



Experiencia N°1

Métodos de Programación
1-2021



- Curso:
13201 METODOS
- Pass:
delunoalnuevesinespacios



CONTENIDO

Introducción a C

Tipos de datos de C

Estructura de un programa en C

Compilar

Escribiendo el primer código

Actividad 1

Función Imprimir

Actividad 2



INTRODUCCIÓN A C [1/4]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Historia:

- El lenguaje C es creado en **1972** por **Dennis Ritchie** y **Brian Kernighan**

- Diseñado en los laboratorios Bell de AT&T

- Implementado por primera vez en un computador DEC PDP-11

- Nació modificando el lenguaje B.

- No, antes de B no estaba el lenguaje A
 - B es la abreviación de BCPL (*Basic Combined Programming Language*)

- Fue utilizado por Ken Thompson para reescribir el sistema operativo Unix en C.





INTRODUCCIÓN A C [2/4]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Características:

- C es un **lenguaje de propósito general**, es decir, se pueden desarrollar aplicaciones de diversas áreas.
- Es un **lenguaje estructurado**, con una variedad de operadores y tipos de datos.
- Es un **lenguaje de nivel medio**, pero se puede codificar a alto nivel, produce código objeto **altamente optimizado**, posee punteros y **capacidad de operaciones sobre las direcciones de memoria**.
- Es un lenguaje muy **portable**, es decir, es independiente de la arquitectura de la máquina y con alguna o ninguna modificación un programa puede correr en una amplia variedad de computadores.
- Es relativamente flexible en la conversión de datos.



INTRODUCCIÓN A C [3/4]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Características:

- Su eficiencia y claridad han hecho que el lenguaje **assembler** casi no haya sido utilizado en Unix.
- El compilador de C es pequeño y tiene un gran poderío debido a sus múltiples bibliotecas.

INLINE ASSEMBLER VS. C++

```
_asm
{
    mov eax, a
    add eax, b
    mov c, eax
};
```

```
c = a + b;
```



INTRODUCCIÓN A C [4/4]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



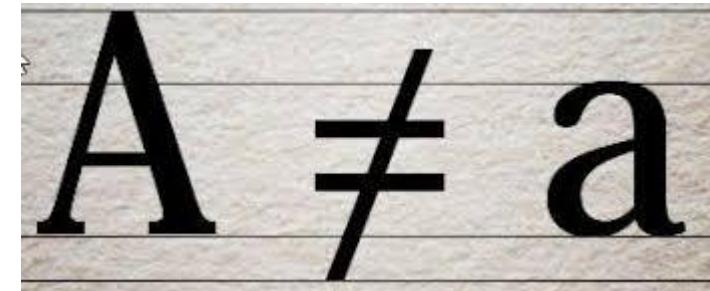
Función Imprimir



Actividad 2

- Características:

- Carece de instrucciones de entrada y salida, de manejo de **strings** (cadenas de caracteres), quedando el trabajo en manos de las bibliotecas provocando con esto algunos problemas de portabilidad.
- Es **case sensitive**, lo que significa que para C las minúsculas y las mayúsculas son distintas.
 - Para C: sumar, Sumar y SUMAR son tres palabras distintas.





TIPOS DE DATOS

Introducción a C

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

- Tipos de datos básicos:

Tipo	Representa	Tamaño en la RAM	Rango de Representación (mínimo-máximo)
char	Caracteres	1 byte (8 bits)	0 a 255
int	Números Enteros	4 bytes (*) (32 bits)	-2.147.483.648 a 2.147.483.647
float	Números en punto flotante	4 bytes (**) (32 bits)	(±) 3.4E-38 a 3.4E38
double	Números en punto flotante de doble precisión	8 bytes (***) (64 bits)	(±) 1.7E-308 a 1.7E308

(*): En compiladores de 32 bits, el tipo int ocupa 4 bytes, los compiladores antiguos de C, representaban a los int sólo en 2 bytes.

Los números reales (float o double) representados en notación científica:
mantisa·10^{exponente}

(**): 32 bits: 1 bit para el signo + 8 bits para el exponente + 23 bits para la mantisa.
(Estándar IEEE 754)

(***): 64 bits: 1 bit para el signo + 11 bits para el exponente + 52 bits para la mantisa.
(Estándar IEEE 754)



ESTRUCTURA DE UN PROGRAMA EN C [1/3]

Introducción a C

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

#include <stdio.h>

/* Mi primer programa. */

int main()

{

 printf("Hola Mundo\n");

 return 0;

}



ESTRUCTURA DE UN PROGRAMA EN C [1/3]

Introducción a C

Tipos de datos de C

Estructura de un programa en C

Compilar

Escribiendo el primer código

Actividad 1

Función Imprimir

Actividad 2

#include <stdio.h>

Es una **directiva**: Le indica al compilador que incluya otro archivo.

