	cosh(x)
double	ظل القطع الزائد	tanh(x)
double	الدالة الاسية	exp(x)
double	لوغاريتم الطبيعي	log(x)
double	لوغاريتم العشري	log10(x)

برنامج يوضح كيفية استخدام دوال الرفع الاسي والقيمة المطلقة وقيمة جيب الزاوية

```
1
2 #include <stdio.h>
3 #include <math.h>
4 main ( )
5 {
6     double x , y ;
7     printf(" Enter value :");
8     scanf ("%f", &x);
9     y= fabs(x)+pow(x,3)-5*sin(3*x-2) ;
10    printf( " Y=%f " , y );
11 }
12
```

Output

```
Enter value :10
Y=4.546487
```



مقدمة في البرمجة

The strlen() function calculates the length of a given string. It doesn't count the null character '\0' here's an example...

```
1 |  
2 | #include <stdio.h>  
3 | #include <string.h>  
4 |  
5 | int main()  
6 | {  
7 |     // Declare and initialize a character array 'str' with  
8 |     // the string "IT"  
9 |     char str[] = "IT";  
10 |  
11 |     // Calculate the length of the string using the strlen()  
12 |     // function and store it in the variable 'length'  
13 |     size_t length = strlen(str);  
14 |  
15 |     // Print the length of the string  
16 |     printf("String: %s\n", str);  
17 |  
18 |     printf("Length: %zu\n", length);  
19 |  
20 |     return 0;  
21 | }  
22 |
```

Output

```
String: IT  
Length: 2
```



مقدمة في البرمجة

The `strtok()` function is used to split the string into small tokens based on a set of delimiter characters, here's an example..

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main()
5  {
6      char str[] = "This.is.IT";
7      // Delimiters: space, comma, dot,
8      // exclamation mark
9      const char delimiters[] = ",.";
10
11     // Tokenize the string
12     char* token = strtok(str, delimiters);
13     while (token != NULL) {
14         printf("Token: %s\n", token);
15         token = strtok(NULL, delimiters);
16     }
17
18     return 0;
19 }
20
```

Output

```
Token: This
Token: is
Token: IT
```