	<b>Carátula para entrega de prácticas</b>	
Facultad de Ingeniería		Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Karina García Morales

*Asignatura:* Fundamentos de programación

*Grupo:* 20

*No. de práctica(s):* 09

*Integrante(s):* Suzán Herrera Álvaro

*No. de lista o brigada:* 49

*Semestre:* 1

*Fecha de entrega:* 22 noviembre 2022

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

**Objetivo:** El alumno elaborará programas en C para la resolución de problemas básicos que incluyan las estructuras de repetición.

### Conceptos:

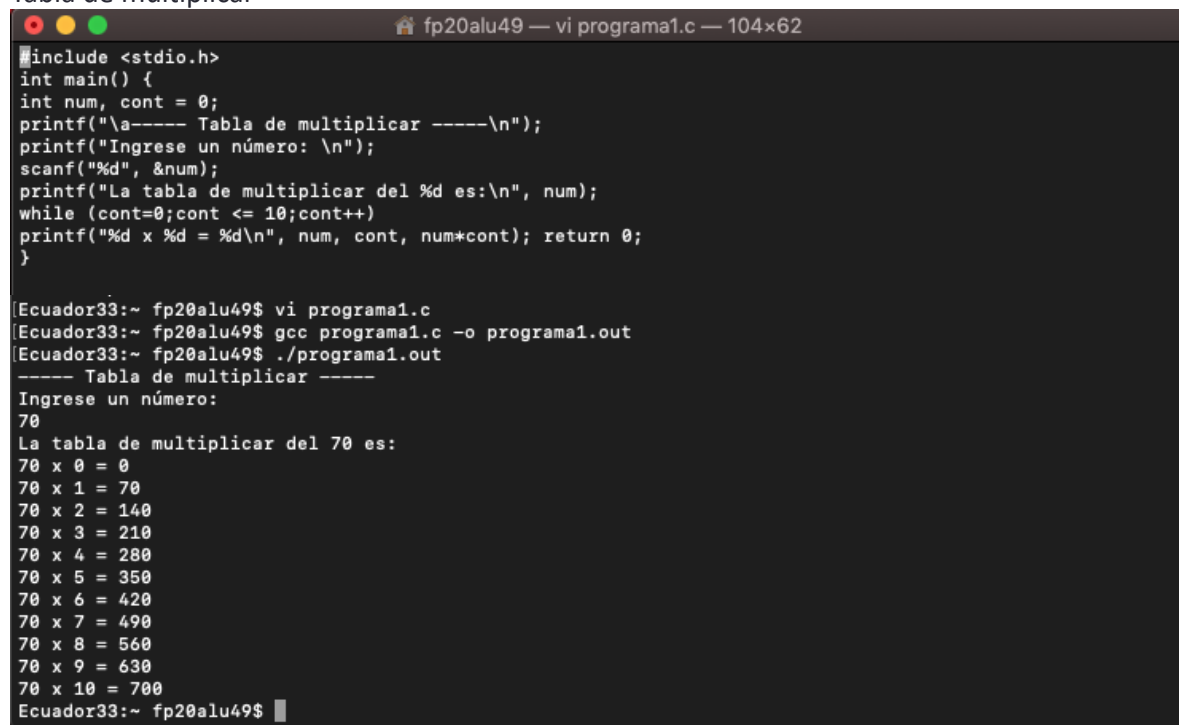
**Ciclo mientras**, repite una acción o cadena de acciones hasta que cierto parámetro sea verdadero

**Ciclo hacer-mientras**, realiza una acción o cadena de acciones, luego, comprueba que un parámetro sea verdadero, si no lo es, repite la acción o cadena de acciones.

**Ciclo para**, ejecuta un comando con una cantidad de pasos especificada, cuando se cubren los pasos termina la estructura.

