



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Ing. Karina García Morales

*Asignatura:* Fundamentos de programación

*Grupo:* 20

*No. de práctica(s):* Práctica 7

*Integrante(s):* Francisco Javier Gómez Mendoza

*No. de lista o brigada:* 22

*Semestre:* 2023-1

*Fecha de entrega:* 20 de noviembre del 2022

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

# Estructuras de selección

## Objetivo:

El alumno elaborará programas en lenguaje C que incluyan las estructuras de selección if, if-else, switch y ternaria (o condicional) para la resolución de problemas básicos.

## Introducción

### Estructura de control selectiva if

La estructura de control de flujo mas simple es el **if** la sintaxis es esta

```
if (expresión_lógica)
{
    // bloque de código a ejecutar
}
```

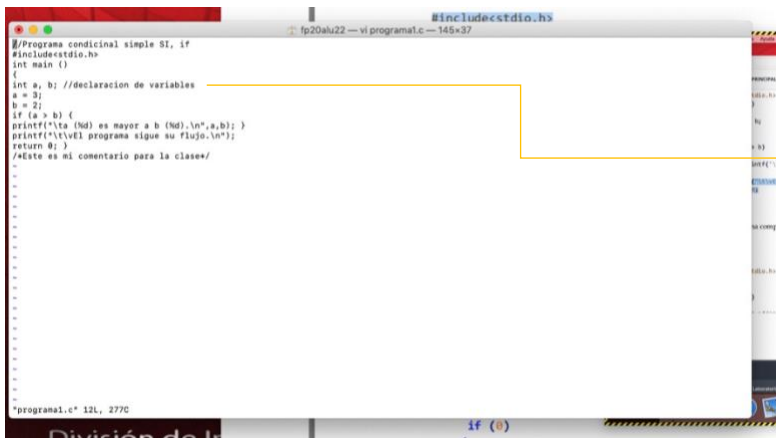
Esta estructura evalúa la expresión lógica, y de ser cumplida ósea verdadera, se ejecutarán las instrucciones de lo que se encuentra entre llaves

Es de conociendo si las instrucciones del código solo consta de una línea no es necesario el uso de llaves

## Trabajo en clase

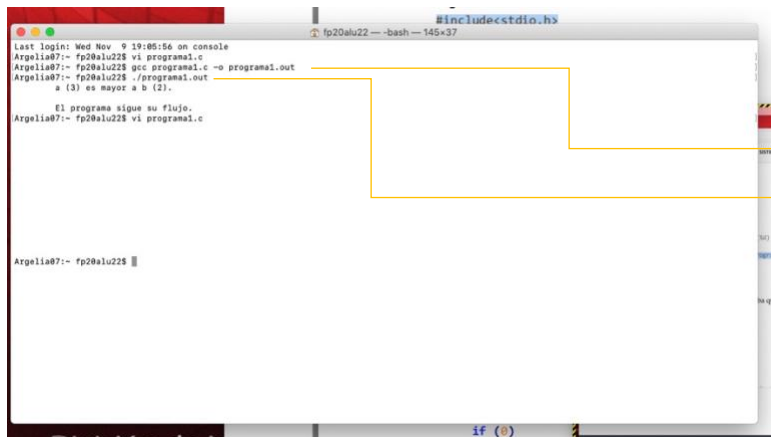
### Códigos (estructura de control selectiva if)

El **programa1.c** valida si el valor de la variable **a** es mayor al valor de la variable **b**



```
#include <stdio.h>
int main ()
{
    int a, b; //declaracion de variables
    a = 3;
    b = 2;
    if (a > b) {
        printf("\ta es mayor a b (%d,%d)\n",a,b);
        printf("\tEl programa sigue su flujo\n");
        return 0;
    }
    /*Este es mi comentario para la clase*/
}
```

Aquí solo copiamos el código y agregamos comentarios respecto a que hacia el programa usando doble vez slash //



```
Argelia07:~$ gcc programa1.c -o programa1.out
Argelia07:~$ ./programa1.out
a es mayor a b (3,2)
El programa sigue su flujo.
```

