

C BASICO

¿Qué es C?

C es un lenguaje de programación de alto nivel desarrollado en el año 1972 por Dennis Ritchie en AT&T Bell Labs. La legibilidad, facilidad de mantenimiento y la portabilidad son algunas de las ventajas de este lenguaje, además que permite descender a nivel de hardware.

La estructura de un programa realizado en c

La estructura de un programa en c es de la siguiente manera:

ZONA DE DECLARACIONES Y CUERPO DEL PROGRAMA.

Zona de declaraciones contiene las librerías y las variables que se van a utilizar en la realización del programa y la función principal además de la declaración de constantes.

Nota: Las variables pueden ir antes de iniciar las instrucciones del programa (Yo lo realizo así).

El cuerpo del programa se divide en inicio, instrucciones y fin.

La estructura quedaría de la siguiente manera:

```
**ZONA DE DECLARACIONES**  
  
#include <stdio.h>  
  
main() *FUNCIÓN PRINCIPAL*  
  
**CUERPO DEL PROGRAMA**  
int ← Tipo de variables nombre; ← Variables  
  
{ ← Inicio del programa  
  
    INTRUCCIONES  
  
} ← Fin del programa
```

Nota: En el lenguaje C no es lo mismo una variable en minúsculas (*ejemplo* nom) a una en mayúsculas (*ejemplo* NOM), por lo que te recomiendo que todo lo hagas en minúsculas.

TIPOS DE VARIABLES

Existen diferentes tipos de variables, aquí menciono las más comunes.

TIPO	MANEJA	CARACTER
int	Enteros	i, d
float	Notación decimal	f
char	Un solo caracter	c
char <i>variable[largo]</i>	Cadena de caracteres	s
int	Entero octal	o
int	Entero Hexadecimal	x

DECLARACIÓN DE LIBRERÍAS Y VARIABLES

Las librerías contienen los parámetros para las instrucciones del lenguaje.

Su declaración es de la siguiente forma:

#include <nombre del archivo> Donde en “<nombre del archivo>” se escribe el nombre de la librería.

Ejemplo:

#include <stdio.h> ← Librería.

Las variables son aquellas cuyo valor puede cambiar durante la ejecución de un programa.

La declaración de variables es de la siguiente forma:

Tipo de variable nombre de la variable

Ejemplo:

int (Tipo de variable Entero) x; (Nombre de la variable)

Nota: Cuando se declara más de una variable se separan por comas y se finaliza con punto y coma (a, b, c;).

OPERADORES ARITMETICOS DE C

símbolo	Significado
+	Suma
-	Resta
*	Multiplicación
/	División

OPERADORES LOGICOS

Símbolo	Descripción
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
= =	Igual
!=	Diferente
&&	And (" Y ")
	Or (" O ")
!	Not (" No ")

ENTRADA Y SALIDA DE DATOS

SALIDA

printf(" DATOS");

Esta instrucción permite la escritura de datos y mensajes en pantalla.

- Todos los mensajes van entre comillas (" ").
- Para escribir el resultado de una variable se escribe `printf("%tipo",variable);`
- Para imprimir en una sola instrucción un mensaje y el resultado :
`printf("mensaje %tipo", variable);`
- Para imprimir un dato y bajar el cursor al sig. Renglón se utiliza " \n " :
`printf("mensaje %tipo \n");`

Ejemplo:

```
printf("El resultado de la suma es: %i \n", resultado);
```

ENTRADA

scanf(“%tipo de variable”,&nombre de la variable); Esta instrucción permite la recibir datos desde el teclado.

Código	Tipo de variable
%d	decimal (con signo)
%i	entero (con signo)
%f	numérico con punto flotante
%o	octal

%u	entero decimal sin signo
%x	hexadecimal
%X	hexadecimal entero largo
%{largo}s	cadena
%c	carácter (solo uno)

Ejemplo:

```
printf("Anote un número \n");
```

```
scanf("%i",&num);
```

PROGRAMAS

1. Programa que imprime en pantalla: HOLA COMO ESTAS.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    printf("HOLA\n");
    printf("COMO ESTAS\n");
    getch();
    return 0;
}
```

Notaras que en el programa anterior esta incluida la librería <conio.h> esta librería es utilizada para getch() que se encuentra al final de las instrucciones; getch(); te obliga a presionar una techa antes de finalizar tu programa.

2. Programa que lee 2 números, los suma, imprime el resultado de la suma y lo multiplica por 2.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int num1, num2, res1, res2;
    printf("Anote el primer numero\n");
```

```

scanf("%i",&num1);
printf("Anoté el segundo numero\n");
scanf("%i",&num2);
res1=num1+num2;
res2=res1*2;
printf("El resultado de la suma es: %i \n y el de la multiplicación es: %i",res1, res2);
getch();
return 0;
}

```

Ejercicios.