Ejercicios:

Tabla de multiplicar



```
fp20alu49 — vi programa1.c — 104x62
#include <stdio.h>
int main() {
    int num, cont = 0;
    printf("\a----- Tabla de multiplicar -----\\n");
    printf("Ingrese un número: \\n");
    scanf("%d", &num);
    printf("La tabla de multiplicar del %d es:\\n", num);
    while (cont=0;cont <= 10;cont++)
    printf("%d x %d = %d\\n", num, cont, num*cont); return 0;
}

Ecuador33:~ fp20alu49$ vi programa1.c
Ecuador33:~ fp20alu49$ gcc programa1.c -o programa1.out
Ecuador33:~ fp20alu49$ ./programa1.out
----- Tabla de multiplicar -----
Ingrese un número:
70
La tabla de multiplicar del 70 es:
70 x 0 = 0
70 x 1 = 70
70 x 2 = 140
70 x 3 = 210
70 x 4 = 280
70 x 5 = 350
70 x 6 = 420
70 x 7 = 490
70 x 8 = 560
70 x 9 = 630
70 x 10 = 700
Ecuador33:~ fp20alu49$
```

Usando la estructura while se genera un ciclo que puede ser infinito o que puede ser de única ejecución; esto se logra forzando al while con un valor siempre verdadero o un valor siempre falso (0 – 1)

Usando las estructuras do-while se obtienen un indeterminado número de calificaciones que después se promedian. Se indica con la letra s o S si se desean agregar más calificaciones o no.

[illegible]

```
#include <stdio.h>
int main()
{
    /* Al igual que en la estructura if-else, 0 -> falso y diferente de 0 -> verdadero. El siguiente es un c
    iclo infinito porque la condición siempre es verdadera.
    Así mismo, debido a que el ciclo consta de una sola línea, las llaves { } son opcionales.*/
    while (12)
    {
        printf("Ciclo infinito.\nPara terminar el ciclo presione ctrl + c.\n");
    }
    return 0;
}
```

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

~

## Suma de calificaciones

```
#include <stdio.h>
int main ()
{
    char op = 'n';
    double sum = 0, calif = 0; int veces = 0;
    do
    {
        printf("\tSuma de calificaciones\n");
        printf("Ingrese la calificación:\n");
        scanf("%lf", &calif);
        veces++;
        sum = sum + calif;
        printf("¿Desea sumar otra? S/N\n");
        setbuf(stdin, NULL); // limpia el buffer del teclado
        scanf("%c",&op);
        getchar();
    }
    while (op == 'S' || op == 's');
    printf("El promedio de las calificaciones ingresadas es: %lf\n", sum/veces);
    return 0;
}
```

~  
"Ejercicio\_3.c" 22L, 467C

```
Ecuador33:~ fp20alu49$ vi programa3.c
Ecuador33:~ fp20alu49$ gcc programa3.c -o programa3.out
Ecuador33:~ fp20alu49$ ./programa3.out
    Suma de calificaciones
Ingrese la calificación:
10
¿Desea sumar otra? S/N
S
El promedio de las calificaciones ingresadas es: 10.000000
Ecuador33:~ fp20alu49$
Ecuador33:~ fp20alu49$
```

El código genera un menú de operaciones básicas, sumar, restar, multiplicar o dividir; el usuario indica qué operación quiere realizar y después indica los valores con los cuales será ejecutada

```
#include <stdio.h>
int main ()
{
    int op, uno, dos;
    do
    {
        printf(" --- Calculadora ---\n");
        printf("\n¿Qué desea hacer\n");
        printf("1) Sumar\n");
        printf("2) Restar\n");
        printf("3) Multiplicar\n");
        printf("4) Dividir\n");
        printf("5) Salir\n");
        scanf("%d",&op);
        switch(op)
        {
            case 1:
                printf("\tSumar\n");
                printf("Introduzca los números a sumar separados por comas\n");
                scanf("%d,%d",&uno,&dos);
                printf("%d + %d = %d\n", uno, dos, (uno + dos));
                break;
            case 2:
                printf("\tRestar\n");
                printf("Introduzca los números a restar separados por comas\n");
                scanf("%d,%d",&uno,&dos);
                printf("%d - %d = %d\n", uno, dos, (uno - dos));
                break;
            case 3:
                printf("\tMultiplicar\n");
                printf("Introduzca los números a multiplicar separados por comas\n");
                scanf("%d,%d",&uno,&dos);
                printf("%d * %d = %d\n", uno, dos, (uno * dos));
                break;
            case 4:
                printf("\tDividir\n");
                printf("Introduzca los números a dividir separados por comas\n");
                scanf("%d,%d",&uno,&dos);
                printf("%d / %d = %d\n", uno, dos, (uno / dos));
                break;
            case 5:
                printf("Salir\n");
                break;
        }
    }
    while (op != 5);
    return 0;
}
```