Aquí solo compilamos con el uso de gcc. Después solo ejecutamos con ./

En el **programa2.c** comprobamos que las condiciones son numéricas 0 → falso, ≠ 0 → verdadero.

```
#include<stdio.h>
int main()
{
    if (0)
    {
        printf("Esta instrucción nunca se ejecuta\n");
        printf("porque la condición siempre es falsa (0).\n"); }
    if (-30) // El bloque de código de esta estructura if
    // solo consta de una línea porque los comentarios // no son tomados en cuenta por el compilador.
    // La condición siempre es verdadera (diferente de 0)
    printf("Esta instrucción siempre se ejecuta.\n"); return 0;
}
```

Solo codificamos y agregamos comentarios relacionados con las acciones que realiza nuestro programa, también hicimos uso de el condicional simple if

Después lo que hicimos es compilar para detectar algún error en nuestra deificación, y al final compilamos y nos señalo el programa Esta instrucción siempre se ejecuta.

```
Argelia07:~ fp20alu22$ vi programa2.c
Argelia07:~ fp20alu22$ gcc programa2.c -o programa2.out
Argelia07:~ fp20alu22$ ./programa2.out
Esta instrucción siempre se ejecuta.
Argelia07:~ fp20alu22$
```

## Actividad en clase

En esta actividad en el laboratorio creamos un código que le pidiera al usuario su edad y el programa imprimiera si era menor de edad o mayor de edad

```
#include <stdio.h>
int edad;
int main() {
    printf("ingresa tu edad");
    scanf("%d",& edad);
    if(edad<=10){
        printf("eres mayor de edad");
    }
    else
        printf("eres menor de edad");
    return 0;
}
```

En esta codificación también hicimos uso de **if**, la cual nos ayudo a definir un rango para determinar si era mayor o menor, hicimos uso de **scanf** para que el usuario nos brinde información y poder determinar.

```
Argelia07:~ fp20alu22$ vi Ejercicio1.c
Argelia07:~ fp20alu22$ gcc Ejercicio1.c -o Ejercicio1.out
Argelia07:~ fp20alu22$ ./Ejercicio1.out
ingresa tu edad 12
eres menor de edadArgelia07:~ fp20alu22$ gcc Ejercicio1.c -o Ejercicio1.out
Argelia07:~ fp20alu22$ ./Ejercicio1.out
ingresa tu edad 21
eres mayor de edadArgelia07:~ fp20alu22$
```

En esta etapa solo compilamos, para asi saber si teníamos algún error, y ejecutamos y agregamos valores y supervisar que funcione nuestro programa

## Estructura de control selectiva if-else

La sintaxis de la estructura de control de flujo if-else es la siguiente:

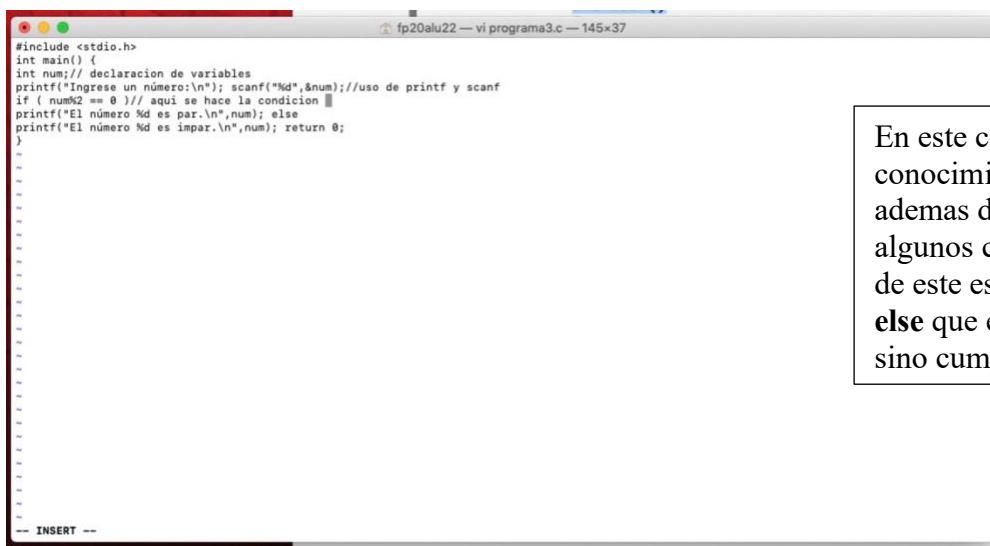
```
if (expresión_lógica)
{
    // bloque de código a ejecutar
    // si la condición es verdadera
}
else
{
    // bloque de código a ejecutar
    // si la condición es falsa
}
```

Esta estructura evalúa la expresión lógica y si la condición es verdadera se ejecutan las instrucciones del bloque que se encuentran entre las primeras llaves, si la condición no cumple se ejecuta el else y hace otro tipo de instrucciones.

