

¿Que es un lenguaje de alto nivel?

Se engloban aquí todos los lenguajes de programación que por sus características se asemejan más al lenguaje natural del programador. Algunos de los más conocidos son: FORTRAN, BASIC, Pascal, Modula, C, Python, Java, etc.

Estructura básica de un algoritmo

1: *#include <stdio.h>*

2:

3: *int main()*

4: *{*

5:

6:

7: *return 0;*

8: *}*

Los componentes de un programa en C

Antes de iniciar hay que analizar que es una **función** en un lenguaje de programación.

Una función es una sección independiente de un código de un programa que permite realizar ciertas tareas y son asignadas por un nombre.

Los componentes de un programa en C

- La función principal llamada ***main()*** (líneas 3 hasta 8). Es la parte requerida en un programa en C y consiste del nombre ***main*** seguida por el par de paréntesis conteniendo la palabra ***void*** (***(void)***), y un par de brackets (***{}***). Se puede omitir la palabra ***void*** y funciona el programa.
- La ***#include*** (línea 1) instrucción permite al compilador de C añadir el contenido de un archivo en tu programa durante la ejecución y compilación. Un archivo incluido es un código que contiene información y puede ser usado por tu programa o el compilador. Varios de estos archivos son llamados cabeceras principales, y nosotros usaremos la librería ***stdio.h***
- La frase ***return*** de la línea 7 regresa el valor de 0 al sistema operativo justo antes de que termine el programa.

Palabras reservadas de C

auto	double	int	struct	break	else
long	switch	case	enum	register	typedef
char	extern	return	union	float	short
unsigned	continue	for	signed	void	default
go to	sizeof	volatile	do	if	while
#define	#include

Cada palabra reservada tiene su formato y se analizará cuando se estudien durante el curso.

La frase *printf()*

La frase *printf()* es una función de la librería que despliega información sobre la pantalla. La frase *printf()* puede desplegar simple texto o un mensaje mixto con el valor de una o más variables.

Formato:

printf("Mensaje a desplegar");

printf("Mensaje a desplegar y valor de la(s) variable(s)", variables);

La frase *scanf()*

La frase *scanf()* es una función de la librería que lee datos escritos sobre el teclado y asigna los datos despliega información sobre la pantalla. La frase *printf()* puede desplegar simple texto o un mensaje mixto con el valor de una o más variables.

Formato:

scanf(“valor de la(s) variable(s)”, variables);

Tipo de variable en C

Full Name	Commonly Used Keyword
char	signed char
short	signed short int
int	signed int
long	signed long int
long long	signed long long int
unsigned char	unsigned char
unsigned short	unsigned short int
unsigned int	unsigned int
unsigned long	unsigned long int
unsigned long long	unsigned long long int

Variable Type	Keyword	Bytes Required	Range
Character	char	1	-128 to 127
Short integer	short	2	-32767 to 32767
Integer	int	4	-2,147,483,647 to 2,147,438,647
Long integer	long	4	-2,147,483,647 to 2,147,438,647
Long long integer	long long	8	-9,223,372,036,854,775,807 to 9,223,372,036,854,775,807
Unsigned character	unsigned char	1	0 to 255
Unsigned short integer	unsigned short	2	0 to 65535
Unsigned integer	unsigned int	4	0 to 4,294,967,295
Unsigned long integer	unsigned long	4	0 to 4,294,967,295
Unsigned long long integer	unsigned long long	8	0 to 18,446,744,073,709,551,615
Single-precision floating-point	float	4	1.2E-38 to 3.4E38 ¹
Double-precision floating-point	double	8	2.2E-308 to 1.8E308 ²

¹Approximate range; precision = 7 digits.

²Approximate range; precision = 19 digits.

Para usar nombres con las variables se deben usar estas reglas:

- El nombre puede contener letras (a -> z y A -> Z), dígitos (0 a 9), y el carácter (_).
- El primer carácter debe ser una letra.

Al usar las variables con scanf() y printf() se debe escribir:

- Al usar variables que sean de tipo ***int*** se debe usar ***%d***
- Al usar variables que sean de tipo ***float*** se debe usar ***%f***
- Al usar variables que sean de tipo ***char*** se debe usar ***%c***
- Al usar variables que sean de tipo ***long*** se debe usar ***%d***