"Ejercicio\_4.c" 51L, 1139C

```

case 3:
printf("\tMultiplicar\n");
printf("Introduzca los números a multiplicar separados por comas\n");
scanf("%d, %d", &uno, &dos);
printf("%d * %d = %d\n", uno, dos, (uno * dos));
break;
case 4:
printf("\tDividir\n");
printf("Introduzca los números a dividir separados por comas\n");
scanf("%d, %d", &uno, &dos);
printf("%d / %d = %.2lf\n", uno, dos, ((double)uno / dos));
break;
case 5:
printf("\tSalir\n");
break;

default:
printf("\tOpción inválida.\n");
}
}
while (op!=5);
}

```

```

Ecuador33:~ fp20alu49$ vi programa4.c
Ecuador33:~ fp20alu49$ gcc programa4.c -o programa4.out
Ecuador33:~ fp20alu49$ ./programa4.c
-bash: ./programa4.c: Permission denied
Ecuador33:~ fp20alu49$ ./programa4.out
--- Calculadora ---

```

```

¿Qué desea hacer
1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
2
      Restar
Introduzca los números a restar separados por comas
10,5
10 - 5 = 5
--- Calculadora ---

```

```

¿Qué desea hacer
1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
5
      Salir
Ecuador33:~ fp20alu49$

```

Programa para calificaciones de alumnos}

```

#include <stdio.h>
int main ()
{
    int enteroNumAlumnos = 5;
    float realCalif = 0.0, realPromedio = 0.0;
    printf("\tPromedio de calificaciones\n");
    for (int indice = 0 ; indice < enteroNumAlumnos ; indice++)
    {
        printf("\nIngrese la calificación del alumn %d\n", indice+1);
        scanf("%f",&realCalif);
        realPromedio += realCalif;
    }
    printf("\nEl promedio de las calificaciones ingresadas es: %f\n",
        realPromedio/enteroNumAlumnos);
    return 0;
}
~
~
~
~
~
~
~
"Ejercicio_5.c" 16L, 444C

```

```

Ecuador33:~ fp20alu49$ gcc programa5.c -o programa5.out
Ecuador33:~ fp20alu49$ ./programa5.out
    Promedio de calificaciones

Ingrese la calificación del alumn 1
7

Ingrese la calificación del alumn 2
9

Ingrese la calificación del alumn 3
10

Ingrese la calificación del alumn 4
5

Ingrese la calificación del alumn 5
6

El promedio de las calificaciones ingresadas es: 7.400000
Ecuador33:~ fp20alu49$ █

```

fp20alu49 — vi programa5.c — 104x62

```
#include <stdio.h>
int main ()
{
    int enteroNumAlumnos = 5;
    float realCalif = 0.0, realPromedio = 0.0;
    printf("\tPromedio de calificaciones\n");
    for (int indice = 0 ; indice < enteroNumAlumnos ; indice++)
    {
        printf("\nIngrese la calificación del alumno %d\n", indice+1);
        scanf("%f",&realCalif);
        realPromedio += realCalif;
    }

    printf("\nEl promedio de las calificaciones ingresadas es: %f\n",
        realPromedio/enteroNumAlumnos);
    return 0;
}
```

```
"programa5.c" 16L, 444C
```

```
Ecuador33:~ fp20alu49$ vi programa6.c
Ecuador33:~ fp20alu49$ gcc programa6.c -o programa6.out
Ecuador33:~ fp20alu49$ ./programa6.out
Ingrese el valor 1 del arreglo: 7
Ingrese el valor 2 del arreglo: 3
Ingrese el valor 3 del arreglo: 4
Ingrese el valor 4 del arreglo: 1
Ingrese el valor 5 del arreglo: 8
El valor ingresado para cada elemento del arreglo es:
[7 3 4 1 8 ]
Ecuador33:~ fp20alu49$
```