## Trabajo en clase

### Código (estructura de control selectiva if-else)

Este programa permite validar si un número es par o impar. El número se lee desde la entrada estándar.



```
#include <stdio.h>
int main() {
    int num; // declaracion de variables
    printf("Ingrese un número:\n"); scanf("%d",&num); //uso de printf y scanf
    if (num%2 == 0) // aqui se hace la condicion
        printf("El número %d es par.\n",num); else
        printf("El número %d es impar.\n",num); return 0;
}
```

En este código usamos todos lo conocimientos antes planteados, ademas de que también agregamos algunos comentarios, pero lo especial de este es que hicimos el uso de el **if else** que es el si no, la condición del sino cumple hacer esto

```
[Argelia07:~ fp20alu22$ vi programa3.c
[Argelia07:~ fp20alu22$ gcc programa3.c -o programa3.out
[Argelia07:~ fp20alu22$ ./programa3.out
Ingrese un número:
14
El número 14 es par.
[Argelia07:~ fp20alu22$
```

Después solo lo compilamos para asegurarnos que no tuviera errores y poder ejecutarlo, cuando lo ejecutamos solo nos pidió un numero y nos tenia que decir si era par o impar

## Trabajo en clase

### Código (estructura de control selectiva if-else anidada)

Este programa ordena en forma descendente tres valores enteros dados. Los valores se leen desde la entrada estándar

En el programa4.c usamos la condición de **if else**, pero para este código tenía que hacer el proceso 3 veces, por que tiene que revisar cada numero que el usuario ingrese

```
#include <stdio.h>
int main() {
    int uno, dos, tres; // aqui declaramos variables
    printf ("Ingrese 3 números separados por espacios:\n"); scanf ("%d %d %d", &uno, &dos, &tres);
    if (uno > dos) // aqui hacemos uso de una condicion
    {
        if (dos > tres) {
            printf("%d es mayor a %d que es mayor a %d\n", uno, dos, tres);
        }
        else
        {
            if (uno > tres)
            {
                printf("%d es mayor a %d que es mayor a %d\n", uno, tres, dos);
            }
            else
            {
                printf("%d es mayor a %d que es mayor a %d\n", tres, uno, dos);
            }
        }
    }
    else
    {
        if (dos > tres)
        {
            if (tres > uno)
            {
                printf("%d es mayor a %d que es mayor a %d\n", dos, tres, uno); //usamos el printf para imprimir lo que deseamos que vea el usuario
            }
            else
            {
                printf("%d es mayor a %d que es mayor a %d\n", dos, uno, tres);
            }
        }
        else {
            printf("%d es mayor a %d que es mayor a %d\n", tres, dos, uno);
        }
    }
    return 0;
}
-- INSERT --
```

Para la compilación no tuvimos problema así que ejecutamos y ingresamos 3 numero distintos y el programa los ordeno de mayor a menor

```
Argelia07:~ fp20alu22$ vi programa4.c
Argelia07:~ fp20alu22$ gcc programa4.c -o programa4.out
Argelia07:~ fp20alu22$ ./programa4.out
Ingrese 3 números separados por espacios:
2 14 8
14 es mayor a 8 que es mayor a 2
Argelia07:~ fp20alu22$
```

## Estructura de control selectiva switch-case

La sintaxis de la estructura switch-case es la siguiente:

```
switch (opcion_a_evaluar)
{
    case valor1:
        /* Código a ejecutar*/
        break;
    case valor2:
        /* Código a ejecutar*/
        break;
    case valorN:
        /* Código a ejecutar*/
        break;
    default:
        /* Código a ejecutar*/
}
```

Esta estructura es la de condicional múltiple, al momento de confiscar a cada opción le llamaremos caso y entre cada caso debe haber alguna acción por ejecutar y al final de cada caso escribir break, este nos funcionará de manera que sólo realiza solo esa vez el proceso y no lo realiza varias veces, y incluir en el programa el default, el cual hará la acción de terminar el proceso si el usuario no ingresa alguna de las opciones proporcionadas.