3. Elabore un programa que calcule el salario semanal de un trabajador (solo se toma en cuenta el salario diario y los días trabajados).
4. Elabore un programa que lea el nombre y la edad de una persona y que los imprima en pantalla.
5. Elaborar un programa que sume 3 números y el resultado lo divida entre el primero, desde luego el resultado final se imprime en pantalla.

EL CICLO FOR

La forma de la instrucción *for* es:

```

for( inicialización de la variable; condición; incremento)
{
    INSTRUCCIONES
}

```

Ejemplo:

```
for (i=0; i<=10; i++)
```

6. Programa que imprime una serie de números del 1 al 100

```

#include <stdio.h>
#include <conio.h>
int main()
{
    int i;

```

```
for(i=1;i<=100;i++)
    printf(" %i",i);
getch();
return 0;
}
```

7. Programa que lee 5 números e imprime la suma de ellos.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int i, num, r=0;
    for (i=1; i<=5; i++)
    {
        printf("Anote un numero\n");
        scanf("%i",&num);
        r=r+num;
    }
    printf("La suma es: %i",r);
    getch();
    return 0;
}
```

En el programa anterior después del ciclo for se habrá una llave (}) puesto que hay más de una instrucción (en este caso 3) y se finaliza todo lo que realizara el ciclo cerrando con otra llave (}), después que finalice ese ciclo el programa continuara con lo que esta después de } que marca el fin del ciclo.

Ejercicios

8. Desarrolle un programa que lea 10 números e imprima la suma de ellos.
9. Elabore un programa que imprima la serie del 2 al 200.
10. Elabore un programa que imprima el abecedario.

11. Elabore un programa que pueda calcular cualquier tabla de multiplicar.

EL CICLO WHILE

La forma general de la instrucción while es:

While(expresión)

```
{
  Instrucción o instrucciones
}
```

Ejemplo:

12. Programa que suma números indefinidamente hasta que el usuario lo desee.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int a, b, opcion=1;
    while(opcion==1)
    {
        printf("Anoté un numero\n");
        scanf("%i",&a);    b=a*2;
        printf("El resultado es: %i\n",b);
        printf("Desea continuar 1. SI 2. NO\n");
        scanf("%i",&opcion);
    }
    printf("Presione una tecla para salir\n");
    getch();
}
```

13. Programa que lee continuamente un carácter, lo copia y después lo pega en pantalla, el programa termina cuando el carácter sea igual a S.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int opcion;
```

```
clrscr();
opcion = '';
printf("INTRODUSCA UN CARACTER O S PARA SALIR\n");
while (opcion!='S')
{
opcion=getc(stdin);
putchar(opcion);
}
printf("\nBYE");
getch();
}
```

Este es otro ejemplo para empleo del ciclo while. Notaras que ahora al inicio del programa justo después de la declaración de variables hay una instrucción (clrscr();) esta instrucción lo que hace es borrar el contenido de la pantalla anterior es decir del programa que ejecutaste anteriormente. Siempre que se usan caracteres o cadenas de texto se ponen entre apostrofes.

Ejercicios

14. Elabore un programa que sume números y que su ejecución finalice cuando la suma de los números sea mayor a 50.
15. Ahora que conoces la aplicación del clrscr(); utiliza esta instrucción para el problema número 8 de tal forma que las 10 veces que te va a pedir el numero solo aparezca 1 sola vez. Es decir no lo tiene que imprimir las 10 veces como lo hacia antes.

CICLO DO-WHILE

En la instrucción while que acabamos de ver la condición se coloca al principio del ciclo. En el ciclo do-while la expresión se pone al final. De esta forma se garantiza que las instrucciones del ciclo se ejecutan por lo menos una vez antes de pasar por la expresión.

La forma general de la instrucción do-while es:


```
do
{
    Instrucciones;
}
while (expresión);
```

En el ciclo do-while las instrucciones se ejecutan una vez y luego se evalúa la condición para determinar si se continua o no con el ciclo.

Es importante hacer notar que la instrucción do-while termina con un punto y coma.

16. Programa que imprime en pantalla los caracteres de la A hasta la P y sus valores numéricos.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int x=65;
    do
    {
        printf("El valor en decimal de %c es ----> %i \n",x, x);
        x++;
    }
    while (x<=80);
    getch();
    return 0;
}
```

Como se puede observar las instrucciones se ejecutan primero puesto que la instrucción while esta al final; el printf imprime en la pantalla el valor y después se incrementa la variable. Se verifica la condición y como después del incremento el valor de la variable es 66 se sigue cumpliendo la condición y el programa vuelve a iniciarse desde el do, y así sucesivamente hasta que la condición sea mayor a 80.