## Ejercicios de tarea

### 1. Solicitar al usuario el radio y calcular el área

```
+
> ...
24 case 2: {
25     printf("\n )Ingrese el tamaño del largo: ");
26     scanf("%f",&L1);
27     printf("\n )Ingrese el tamaño del ancho: ");
28     scanf("%f",&L2);
29     Res=(2*(L1+L2));
30     printf("\n El perimetro es: %.4f",Res);
31     Res=(L1*L2);
32     printf("\n El área es: %.4f\n",Res);
33     break;
34 }
35 case 3: {
36     printf("\n )Ingrese el tamaño de la base: ");
37     scanf("%f",&L1);
38     printf("\n )Ingrese el tamaño de la altura: ");
39     scanf("%f",&L2);
40     Res=(L1*L2)/2;
41     printf("\n El área es: %.4f\n",Res);
42     break;
43 }
44
45 case 0: {
46     printf("\n Nos vemos luego 🍌\n\n");
47     break;
48 }
49 default: {
50     printf("\t oops! 🍌 Intentelo de nuevo 🍌 \n");
51     break;
52 }
53 }
54 }
55 while(Ans!=0);
56 return 0;
57 }
```

```
> make -s
> ./main

<<<< Poligonos >>>>

| Seleccione entre las diferentes figuras |
1) Cuadrado
2) Rectangulo
3) Triangulo

>>> 0) Salir

*)1

)Ingrese el tamaño del lado: 2

El perimetro es: 8.0000
El área es: 4.0000

<<<< Poligonos >>>>

| Seleccione entre las diferentes figuras |
1) Cuadrado
2) Rectangulo
3) Triangulo

>>> 0) Salir

*)2

)Ingrese el tamaño del largo: 3
)Ingrese el tamaño del ancho: 2

El perimetro es: 10.0000
El área es: 6.0000

<<<< Poligonos >>>>

| Seleccione entre las diferentes figuras |
1) Cuadrado
2) Rectangulo
3) Triangulo

>>> 0) Salir

*)0
```

