



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Marco Antonio Martínez Quintana

Asignatura: Fundamentos de Programación

Grupo: 3

No de Práctica(s): 9

Integrante(s): Ramírez Pérez Daniela Itzel

*No. de Equipo de
cómputo empleado:* No aplica

No. de Lista o Brigada: 36

Semestre: 1er

Fecha de entrega: Viernes 4 de Diciembre

Observaciones:

CALIFICACIÓN: _____

Estructuras de repetición

Objetivo:

Elaborar programas en C para la resolución de problemas básicos que incluyan las estructuras de repetición y la directiva define.

Introducción

Las estructuras de repetición permiten ejecutar un conjunto de instrucciones de manera repetida (o cíclica) mientras que la expresión lógica sea verdadera.

En lenguaje C existen tres estructuras de repetición:

- **while**,
- **do-while** y
- **for**

Estructura de control repetitiva while

La estructura repetitiva (o iterativa) while primero valida la expresión lógica y si es verdadera procede a ejecutar el bloque de instrucciones de la estructura. Si la condición no se cumple se continúa el flujo normal del programa.

```
while (expresión_lógica) {  
    // Bloque de código a repetir  
    // mientras que la expresión  
    // lógica sea verdadera.  
}
```

Si el bloque de código a repetir consta de una sola sentencia, entonces se pueden omitir las llaves.

Código (estructura de repetición while)

```
#include <stdio.h>  
/*  
    Este programa genera la tabla de multiplicar de un número dado.  
    El número se lee desde la entrada estándar (teclado).  
*/  
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 <= 10)  
        printf("%d x %d = %d \n", num, cont, num*cont);  
    return 0;  
}
```

```
----- Tabla de multiplicar -----
Ingrese un número:
7
La tabla de multiplicar del 7 es:
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
7 x 6 = 42
7 x 7 = 49
7 x 8 = 56
7 x 9 = 63
7 x 10 = 70
```

Código (estructura de repetición while)

```
#include <stdio.h>
/*
Este programa genera un ciclo infinito.
*/
int main(){
// Al igual que en la estructura if-else
// 0 -> falso
// diferente de 0 -> verdadero

// El siguiente es un ciclo infinito
// porque la condición siempre es verdadera.
// Así mismo, debido a que el ciclo consta de una sola línea, las
// llaves { } son opcionales.

while (100) {
printf("Ciclo infinito.\nPara terminar el ciclo presione ctrl + c.\n");
}
return 0;
}
```

[illegible]

Estructura de control repetitiva do-while

do-while es una estructura cíclica que ejecuta el bloque de código que se encuentra dentro de las llaves y después valida la condición. Su sintaxis es la siguiente:

```
do {  
    /*  
    Bloque de código que se ejecuta  
    por lo menos una vez y se repite  
    mientras la expresión lógica sea  
    verdadera.  
    */  
} while (expresión_lógica);
```

Si el bloque de solo se repetirá una vez se omiten las llaves {}.

Código (estructura de repetición do-while)

```
#include <stdio.h>  
/*  
Este programa obtiene el promedio de calificaciones ingresadas por  
el usuario. Las calificaciones se leen desde la entrada estándar (teclado).  
La inserción de calificaciones termina cuando el usuario presiona una tecla  
diferente de 'S' o 's'.  
*/  
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; }
```

```
Suma de calificaciones
Ingrese la calificación:
4
¿Desea sumar otra? S/N
S
Suma de calificaciones
Ingrese la calificación:
23
¿Desea sumar otra? S/N
n
El promedio de las calificaciones ingresadas es: 13.500000

C:\Users\twins\Desktop\Lenguaje C\Ejemplos>gcc practica9.c -o practica9.exe

C:\Users\twins\Desktop\Lenguaje C\Ejemplos>practica9.exe
Suma de calificaciones
Ingrese la calificación:
9
¿Desea sumar otra? S/N
S
Suma de calificaciones
Ingrese la calificación:
10
¿Desea sumar otra? S/N
S
Suma de calificaciones
Ingrese la calificación:
7
¿Desea sumar otra? S/N
n
El promedio de las calificaciones ingresadas es: 8.666667
```

Código (estructura de repetición do-while)

```
#include <stdio.h>
/* Este programa genera una calculadora básica. */
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 = %.2lf\n", uno, dos, ((double)uno / dos));
                break;
            case 5:
                printf("\tSalir\n");
                break;
            default:
                printf("\tOpción inválida.\n");
        }
    } while (op != 5);
    return 0;
}
```

```
{Qué desea hacer?
1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
1
    Sumar
Introduzca los números a sumar separados por comas
5,2
5 + 2 = 7
--- Calculadora ---

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

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

{Qué desea hacer?
1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
4
    Dividir
Introduzca los números a dividir separados por comas
5,2
5 / 2 = 2.50
--- Calculadora ---

{Qué desea hacer?
1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
5
    Salir
```


Estructura de control de repetición for

Lenguaje C posee la estructura de repetición for la cual permite realizar repeticiones cuando se conoce el número de elementos que se quiere recorrer. La sintaxis que generalmente se usa es la siguiente:

```
for (inicialización ; expresión_lógica ; operaciones por iteración) {  
    /*  
    Bloque de código  
    a ejecutar  
    */  
}
```

La estructura for ejecuta 3 acciones básicas antes o después de ejecutar el bloque de código.

