

Funciones en C



Variables Locales y Globales

Las variables tienen tiempo de vida

Globales

Permanecen activas durante todo el programa.

Locales

Son creadas cuando el programa llega a la función en la que están definidas.

Ejemplo

```
14 #include <stdio.h>
15 #include <stdlib.h>
16
17 int num = 30;
18
19 int main(int argc, char** argv) {
20     int num = 21;
21     printf("%d\n", num);
22 }
```

Output x

ejemplo1 (Build, Run) x

ejemplo1 (Run) x

21

Sintaxis

De una función

```
tipo_funcion nombre_funcion (parametros){  
  
    sentencias  
  
}
```

1

Tipo

2

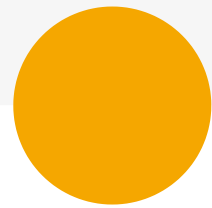
Parámetros

3

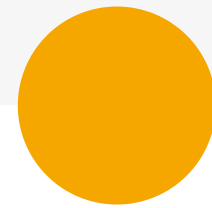
Sentencias



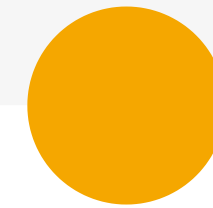
Tipo



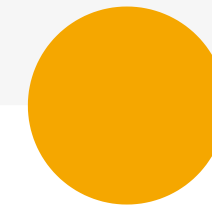
Puede ser de cualquier tipo de los que conocemos.



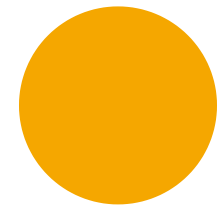
El valor devuelto por la función será de este tipo



Por defecto, devolverá un valor de tipo entero

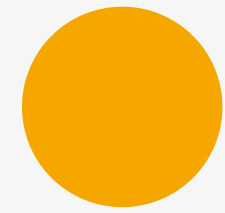


Si no queremos que retorne ningún valor deberemos indicar el tipo vacío



Parámetros

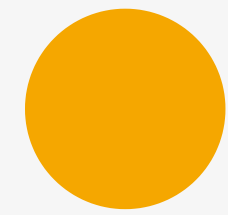
- Son variables locales que reciben un valor.
- Este valor se lo enviamos al hacer la llamada a la función.
- Pueden existir funciones que no reciban argumentos.



Sentencias

Es el conjunto de sentencias que serán ejecutadas cuando se realice la llamada a la función.

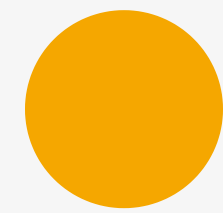
Las funciones pueden ser llamadas desde la función main o desde otras funciones.



Declaración de una función

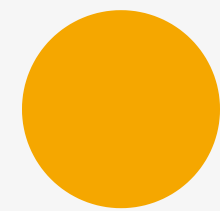
Así como las variables, las funciones también deben de ser declaradas.

Esto es lo que se conoce como **prototipo** de una función.



¿Para qué realizar el prototipo de una función?

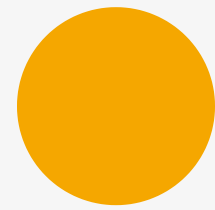
- Para que un programa en C sea compatible entre distintos compiladores
- Para poder llamarla desde main u otra función
- Para verificar que se accede a la función de la manera adecuada con respecto al número y tipo de parámetros, y al tipo de valor de retorno de la misma.



Ejemplo de Prototipo

```
tipo_funcion nombre_funcion (parametros);
```

- En el prototipo de una función no se especifican las sentencias que forman parte de la misma, sino sus características.
- `int cubo(int numero);`
- El nombre de los parámetros es opcional y se utiliza para mejorar la comprensión del código fuente.
- `int cubo(int);`



Return

Retorna un valor o vacio
y finaliza con la
ejecucion de la funcion

En una funcion puede
existir mas de un return

```
return a + b;
```

Debe retornar el mismo tipo de
dato que tiene la funcion



Dudas?

...