### 2. Mostrar numeración del 10 al 1 con ciclo while

```
+
> ...
1 #include <stdio.h>
2 #include <stdlib.h>
3 //
4 int main(){
5     int Res, n, i, Ans;
6     do {
7         printf("\n<<<<< Cuenta regresiva >>>>> Ingrese el número desde el cual\ncomenzará la cuenta regresiva: ");
8         scanf("%d",&n);
9         printf("\n");
10        for(i=0;i<n;i++){
11            Res=(n-i);
12            printf("\td \n",Res);
13        }
14        printf("\n🍌🍌🍌 Feliz Año nuevo!! 🍌🍌🍌\n\n");
15        printf("\tContamos de nuevo?\n\n 1)Si\n 2)Mejor luego\n\n\t *");
16        scanf("%d",&Ans);
17    }
18    while(Ans!=2);
19    return 0;
20 }
```

```
> make -s
> ./main

<<<<< Cuenta regresiva >>>>>

Ingresa el número desde el cual
comenzará la cuenta regresiva: 10

10
9
8
7
6
5
4
3
2
1

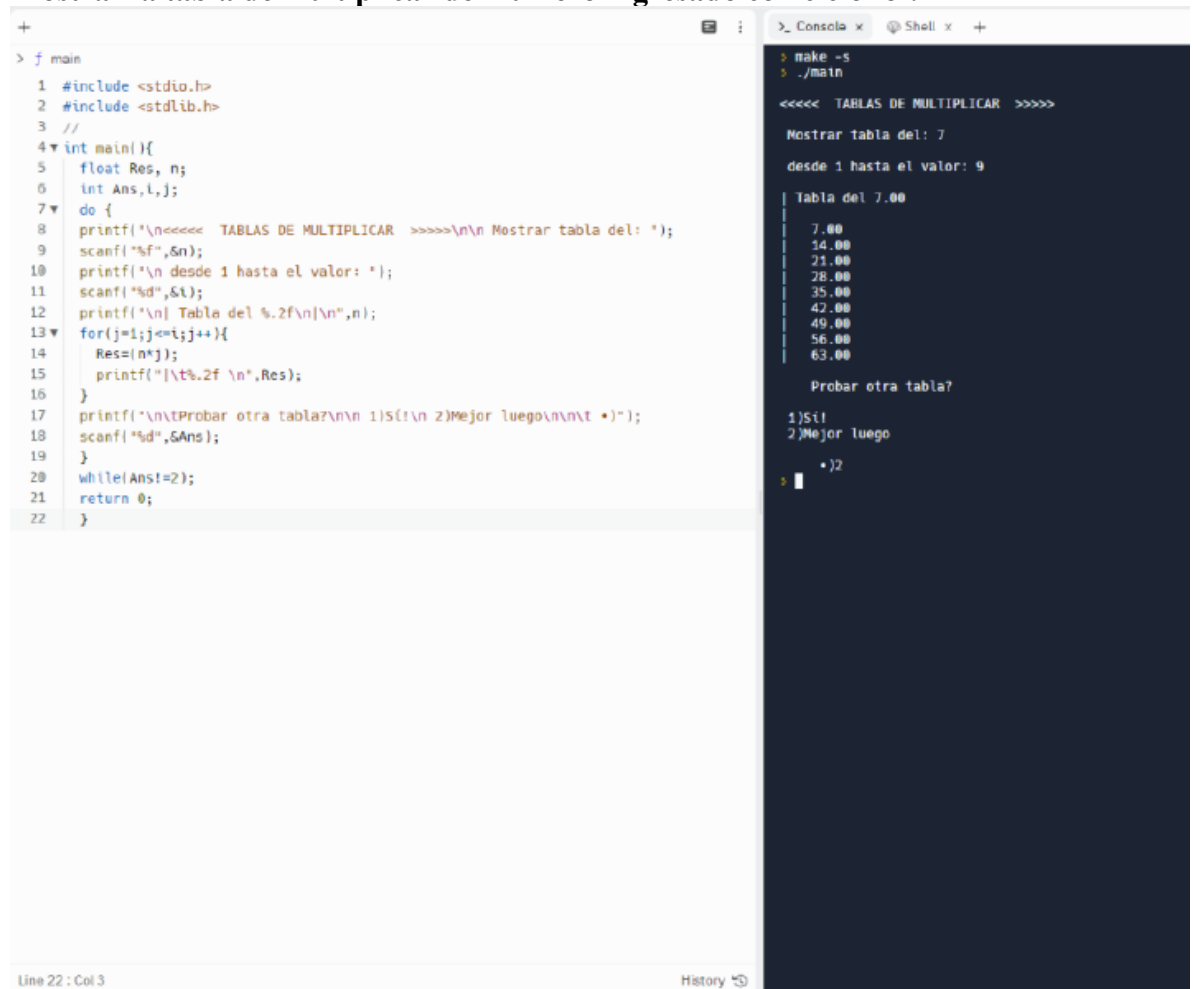
🍌🍌🍌 Feliz Año nuevo!! 🍌🍌🍌

Contamos de nuevo?

1)Si
2)Mejor luego

*)2
```