## Trabajo en clase

### Código (estructura de control selectiva switch-case)

Este programa permite elegir una opción del menú a partir del carácter ingresado. La opción se lee desde la entrada estándar

```
#include <stdio.h>
int main()
{
    char op = '\0';
    printf("\tcomida Mexicana\n\n");
    printf("Elegir la opción deseada\n"); printf("a) Mole\n");
    printf("b) Pozole\n"); printf("c) Chilaquiles\n"); scanf("%c",&op);
    switch(op)
    {
        default:
            printf("Opción no valida.\n"); break;
        case 'a':
            printf("Se seleccionó 'Mole'.\n"); break;
        case 'b':
            printf("Se seleccionó 'Pozole'.\n"); break;
        case 'c':
            printf("Se seleccionó 'Chilaquiles'.\n"); break;
    }
    return 0;
}
```

En el programa5.c. Usamos el switch y en mi caso hice un menú de comida mexicana con 3 ricas opciones de platillos, usando la codificación correspondiente pude poner 1. Mole, 2.Pozole, 3.Chilaquiles .

```
Argelia07:~ fp20alu22$ gcc Program5.c -o Program5.out
Argelia07:~ fp20alu22$ ./Program5.out
comida Mexicana

Elegir la opción deseada
a) Mole
b) Pozole
c) Chilaquiles
c
Se seleccionó 'Chilaquiles'.
Argelia07:~ fp20alu22$
```

Al compilar y ejecutar le pedia al usuario ingresar de forma de carácter la opción que quería seleccionar, mi código no tubo ningún problema así i que ejecuto a la perfección

## Enumeración

Existe otro tipo de dato constante conocido como enumeración. Una variable enumerador se puede crear de la siguiente manera:

enum identificador {VALOR1, VALOR2, ... , VALORN};

Para hacer una enumeración se usa enumera que es una palabra recerbaday después entre llaves los valores que puede tomar esa enumeración

La enumeración se llama 'boolean' y contiene dos elementos, el primero (FALSE) posee el valor 0 y el siguiente (TRUE) posee el valor 1. Si hubiese más elementos en la enumeración, la numeración correría de manera ascendente.

## Trabajo en clase

### Códigos (variables tipo enumeración)

Este programa permite elegir una opción del menú a partir del entero ingresado. La opción se lee desde la entrada estándar

Como bien sabemos la enumeración pueden ser palabras, así si que en el ejercicio de clase 8, pusimos en lista los días de la semana.

Solo podíamos elegir una vez, por que el switch tiene breaks, que hacen que no sea un ciclo las opciones

```
#include <stdio.h>
int main() {
    // Los valores de una enumeración son enteros y constantes
    enum diasSemana {LUNES, MARTES, MIERCOLES, JUEVES, VIERNES, SABADO, DOMINGO}; int op;
    printf("Ingrese el día de la semana.\n");
    printf("1) Lunes\n");
    printf("2) Martes\n"); printf("3) Miércoles\n"); printf("4) Jueves\n");
    printf("5) Viernes\n"); printf("6) Sábado\n"); printf("7) Domingo\n"); scanf("%d", &op);
    switch(op-1) { //aquí usamos la condición multiple donde son mas de dos opciones.
        case LUNES: case MARTES:
            printf("Inicio de semana.\n");
            break; case MIERCOLES:
            printf("Mitad de semana.\n");
            break; case JUEVES:
            printf("¡Casi inicia el fin de semana!\n");
            break; case VIERNES:
            printf("¡Fin de semana!\n"); break;
        case SABADO:
            printf("Fin de semana!\n"); break;
        case DOMINGO:
            printf("Día de descanso.\n"); break;
        // No se necesita default
    }
    return 0; // Valor entero en hexadecimal }
~
~
~
~
~
~
~
~
~
~
```

```
[Argentina08:~ fp20alu45$ vi programa8.c
[Argentina08:~ fp20alu45$ gcc programa8.c -o programa8.out
[Argentina08:~ fp20alu45$ ./programa8.out
Ingrese el día de la semana.
1) Lunes
2) Martes
3) Miércoles
4) Jueves
5) Viernes
6) Sábado
7) Domingo
6
¡Fin de semana!
Argentina08:~ fp20alu45$ █
```

A momento de ejecutar se eligió el numero 6, y termino el programa.