/* Mi primer programa. */

int main()

{

 printf("Hola Mundo\n");

 return 0;

}



ESTRUCTURA DE UN PROGRAMA EN C [1/3]

Introducción a C

Tipos de datos de C

Estructura de un programa en C

Compilar

Escribiendo el primer código

Actividad 1

Función Imprimir

Actividad 2

#include <stdio.h>

/* Mi primer programa. */

int main()

{

 printf("Hola Mundo\n");

 return 0;

}

Es una **directiva**: Le indica al compilador que incluya otro archivo.

En este caso se está incluyendo al archivo **stdio.h** (abreviación de standard input output).



ESTRUCTURA DE UN PROGRAMA EN C [1/3]

Introducción a C

Tipos de datos de C

Estructura de un programa en C

Compilar

Escribiendo el primer código

Actividad 1

Función Imprimir

Actividad 2

#include <stdio.h>

/* Mi primer programa. */

int main()

{

 printf("Hola Mundo\n");

 return 0;

}

Es una **directiva**: Le indica al compilador que incluya otro archivo.

En este caso se está incluyendo al archivo **stdio.h** (abreviación de standard input output).

- Están organizadas según el trabajo que hacen en diversos archivos llamados **cabecera (header).h**
- Se aconseja sólo incluir librerías del **C estándar**. Para que su programa funcione en **C para Linux**.
- Se incluye a **stdio.h** para poder usar la función “**printf**”.



ESTRUCTURA DE UN PROGRAMA EN C [2/3]

Introducción a C

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

#include <stdio.h>

/* Mi primer programa. */

int main()

{

 printf("Hola Mundo\n");

 return 0;

}



ESTRUCTURA DE UN PROGRAMA EN C [2/3]

Introducción a C

Tipos de datos de C

Estructura de un programa en C

Compilar

Escribiendo el primer código

Actividad 1

Función Imprimir

Actividad 2

#include <stdio.h>

/* Mi primer programa. */

int main()

{

 printf("Hola Mundo\n");

 return 0;

}

Función principal del programa.

Con esta se inicia el programa.
Es como el bloque principal de lo que vieron en Python (no es lo mismo)

int (de integer = entero), indica que la función entrega un valor entero cuando termina la ejecución. Este valor se puede usar para verificar cómo ha terminado su programa

Se debe retornar este tipo cuando finalice la función.



ESTRUCTURA DE UN PROGRAMA EN C [3/3]

Introducción a C

Tipos de datos de C

Estructura de un programa en C

Compilar

Escribiendo el primer código

Actividad 1

Función Imprimir

Actividad 2

```
#include <stdio.h>
```

```
/* Mi primer programa. */
```

```
int main()
```

```
{
```

```
    printf("Hola Mundo\n");
```

```
    return 0;
```

```
}
```

Delimitadores, indican en dónde comienza “{” y dónde termina una función “}”

Fin de línea de instrucción.



COMPILAR [1/5]

Introducción a C

- C es un lenguaje que debe ser compilado.

Tipos de datos de C

- Compilar significa traducirlo a lenguaje máquina, pero no cualquier máquina, sino en la que se está compilando.

Estructura de un
programa en C

- El ejecutable que se creará podrá funcionar en máquinas de similar arquitectura.

Compilar

- Para compilar es necesario utilizar un programa.

Escribiendo el primer
código

Actividad 1

- Nosotros en el curso utilizaremos **GCC** (*GNU Compiler Collection*)

Función Imprimir

Actividad 2



COMPILAR [2/5]

Introducción a C

- GCC hay que instalarlo

— Para sistemas operativos UNIX:

- Utilizar desde la Shell:

— sudo <gestorRepositorio> install gcc

— sudo apt-get install gcc (Para Debian y Ubuntu)

Estructura de un
programa en C

Compilar

— Para sistemas operativos Windows

- Instalar MinGW

- Configurar variables de entorno

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2



COMPILAR [3/5]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Antes de compilar hay que saber moverse por la consola (Shell)
 - **cd <nombreCarpeta>**: Sirve para abrir la carpeta o la ruta indicada.
 - **cd ..** : Sirve para retroceder al directorio padre de donde se encuentra
 - **ls / dir** : Sirve para listar el contenido de la carpeta actual, ls es para UNIX, mientras que dir para Windows.
 - **mkdir** : Sirve para crear una carpeta en la carpeta actual.
 - **nano**: Sirve editar un archivo sin necesidad de abrirlo desde otro lado.



COMPILAR [4/5]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Para compilar se necesita estar en el lugar donde está nuestro código fuente.
 - Nuestro código fuente será de extensión .c
- Le decimos al compilador que compile nuestro archivo

`gcc -o <Salida> <codigoFuente>`



COMPILAR [4/5]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Para compilar se necesita estar en el lugar donde está nuestro código fuente.
 - Nuestro código fuente será de extensión .c
- Le decimos al compilador que compile nuestro archivo

gcc **-o** <Salida> <codigoFuente>

Indica que se
indicará el
nombre del
ejecutable a crear



COMPILAR [4/5]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Para compilar se necesita estar en el lugar donde está nuestro código fuente.
 - Nuestro código fuente será de extensión .c
- Le decimos al compilador que compile nuestro archivo

gcc -o <Salida> <codigoFuente>

Es el nombre que tendrá el ejecutable.
No se debe colocar la extensión.