3. Mostrar la tabla de multiplicar del número ingresado con ciclo for.



The image shows a C program in a code editor and its execution in a terminal. The program is designed to calculate multiplication tables for a user-defined number. It includes headers for standard input/output and library functions. The main function prompts the user to enter a number, then uses a for loop to calculate and display the multiplication table from 1 to 9. It also asks if the user wants to try another table and loops back if the answer is 1.

```
> f main
1 #include <stdio.h>
2 #include <stdlib.h>
3 //
4 int main(){
5     float Res, n;
6     int Ans,i,j;
7     do {
8         printf("\n<<<< TABLAS DE MULTIPLICAR >>>>\n\nMostrar tabla del: ");
9         scanf("%f",&n);
10        printf("\n desde 1 hasta el valor: ");
11        scanf("%d",&i);
12        printf("\n| Table del %.2f\n|\n",n);
13        for(j=1;j<=i;j++){
14            Res=(n*j);
15            printf("| \t%.2f \n",Res);
16        }
17        printf("\n\n\tProbar otra tabla?\n\n 1)Si(\n 2)Mejor luego\n\n\t *)-");
18        scanf("%d",&Ans);
19    }
20    while(Ans!=2);
21    return 0;
22 }
```

Terminal output:

```
> make -s
> ./main
<<<< TABLAS DE MULTIPLICAR >>>>
Mostrar tabla del: 7
desde 1 hasta el valor: 9
| Table del 7.00
|
| 7.00
| 14.00
| 21.00
| 28.00
| 35.00
| 42.00
| 49.00
| 56.00
| 63.00
|
Probar otra tabla?
1)Si
2)Mejor luego
 * )2
>
```

4. Genera un menú con los 3 ejercicios anteriores y agrega el ciclo do-while con el fin de que el usuario no pueda salir del programa hasta seleccionar la opción en el menú.



```

+
> f main
41     printf("\n Ingrese el tamaño de la altura: ");
42     scanf("%f",&L2);
43     Res=(L1*L2)/2);
44     printf("\n El área es: %.4f\n",Res);
45     break;
46 }
47 ▼ case 0: {
48     printf("\n Nos vemos luego 🍌\n\n");
49     break;
50 }
51 ▼ default: {
52     printf("\t oops! 😞 Intentalo de nuevo 🍌 \n");
53     break;
54 }
55 }
56 }
57 while(Ans!=0);
58 break;
59 }
60 // COUNTDOWN
61 ▼ case 2:{
62 ▼ do {
63     printf("\n===== CUESTA REGRESIVA >>>>\n\n Ingrese el número desde el cual\n
comenzará la cuenta regresiva: ");
64     scanf("%d",&n);
65     printf("\n");
66 ▼ for(i=0;i<n;i++){
67         Res=(n-i);
68         printf("\t%d \n",Res);
69     }
70     printf("\n🍌🍌🍌 Feliz Año nuevo!! 🍌🍌🍌\n\n");
71     printf("\tContamos de nuevo?\n\n 1)Si!\n 2)Mejor luego\n\n\t*");
72     scanf("%d",&Ans);
73 }
74 while(Ans!=2);
75 break;
76 }
77 // MULTIPLICATION CHARTS
78 ▼ case 3:{
79 ▼ do {
80     printf("\n===== TABLAS DE MULTIPLICAR >>>>\n\n Mostrar tabla del: ");
81     scanf("%f",&k);
82     printf("\n desde 1 hasta el valor: ");

```

```

>_ Console x Shell x +
<<<< CUESTA REGRESIVA >>>>

Ingresa el número desde el cual
comenzará la cuenta regresiva: 5

5
4
3
2
1

*** Feliz Año nuevo!! ***

Contamos de nuevo?

1)Si!
2)Mejor luego

*2

| >>>>>>> EL GRAN MENÚ <<<<<<< |

1)Programa < Poligonos >
2)Programa < Cuenta regresiva >
3)Programa < Tablas de multiplicar >

>>> 0)Salir

Seleccione su Programa<-

*3

<<<< TABLAS DE MULTIPLICAR >>>>

Mostrar tabla del: 5

desde 1 hasta el valor: 12

| Tabla del 5.00
|
| 5.00
| 10.00
| 15.00
| 20.00
| 25.00
| 30.00
| 35.00
| 40.00
| 45.00
| 50.00
| 55.00
| 60.00
|

Probar otra tabla?

1)Si!
2)Mejor luego

*2