## Estructura de control selectiva condicional

Permiten hacer una comparación rápida. Su sintaxis es la siguiente

Condición ? SiSeCumple : SiNoSeCumple

Es de tres partes, una condición y dos acciones a seguir con base en la expresión condicional. Si es verdad la condición se ejecuta la instrucción que se encuentra después del símbolo del signo de interrogación, si es falsa se ejecuta la que está después de la que está después.

## Código (Estructura de control selectiva condicional o ternaria).

Este programa permite calcular el error matemático a partir de dos valores (a y b) ingresados desde la entrada estándar (el teclado), a partir de la fórmula:  $E = |a - b|$ . Donde a es el valor real y b es el valor aproximado o viceversa.

```
#include <stdio.h>
int main()
{
    double a, b, res;
    printf("Calcular el error matemático E = |a - b|\n\n"); printf("Ingrese el valor de a:\n");
    scanf("%lf",&a);
    printf("Ingrese el valor de b:\n");
    scanf("%lf",&b);
    res = a < b ? b-a : a-b;
    printf("El error matemático de\n");
    printf("| %lf - %lf | es %lf\n", a, b, res);
    return 0;
}
```

En este ejercicio no se uso el condicional if-else, si no que sustituyo por la condición ternaria por un signo de interrogación

Ademas usamos una variable de tipo double

```
Argelia07:~ fp20alu22$ vi programa9.c
Argelia07:~ fp20alu22$ gcc programa9.c -o programa9.out
Argelia07:~ fp20alu22$ ./programa9.out
Calcular el error matemático E = |a - b|

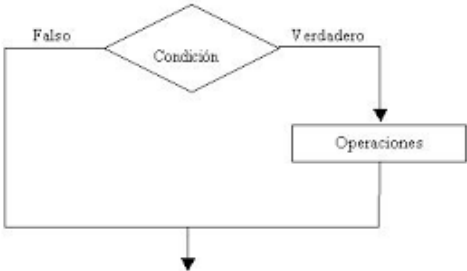
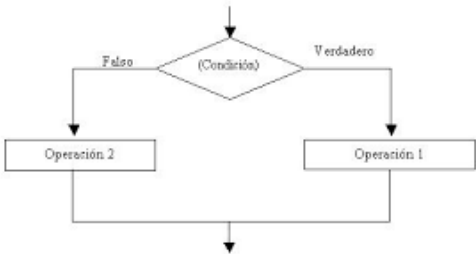
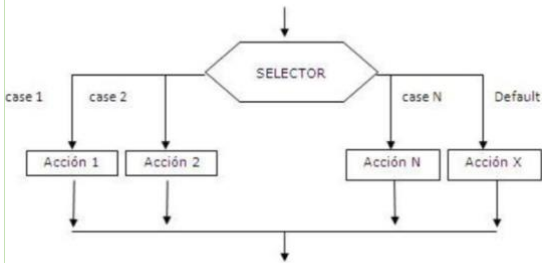
Ingrese el valor de a:
14
Ingrese el valor de b:
2
El error matemático de
| 14.000000 - 2.000000 | es 12.000000
Argelia07:~ fp20alu22$
```

En la compilación y ejecución no tuvimos problemas, solo se realizo una resta



## TAREA

### 1.- Completar la tabla con diagrama, pseudocódigo y codificación para la estructura condicional

DIAGRAMA	PSEUDOCÓDIGO	CODIFICACIÓN
<b>SIMPLE</b> 	SI<expresión lógica>ENTONCES Instrucciones FIN SI	If (expresión) { Sentencia; } 
<b>COMPUESTO</b> 	SI<expresión lógica>ENTONCES Instrucciones DE LO CONTRARIO Instrucciones FIN SI	If(expresión) { Sentencia; } Else { Sentencia; } 
<b>MÚLTIPLE</b> <p>DIAGRAMA DE FLUJO</p> 	SELECCIONAR (variable)EN CASO VAR1 -> Instrucciones CASO VAR2 -> Instrucciones CASO VAR3 -> Instrucciones CASO VAR4 -> Instrucciones FIN SELECCIONAR	Switch(opcion) { Case valor1: Sentencia Break; Default; Sentencia } 