COMPILAR [4/5]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Para compilar se necesita estar en el lugar donde está nuestro código fuente.
 - Nuestro código fuente será de extensión .c
- Le decimos al compilador que compile nuestro archivo

gcc -o <Salida> <codigoFuente>

Es el nombre del
archivo del código
fuente.
Debe ir la extensión



COMPILAR [5/5]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Para compilar se necesita estar en el lugar donde está nuestro código fuente.
 - Nuestro código fuente será de extensión .c
- Le decimos al compilador que compile nuestro archivo
 - gcc –o <Salida> <codigoFuente> o
 - gcc <codigoFuente> –o <Salida>
- No es la única forma de hacerlo, pero es la que utilizaremos por el momento



ESCRIBIENDO MI PRIMER CÓDIGO

Introducción a C

#include <stdio.h>

Tipos de datos de C

/* Mi primer programa. */

Estructura de un
programa en C

int main()

{

printf("Hola Mundo\n");

return 0;

}

Escribiendo el primer
código

Actividad 1

Después de escribir y compilar el código, es necesario ejecutarlo, para eso, dentro de la misma consola, escribimos el nombre del archivo ejecutable (sin la extensión)

Función Imprimir

Actividad 2



ACTIVIDAD 1

Introducción a C

- Crear un programa en C que pueda imprimir su nombre por pantalla

Tipos de datos de C

- Compile el programa

Estructura de un
programa en C

- Enseñe el resultado al profesor o ayudante

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2



FUNCIÓN IMPRIMIR [1/4]

Introducción a C

- Analicemos la función **printf**
 - La sintaxis del lenguaje es la siguiente:
`printf(" Mensaje", arg1, arg2,)`

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

Mensaje: Es el texto que se mostrará por pantalla

arg1, arg2: Corresponden a valores de variables que deseamos imprimir.



FUNCIÓN IMPRIMIR [2/4]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

- Para que se imprima el contenido de una variable, debe colocar el símbolo **%**, en el lugar exacto donde desea que aparezca el contenido de esta variable.
- Luego del **%**, debe escribir alguno de los posibles caracteres de conversión.
- Estos caracteres de conversión, se escogen según sea el tipo de dato de la variable.



FUNCIÓN IMPRIMIR [3/4]

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

Éstos son los caracteres de conversión:

- **d ó i** notación decimal. (si su variable es **int**) (en base octal use **o**, en base hexadecimal use **x ó X**)
- **u** notación decimal sin signo. (si su variable es **unsigned int**)
- **c** El argumento se toma como un carácter (**char**).
- **s** El argumento es una cadena de caracteres (**char**).
- **e ó E** El argumento se toma como un valor **double** o **float** (En notación científica)
- **f** El argumento se toma como un valor **float**.
- **lf** El argumento se toma como un valor **double**.



FUNCIÓN IMPRIMIR [4/4]

Introducción a C

Para definir una variable se debe hacer de la siguiente forma:

Tipos de datos de C

Estructura de un programa en C

<tipoDato> <NombreVariable> = <Valor>;

Compilar

Escribiendo el primer código

int numero1 = 5 ;

Actividad 1

Función Imprimir

Actividad 2



FUNCIÓN IMPRIMIR [4/4]

Introducción a C

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

Para definir una variable se debe hacer de la siguiente forma:

<tipoDato> <NombreVariable> = <Valor>;

int

numero1 = 5 ;



FUNCIÓN IMPRIMIR [4/4]

Introducción a C

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

Para definir una variable se debe hacer de la siguiente forma:

<tipoDato> <NombreVariable> = <Valor> ;

int numero1 = 5 ;



FUNCIÓN IMPRIMIR [4/4]

Introducción a C

Tipos de datos de C

Estructura de un
programa en C

Compilar

Escribiendo el primer
código

Actividad 1

Función Imprimir

Actividad 2

Para definir una variable se debe hacer de la siguiente forma:

<tipoDato> <NombreVariable> = <Valor>;

int numero1 = 5;



ACTIVIDAD 2

Introducción a C



Tipos de datos de C



Estructura de un
programa en C



Compilar



Escribiendo el primer
código



Actividad 1



Función Imprimir



Actividad 2

En el código anterior, agregue el año actual, como una variable, y el año de su nacimiento, también como variable.

Ahora el programa deberá mostrar por pantalla un mensaje con el texto:

“Hola <su nombre>, tu naciste el año <año de su nacimiento>, y tu edad es de <su edad> años.”

Por ejemplo:

“Hola Clementino, tu naciste el año 1995, y tu edad es de 23 años.”