

Tema 6: Funciones

Oscar Perpiñán Lamigueiro

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

¿Qué es una función?

Una función es un bloque de código que realiza una tarea determinada a partir de unos datos.

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Ventajas

- ▶ Permiten **programación estructurada y abstracta**, sin necesidad de conocer el detalle de la implementación de una tarea concreta.
- ▶ Mejoran la **legibilidad** del código.
- ▶ Facilitan el **mantenimiento** del programa.
- ▶ Permiten **reutilizar código** de manera eficiente (**DRY!**).

¿Cómo se declara una función?

Prototipo de una función:

1. Tipo de valor que devuelve (int, void, ...)
2. Nombre de la función (debe ser un identificador válido y **útil**).
3. Lista de argumentos que emplea, por tipo y nombre (puede estar vacía).

```
tipo nombre_funcion(tipo1 arg1, tipo2 arg2, ...);
```

Ejemplos

```
void printHello(int veces);
```

```
float areaTriangulo(float b, float h);
```

¿Cómo se define una función?

```
// Definicion de la funcion printHello
```

```
// No devuelve nada (void)
```

```
// Necesita un argumento llamado veces,  
// un entero (int), para funcionar.
```

```
void printHello(int veces)  
{  
    int i;  
    for (i = 1; i <= veces; i++)  
        printf("Hello World!\n");  
}
```

¿Cómo se define una función?

```
// Definicion de la funcion areaTriangulo

// Devuelve un real (float)

// Necesita dos argumentos, b y h, reales.

float areaTriangulo(float b, float h)
{
    float area;

    area = b * h / 2.0;

    return area;
}
```

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Estructura de un programa

- ▶ *Puede* incluir directivas de inclusión (`include`).
- ▶ *Puede* incluir directivas de sustitución (`define`).
- ▶ Declaración de funciones (prototipo).
- ▶ Todos los programas tienen al menos una función: `main`.
- ▶ Definición de las funciones.

Directivas de inclusión include

Permiten incluir cabeceras (definiciones) procedentes de otros archivos

```
//Librerías del sistema
#include <stdio.h>
#include <math.h>
//Librerías propias del desarrollador
#include "myHeader.h"
```

Directivas de sustitución `define`

- `define` permite definir símbolos que serán sustituidos por su valor.

```
#include <stdio.h>
//Habitualmente con mayúsculas
//Atención: SIN signo igual NI punto y coma
#define PI 3.141592

int main()
{
    float r = 2.0;
    printf("Una circunferencia de radio %f", r);
    printf(" tiene un area de %f", PI * r * r);
    return 0;
}
```

- `undef` elimina la definición del símbolo.

```
#undef PI
```

Declaración y definición de funciones

```
#include <stdio.h>

// Prototipo de la función (termina en ;)
void printHello(int n);

// Función main
void main()
{
    //Uso de la función en main
    printHello(3);
}

// Definición de la función
void printHello(int n)
{
    int i;
    for (i = 1; i <= n; i++)
        printf("Hello World!\n");
}
```

Definición de
función

Estructura de un
programa

Ámbito de una
variable

Funciones que
llaman a otras
funciones

Funciones en
ficheros

...

Declaración y definición de funciones

```
#include <stdio.h>

// Prototipo de la función (termina en ;)
float areaTriangulo(float b, float h);

// Función main
void main()
{
    float at;
    //Uso de la función en main
    at = areaTriangulo(1, 2);
    printf("%f", at);
}

// Definición de la función
float areaTriangulo(float b, float h)
{
    float area;
    area = b * h / 2.0;
    return area;
}
```

Definición de
función

Estructura de un
programa

Ámbito de una
variable

Funciones que
llaman a otras
funciones

Funciones en
archivos

...

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Variables globales

```
#include <stdio.h>

int gVar = 3; //Variable global

void foo(void);

void main(){
    printf("main (1):\t gVar es %d.\n", gVar);

    foo();

    gVar *= 2;
    printf("main (2):\t gVar es %d.\n", gVar);
}

void foo(void)
{
    gVar = gVar + 1;
    printf("foo:\t gVar es %d.\n", gVar);
}
```

Variables locales

```
#include <stdio.h>

void foo(void);

void main()
{
    int x = 1; // variable local en main
    printf("main (1):\t x es %d.\n", x);

    foo();

    printf("main (2):\t x es %d.\n", x);
}

void foo(void)
{
    int x = 2; // variable local en foo
    printf("foo:\t x es %d.\n", x);
}
```

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Ejemplo

```
#include <stdio.h>
#define PI 3.141592

float eleva3(float x);
float volEsfera(float r);

void main(){
    float radio, vol;
    scanf("%f", &radio);
    vol = volEsfera(radio);
    printf("El volumen es %f", vol);
}

float volEsfera(float r){ //Usa eleva3
    return 4.0/3.0 * PI * eleva3(r);
}

float eleva3(float x){
    return x * x * x;
}
```

Funciones recursivas

```
#include <stdio.h>

int fact(int n);

void main(){
    int x;
    printf("Indica un número:\n");
    scanf("%d", &x);
    printf("El factorial de %d es %d\n", x, fact(x));
}

int fact(int n){
    int res;
    if (n > 1) // Incluye llamada a si misma
        res = n * fact(n - 1);
    else
        res = 1;
    return res;
}
```

Definición de
función

Estructura de un
programa

Ámbito de una
variable

Funciones que
llaman a otras
funciones

Funciones en
ficheros

...

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Motivación

Para poder reutilizar las funciones definidas es conveniente alojarlas en un fichero (o colección de ficheros) que puedan ser incluidos en otros proyectos.

Uso

- ▶ Debe existir un (o varios) fichero(s) .h (cabecera) y un fichero .c (código fuente, implementación de las funciones).
- ▶ Se debe usar `#include "nombre_lib.h"` al comienzo del programa.
- ▶ Hay que compilar conjuntamente (*en un proyecto*).

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Ejemplo (1)

Fichero myLib.h (cabecera)

```
#define PI 3.141592

float eleva3(float x);
float volEsfera(float r);
```

Fichero myLib.c (código fuente)

```
#include "myLib.h"

float volEsfera(float r){
    return 4.0/3.0 * PI * eleva3(r);
}

float eleva3(float x){
    return x * x * x;
}
```

Ejemplo (2)

Programa principal

```
#include <stdio.h>
// Directiva para incluir la librería local
#include "myLib.h"

void main()
{
    float radio, vol;
    scanf("%f", &radio);
    vol = volEsfera(radio);
    printf("El volumen es %f", vol);
}
```

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...

Definición de función

Estructura de un programa

Ámbito de una variable

Funciones que llaman a otras funciones

Funciones en ficheros

...