## 2.- Realiza un programa que permita seleccionar entre 3 programas

a) Programa que ejecute una condición que siempre sea verdadera e imprima "Fundamentos de programación"(Utiliza valores booleanos)

b) Programa que ejecute una condición compuesta y que siempre sea falsa e imprima en el else "Error"(Utiliza valores booleanos)

c) Programa del condicional anidado con condicional ternario y comentarios en el código fuente (//Comentar las líneas modificadas)

```

/*****
Online C++ Compiler.
Code, Compile, Run and Debug C++ program online.
Write your code in this editor and press "Run" button to compile and execute it.
*****/

#include <iostream>

using namespace std;

int main()
{
    char op = '\0';
    double a, b, divicion;
    printf("\tMenÃº\n\n");
    printf("Elegir la opciÃ³n deseada\n");
    printf("a) Imprimir Fundamentod De Programacion\n");
    printf("b) imprimir error\n");
    printf("c) condicional ternario\n");
    scanf("%c",&op);
    switch(op)
    {
        default:
            printf("OpciÃ³n no vÃ¡lida.\n"); break;
        case 'a':
            printf("Se seleccionÃ³ 'Fundamentos de Programacion'.\n"); break;
        case 'b':
            if(0){
                printf("its this 0.\n");
            }
            else
            {
                printf("\n ERROR \n");
            }
            break;
        case 'c': //para el caso 3 me guie del num. 9 de la practica
            printf("dividir dos numeros E = |a / b|\n\n"); //use exactamente la misma formula y variables que la practica
                                                                    //solo que en ves de resta es divivon
            printf("Ingrese el valor de a:\n");
            scanf("%lf",&a);
            printf("Ingrese el valor de b:\n");
            scanf("%lf",&b);
            divicion = a < b ? b/a : a/b; // este es lo que diferencia del if-else, con el signo ?
                                        // este signo hace la funcion de diferente o si no
                                        //lo que hace esta accion antes del signo es si es verdadero, si
                                        // es falso ara lo que le continua.
            printf("la divicion de los dos numeros es: \n");
            printf("| %lf / %lf | es %lf\n", a, b, divicion);
            break;
    }

    return 0;
}
```

**Nota:** el compilador que yo uso me da la opción de mostrarme el código, trabajo desde un iPad y online se me es complicado mostrar todo el código desde el editor.

```
8
9 #include <iostream>
10
11 using namespace std;
12
13 int main()
14 {
15     char op = '\0';
16     double a, b, division;
17     printf("\tMenú\n\n");
18     printf("Elegir la opción deseada\n");
19     printf("a) Imprimir Fundamentod De Programacion\n");
20     printf("b) imprimir error\n");
21     printf("c) condicional ternario\n");
22     scanf("%c",&op);
23     switch(op)
24     {
25     default:
26         printf("Opción no válida.\n"); break;
27     case 'a':
28         printf("Se seleccionó 'Fundamentos de Programacion'.\n"); break;
29     case 'b':
30         if(0){
31             printf("its this 0.\n");
32         }
33         else
34         {
35             printf("\n ERROR \n");
36         }
37         break;
38     case 'c': //para el caso 3 me guie del num. 9 de la practica
39         printf("dividir dos numeros E = |a / b|\n\n"); //use exactamente la misma
40                                                         //solo que en ves de resta
41         printf("Ingrese el valor de a:\n");
42         scanf("%lf",&a);
43         printf("Ingrese el valor de b:\n");
44         scanf("%lf",&b);
45         division = a < b ? b/a : a/b; // este es lo que diferencia del if-else, c
46                                     // este signo hace la funcion de diferente
47                                     //lo que hace esta accion antes del signo e
48                                     // es falso ara lo que le continua.
49         printf("la division de los dos numeros es: \n");
50         printf("| %lf / %lf | es %lf\n", a, b, division);
51         break;
52     }
53     return 0;
54 }
55
56
```

```
Menú
Elegir la opción deseada
a) Imprimir Fundamentod De Programacion
b) imprimir error
c) condicional ternario
a
Se seleccionó 'Fundamentos de Programacion'

...Program finished with exit code 0
Press ENTER to exit console.
```

Ejecución ion de a)

```
Menú
Elegir la opción deseada
a) Imprimir Fundamentod De Programacion
b) imprimir error
c) condicional ternario
b

ERROR

...Program finished with exit code 0
Press ENTER to exit console.
```