```

```
+
> f main
68     printf("%d\n",Res);
69 }
70 printf("\n🎉🎉 Feliz Año nuevo!! 🎉🎉\n\n");
71 printf("\tContamos de nuevo2\n\n 1)Si\n 2)Mejor luego\n\n\t*");
72 scanf("%d",&Ans);
73 }
74     while(Ans!=2);
75     break;
76 }
77 // MULTIPLICATION CHARTS
78 case 3:{
79     do {
80         printf("\n===== TABLAS DE MULTIPLICAR =====\n\nMostrar tabla del: ");
81         scanf("%f",&k);
82         printf("\n desde 1 hasta el valor: ");
83         scanf("%d",&i);
84         printf("\n| Tabla del %.2f\n|\n",k);
85         for(j=1;j<=i;j++){
86             Res=(k*j);
87             printf("\t%.2f \n",Res);
88         }
89         printf("\n\tProbar otra tabla?\n\n 1)Si\n 2)Mejor luego\n\n\t*");
90         scanf("%d",&Ans);
91     }
92     while(Ans!=2);
93     break;
94 }
95 //OUT GATE
96 case 0:{
97     printf("\n\n\t>> Hasta pronto <<\n\n");
98     break;
99 }
100 //ERROR
101 default:{
102     printf("\t oops! 😞 Intentalo de nuevo 2 😊 \n");
103     break;
104 }
105 }
106 }
107 while(GM!=0);
108 return 0;
109 }
```

```
_ Console x Shell x +
2)Mejor luego
+ )2
| >>>>>> EL GRAN MENU <<<<<<< |
1)Programa < Poligonos >
2)Programa < Cuenta regresiva >
3)Programa < Tablas de multiplicar >
>>> 0)Salir
Seleccione su Programa<
+ )3
<<<<< TABLAS DE MULTIPLICAR >>>>>
Mostrar tabla del: 5
desde 1 hasta el valor: 12
Tabla del 5.00
5.00
10.00
15.00
20.00
25.00
30.00
35.00
40.00
45.00
50.00
55.00
60.00
Probar otra tabla?
1)Si
2)Mejor luego
+ )2
| >>>>>> EL GRAN MENU <<<<<<< |
1)Programa < Poligonos >
2)Programa < Cuenta regresiva >
3)Programa < Tablas de multiplicar >
>>> 0)Salir
Seleccione su Programa<
+ )0
>> Hasta pronto <<
```

5. Del ejercicio de la calculadora, agregar condición a la división, si hay división entre cero pedir otro valor al usuario hasta que coloque un valor diferente y se pueda realizar la división.

```
+
> f main
1 #include <stdio.h>
2 #include <stdlib.h>
3 //var
4 float a,b,res1;
5 int Ans;
6 //
7 int main(){
8     do {
9         printf("\n>>>> DIVISIONES <<<<\n\t *) Dividendo: ");
10        scanf("%f",&a);
11        printf("\t *) Divisor: ");
12        scanf("%f",&b);
13        while(b==0){
14            printf("Valor fuera del rango;\nintente de nuevo\n\n");
15            printf("\t *) Divisor: ");
16            scanf("%f",&b);
17        }
18        res1=a/b;
19        printf("\n\nSu división es: %.2f / %.2f\n\t RESULTADO: %.4f",a,b,res1);
20        printf("\n\ndesea probar otra división?\n\t - 1 ) Si\n\t - 2 ) No\n\n>>> *");
21        scanf("%d",&Ans);
22    }
23    while(Ans!=2);
24    return 0;
25 }
```

```
> make -s
> ./main

>>>> DIVISIONES <<<<

 *) Dividendo: 7
 *) Divisor: 0
Valor fuera del rango;
intente de nuevo

 *) Divisor: 7

Su división es: 7.00 / 7.00
RESULTADO: 1.0000

desea probar otra división?
- 1 ) Si
- 2 ) No

>>> *)2
>
```