Nota: El incremento que vemos como $x++$; es igual que $x = x + 1$;

CICLOS ANIDADOS

Un ciclo anidado no es más que un ciclo (ciclo interno) dentro de otro ciclo (ciclo externo). Cuando un programa llegue al ciclo interno será ejecutado como cualquier otra instrucción pero dentro del ciclo externo.

Ejemplo:

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int i, y;
    for(i=1; i<=3; i++)
    {
        printf("Inicio del ciclo externo cuyo valor de incremento es: %i\n",i);
        for(y=1; y<=3; y++)
            printf("    Inicio del ciclo interno cuyo valor de incremento es: %i\n",y);
    }
    getch();
    return 0;
}
```

En el programa anterior se encuentran anidados 2 ciclos for:

17. Elabora un programa que simule un reloj que inicia en 0 horas 0 minutos 0 segundos y termine en 23 horas 59 minutos y 59 segundos.

**** LOS CICLOS ANIDADOS NO SOLO SE USAN CON FOR TAMBIEN ****

**PUEDES UTILIZARLOS CON WHILE, DO-WHILE Y CON LA PROPOSICION IF-ELSE
(QUE VEREMOS A CONTINUACIÓN)**

IF-ELSE

La proposición if-else es utilizada para expresar toma de decisiones.

Forma de la instrucción if-else:

```
if(expresión)
{
```

```
        Instrucción1;
    }
else
    {
        instrucción2;
    }
```

En el caso de if-else, al igual que en el ciclo for, se utilizan llaves ({ }) para marcar el inicio o el final solo en caso de ser más de una instrucción.

Ejemplo:

18. Programa que lee dos números y dice cual es el mayor y cual es el menor.

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

```
#include <conio.h>
```

```
int main()
```

```
{
```

```
    int num1, num2;
```

```
    printf("Anote un numero\n");
```

```
    scanf("%i",&num1);
```

```
    printf("Anote otro numero\n");
```

```
    scanf("%i",&num2);
```

```
    if (num1>num2)
```

```
        printf("El numero %i es mayor que %i\n",num1, num2);
```

```
    else
```

```
        printf("El numero %i es menor que %i\n",num1, num2);
```

```
    getch();
```

```
    return 0;
```

```
}
```

19. Programa que lee un número e imprime si es positivo o negativo.

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

```
#include <conio.h>
```

```
int main()
```

```
{
```

```
int num;
clrscr();
printf("ANOTE UN NUMERO\n");
scanf("%i",&num);
if (num>=0)
    printf("EL NUMERO %i ES POSITIVO",num);
else
    printf("EL NUMERO %i ES NEGATIVO",num);
getch();
return 0;
}
```

ELSE-IF

Básicamente el else-if se utiliza para escribir una decisión múltiple y su forma general es de la siguiente manera:

```
if(condición)
    {
        Instrucciones;
    }
else if (condición)
    }
    Instrucciones;
else
    {
        Instrucciones;
    }
```

Es importante remarcar que no importa la cantidad de else-if que pongas no existe un límite. El último else se maneja en caso de que ninguno de los anteriores cumpla con la condición.

20. Programa que muestra en pantalla una serie de opciones a elegir e imprime el costo del producto elegido.

Ejemplo:

```
#include <stdio.h>
#include <conio.h>
int main()
```

```
{
    int p;
    clrscr();
    printf("SELECCIONE UN PRODUCTO\n\n");
    printf("1. REFRESCO\n");
    printf("2. PAPAS FRITAS\n");
    printf("3. HAMBURGUESA\n");
    printf("4. JUGO\n");
    scanf("%i",&p);
    if (p == 1)
        printf("EL COSTO ES: $5.00");
    else if(p == 2)
        printf("EL COSTO ES: $10.00");
    else if(p == 3)
        printf("EL COSTO ES: $20.00");
    else
        printf("EL COSTO ES: $8.00");
    getch();
    return 0;
}
```

Ejercicios:

21. Elabore un programa que lea un carácter e imprima si es vocal, consonante o numérico.

22. Elabore un programa que imprima si una persona es Niño, Adolescente, Joven o Mayor. Para ello considere que Niño será hasta la edad de 14 años Adolescente de 15 a los 20, joven de los 21 hasta los 40 y mayor en adelante.

23. Elabore el siguiente programa:

Una persona desea saber cuanto dinero se generara por concepto de intereses para una cantidad que va a invertir en un banco. La persona invertirá siempre y cuando la cantidad generada por intereses excedan los \$7,000.00 y en este caso desea saber cuando dinero tendrá al final.

24. Una empresa de bienes raíces ofrece casas bajo las siguientes condiciones:

a) En caso de que los ingresos del comprador sean menores de \$8,000.00 el enganche será del 15% del costo total de la casa y el resto se distribuirá en pagos mensuales a pagar en 15 años.

b) si los ingresos del comprador son otros el enganche será del 25% del costo total de la casa y el resto se distribuirá en pagos mensuales a pagar en 8 años.

c) La empresa desea obtener la cantidad a pagar por concepto de enganche y cuanto por concepto de pagos mensuales.

SWITCH Y BREAK

La instrucción switch es usada para tomar decisiones u opciones ilimitadas con base en un valor de una expresión condicional.

La forma general de la instrucción switch es:

```
Switch (condición)
{
    case constante1:
        Instrucciones;
        break;
    case constante2:
        Instrucciones;
        break;
    default:
        Instrucciones;
        break;
```

La instrucción break se agrega al final de cada etiqueta case y provoca la salida del switch.

Ejemplo:

25. Programa que pide un número del 1 al 7 y te dice que día de la semana es.

```
#include <stdio.h>
#include <conio.h>
int main()
{
```

```
int opcion;
clrscr();
printf("INTRODUSCA UN NUMERO ENTERO DEL 1 AL 7\n");
scanf("%i",&opcion);
switch(opcion)
{
case 1:
printf("EL DIA %i ES DOMINGO",opcion);
break;
case 2:
printf("EL DIA %i ES LUNES",opcion);
break;
case 3:
printf("EL DIA %i ES MARTES", opcion);
break;
case 4:
printf("EL DIA %i ES MIERCOLES",opcion);
break;
case 5:
printf("EL DIA %i ES JUEVES", opcion);
break;
case 6:
printf("EL DIA %i ES VIERNES", opcion);
break;
case 7:
printf("EL DIA %i ES SABADO", opcion);
break;
default:
printf("EL NUMERO NO PERTENECE AL RANGO");
break;
}
getch();
return 0;
}
```

26. Programa que muestra en pantalla un menú con opciones y termina el programa cuando el usuario seleccione salir.

```
#include <sodio.h>
#include <conio.h>
int main()
{
int opcion, i, a, b, c;
```

```

clrscr();
do
{
clrscr();
printf("SELECCIONE UNA OPCION\n");
printf("1. SUMA 2 NUMEROS\n");
printf("2. MUESTRA UNA SERIE DEL 1 AL 50\n");
printf("3. SALIR\n");
scanf("%i",&opcion);
switch(opcion)
{
case 1:
printf("ANOTE UN NUMERO\n");
scanf("%i %i",&a, &b);
c=a+b;
printf("LA SUMA ES: %i\n",c);
printf("\n\nPRESIONE UNA TECLA PARA VOLVER AL MENU
PRINCIPAL");
getch();
break;
case 2:
for(i=1; i<=50; i++)
printf(" %i",i);
printf("\n\nPRESIONE UNA TECLA PARA VOLVER AL MENU
PRINCIPAL\n");
getch();
break;
}
}
while (opcion!=3);
}

```

Este programa es un claro ejemplo de cómo aplicar el switch para hacer un menú con diferentes opciones y el while para continuar dentro del programa.

Ejercicios:

27. Un restaurante esta ofreciendo los siguientes combos:
1. HAMBURGUESA, PAPAS GRANDES Y REFRESCO x \$50.00.
 2. HAMBURGUESA CON QUESO Y DOBLE CARNE PAPAS G. Y REFRESCO \$65.00.
 3. ORDEN DE TACOS Y REFRESCO \$32.00.

Se desea obtener el final del día la cantidad de clientes que compraron cada uno de los diferentes combos, además del total de la venta diaria de cada uno de los combos y la venta total del día.

28. Elabore un programa que muestre en pantalla un menú con las siguientes opciones:

1. SUMA DE 2 NUMEROS.
2. CALCULAR CUALQUIER TABLA DE MULTIPLICAR.
3. IMPRIMIR EL ABECEDARIO.
4. SALIR.

Desde luego el programa debe de ejecutar cada una de estas opciones.

“ESTE ES EL FINAL DE ESTE PEQUEÑO CURSO PERO SEGUIRE AGREGANDO
MAS INFORMACIÓN”

“Dedicado mi trabajo a Yolanda Marisela”