Ejecución de b)

```
Elegir la opción deseada
a) Imprimir Fundamentod De Programacion
b) imprimir error
c) condicional ternario
c
dividir dos numeros E = |a / b|

Ingrese el valor de a:
30
Ingrese el valor de b:
10
la division de los dos numeros es:
| 30.000000 / 10.000000 | es 3.000000

...Program finished with exit code 0
Press ENTER to exit console.
```

Ejecución de c)

3.- Del programa del condicional anidado, sustituir los if-else por el condicional ternario (?), según corresponda y agregar la condición para valores iguales.

```
2
3
4      Online C++ Compiler.
5      Code, Compile, Run and Debug C++ program online.
6      Write your code in this editor and press "Run" button to compile and execute it.
7      *****/
8
9      #include <iostream>
10
11     using namespace std;
12
13     int main()
14     {
15         int uno, dos, tres;
16         printf("ingrese tres numeros separador por espacios:\n");
17         scanf("%d %d %d", &uno, &dos, &tres);
18         uno > dos & dos > tres ? printf("%d es mayor a %d que es mayor a %d\n", uno, dos, tres) : uno > t
19
20         return 0;
21     }
22
```

Input

ingrese tres numeros separador por espacios:  
30 40 4  
40 es mayor a 30 que es mayor a 4

...Program finished with exit code 0  
Press ENTER to exit console.

```
/***/

      Online C++ Compiler.
      Code, Compile, Run and Debug C++ program online.
      Write your code in this editor and press "Run" button to compile and execute it.

      *****/

#include <iostream>

using namespace std;

int main()
{
    int uno, dos, tres;
    printf("ingrese tres numeros separador por espacios:\n");
    scanf("%d %d %d", &uno, &dos, &tres);
    uno > dos ? dos > tres ? printf("%d es mayor a %d que es mayor a %d\n", uno, dos, tres) : uno >
tres ? printf("%d es mayor a %d que es mayor a %d\n", uno, tres, dos) : printf("%d es mayor a %d que es
mayor a %d\n", tres, uno, dos) : dos > tres ? tres > uno ? printf("%d es mayor a %d que es mayor a
%d\n", dos, tres, uno) : printf("%d es mayor a %d que es mayor a %d\n", dos, uno, tres) : printf("%d
es mayor a %d que es mayor a %d\n", tres, dos, uno);

    return 0;
}
```

4.- Del programa del condicional ternario(?), sustituir por if-else, según corresponda y agregar la condición para valores iguales.

```
3 Welcome to GDB Online.
4 GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,
5 C#, OCaml, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.
6 Code, Compile, Run and Debug online from anywhere in world.
7
8 *****/
9 #include <iostream>
10
11 using namespace std;
12
13 int main()
14 {
15
16     int a, b, res;
17     printf("Calcular el error matemático de a-b a mayor que b \n\n");
18     printf("Ingrese el valor de a:\n");
19     scanf("%d",&a);
20     printf("Ingrese el valor de b:\n");
21     scanf("%d",&b);
22     if (a>b)
23     {
24         printf("su error matematico es: %d",res = a-b);
25     }
26     else
27     {
28         printf("su error matematico es %d\n", res= b-a);
29     }
30     printf("\n\nEs el error matemático de a-b o b-a\n");
31     return 0;
32 }
```

## Conclusiones

- Algo que me gustaría reconocer es que ya se me facilita hacer muchas cosas dentro del código.
- Aun tengo algunos problemas para hacer unas cosas, pero me gusta ya empesar hacer la mayoría de cosas si ayuda de nadie.
- Tengo que aceptar que si me tuvieron que apoyar en algún ejercicio por que me atorre mucho como el ejercicio 3 de tarea, aunque así entendí mas el proceso.
- La practica me gusto por que ya estamos codiciado, y me gusta cuando al primer intento pueda resolver el programa, aunque hay cosas que dentro de las etapas de programación aun tenga que mejor me gusta ya entender más cosas.
- La dificultad siempre cada práctica, pero me gusta aprender y mejorar en cosas que me cuesten, pero hasta el momento es la que más trabajo eh tenido que hacer

## Referencias

- Online C++ Compiler - online editor. (s. f.). GDB online Debugger.  
[https://www.onlinegdb.com/online\\_c++\\_compiler](https://www.onlinegdb.com/online_c++_compiler)
- Laboratorios Salas A y B (unam.mx)