Line 25 : Col 4 History

6. Modificar el programa para que permita solicitar al usuario el Número de alumnos a evaluar empleando un ciclo while en lugar del ciclo for.

```
+
> ...
1 #include <stdio.h>
2 #include <stdlib.h>
3 //
4 int main () {
5     int enteroNumAlumnos=1, indice=0;
6     float realCalif = 0.0, realPromedio = 0.0;
7     printf("\tPromedio de calificaciones\n");
8     printf("\n cuántos alumnos se evaluarán? * ) ");
9     scanf("%d", &enteroNumAlumnos);
10    //Ciclo
11    while(indice<enteroNumAlumnos){
12        indice++;
13        printf("\nIngrese la calificación del alumno %d\n", indice);
14        scanf("%f", &realCalif);
15        realPromedio += realCalif;
16    }
17    printf("\nel promedio de las calificaciones ingresadas es: %.4f\n",
18        realPromedio/enteroNumAlumnos);
19    return 0;
20 }
```

```
> _ Console x Shell x +
> make -s
> ./main
Promedio de calificaciones

cuántos alumnos se evaluarán? * ) 5
Ingrese la calificación del alumno 1
7
Ingrese la calificación del alumno 2
10
Ingrese la calificación del alumno 3
10
Ingrese la calificación del alumno 4
9
Ingrese la calificación del alumno 5
6

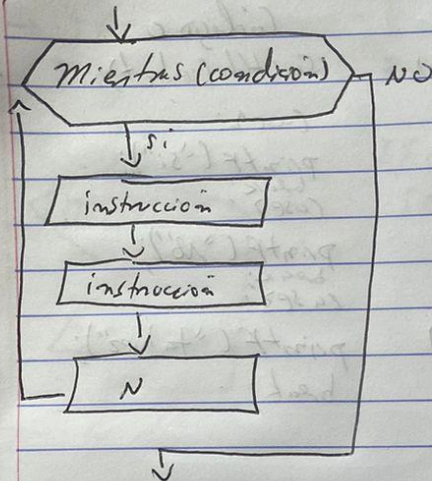
El promedio de las calificaciones ingresadas es: 8.40
> 
```

Line 20 : Col 2 History \*S

7. Completa el cuadro con las estructuras iterativas (MIENTRAS, HACER MIENTRAS Y PARA)



Mientras



Pseudocódigo

mientras (condición) hacer

Instrucción 1  
Instrucción 2  
.  
.  
Instrucción N

Repetir

Código

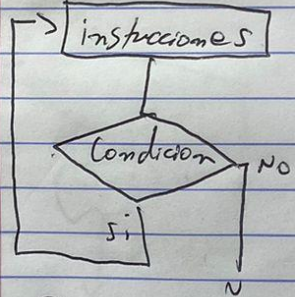
mientras

{  
  <Instrucción 1>  
  <Instrucción 2>  
  .  
  .  
  <Instrucción N>  
}

Repetir

}

Hacer mientras



Pseudocódigo

Hacer

instrucciones

Mientras que expresión lógica

:

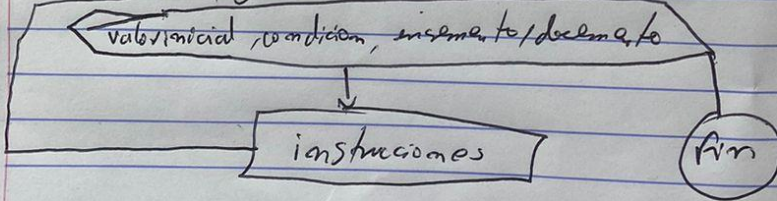
Código

Hacer

{  
  <bloque de instrucciones>  
  while (expresión lógica)

}

Para



Pseudo código	Código
Para valor inicial	For (Inicialización, condición, ejecución)
condición	{
incremento/decremento	Acción 1;
instrucciones	Acción 2;
Fin Para	:
	}

### Conclusión:

En esta práctica aprendí a usar y ejecutar los diferentes ciclos que existen ya en una terminal. Son medio complicados de comprender al inicio, pero al ponerlo en práctica después de hacerlo en el cuaderno te empieza a cuadrar su funcionamiento aplicado.

También facilita la creación de programas y simplifica muchos pasos que sin saber estos ciclos como el while sería muy largo y complicado el funcionamiento de un programa sin esto.