- **Inicialización.** Se definen e inicializan las variables esta parte solo se ejecuta una vez cuando se ingresa al ciclo y es opcional.
- **Expresión Lógica** Se evalúa si la expresión es verdadera para que se ejecute el bloque de código; es opcional.
- **Conjunto de Operaciones** Se realiza esta operación cada vez que termina de ejecutarse el bloque de código y antes de volver a validar la expresión lógica; esta parte también es opcional.

Código (estructura de repetición for)

```
#include <stdio.h>
/* Este programa genera un arreglo unidimensional de 5 elementos y
 * accede a cada elemento del arreglo a través de un ciclo for.
 */
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; }
```

```
    Promedio de calificaciones

Ingrese la calificación del alumn 1
10

Ingrese la calificación del alumn 2
8

Ingrese la calificación del alumn 3
9

Ingrese la calificación del alumn 4
8

Ingrese la calificación del alumn 5
7

El promedio de las calificaciones ingresadas es: 8.400000
```

Define

Las líneas de código que empiezan con # son directivas del preprocesador, el cual se encarga de realizar modificaciones en el texto del código fuente, como reemplazar un símbolo definido con #define por un parámetro o texto, o incluir un archivo en otro archivo con

```
#include.
```

define permite definir constantes o literales; se les nombra también como constantes simbólicas. Su sintaxis es la siguiente:

```
#define <nombre> <valor>
```

Al definir la constante simbólica con #define, se emplea un nombre y un valor. Cada vez que aparezca el nombre en el programa se cambiará por el valor definido. El valor puede ser numCódigo (define)

```
#include <stdio.h>
#define MAX 5
/*
 * Este programa define un valor por defecto para el tamaño del arreglo
 * de tal manera que si el tamaño de éste cambia, solo se debe modificar
 * el valor de la constante MAX.
 * */
int main () {
    int arreglo[MAX], cont;
    for (cont=0; cont<MAX; cont++){
        printf("Ingrese el valor %d del arreglo: ", cont+1);
        scanf("%i", &arreglo[cont]);
    }
    printf("El valor ingresado para cada elemento del arreglo es:\n[");
    for (cont=0; cont<MAX; cont++){
        printf("%d\t", arreglo[cont]);
    }
    printf("]\n");
    return 0;
}
```

```
Ingrese el valor 1 del arreglo: 7
Ingrese el valor 2 del arreglo: 8
Ingrese el valor 3 del arreglo: 9
Ingrese el valor 4 del arreglo: 2
Ingrese el valor 5 del arreglo: 6
El valor ingresado para cada elemento del arreglo es:
[7      8      9      2      6      ]
```

Cuando se compila el programa, se reemplazan la palabra MAX por el valor definido para la misma. Esto permite que, si el tamaño del arreglo cambia, solo se tiene que modificar el valor definido para MAX y en automático todos los arreglos y el recorrido de los mismos adquieren el nuevo valor (Mientras se use MAX para definir el o los

arreglos y para realizar los recorridos).

Break

Algunas veces es conveniente tener la posibilidad de abandonar un ciclo. La proposición `break` proporciona una salida anticipada dentro de una estructura de repetición, tal como lo hace en un `switch`. Un `break` provoca que el ciclo que lo encierra termine inmediatamente.

Código (break)

```
#include <stdio.h>
/*
 * Este programa hace una suma de números. Si la suma rebasa la cantidad
 * de 50 el programa se detiene.
 */
#define VALOR_MAX 5
int main () {
    int enteroSuma = 0;
    int enteroNumero = 0;
    int enteroContador = 0;
    while (enteroContador < VALOR_MAX) {
        printf("Ingrese un número:");
        scanf("%d", &enteroNumero);
        enteroSuma += enteroNumero;
        enteroContador++;
        if (enteroSuma > 50) {
            printf("Se rebasó la cantidad límite.\n");
            break;
        }
        printf("El valor de la suma es: %d\n", enteroSuma);
        return 0;
    }
}
```

```
Ingrese un número:5
Ingrese un número:3
Ingrese un número:8
Ingrese un número:5
Ingrese un número:9
El valor de la suma es: 30
```

Cuando se compila el programa, `MAX` se sustituye por `5`.

Continue

La proposición `continue` provoca que inicie la siguiente iteración del ciclo de repetición que la contiene.

Código (continue)

```
#include <stdio.h>
/*
 * Este programa obtiene la suma de un LIMITE de números pares ingresados
 * */
#define LIMITE 5
int main (){
    int enteroContador = 1;
    int enteroNumero = 0;
    int enteroSuma = 0;
    while (enteroContador <= LIMITE){
        printf("Ingrese número par %d:", enteroContador);
        scanf("%d",&enteroNumero);
        if (enteroNumero%2 != 0){
            printf("El número insertado no es par.\n");
            continue;
        }
        enteroSuma += enteroNumero;
        enteroContador++;
    }
    printf("La suma de los números es: %d\n", enteroSuma);
    return 0;
}
```

```
Ingrese número par 1:1
El número insertado no es par.
Ingrese número par 1:2
Ingrese número par 2:3
El número insertado no es par.
Ingrese número par 2:4
Ingrese número par 3:6
Ingrese número par 4:8
Ingrese número par 5:2
La suma de los números es: 22
```

Actividades

Estructuras de repetición

```
areaPerimetro.c x almacenarV.c x estYCons.c x codigoOperadores.c x expLog.c x gauss.c x
1  #include<stdio.h>
2  int main()
3  {
4      //Declarar variable
5      char au=163, sp=168, aa=160;
6      int n,res;
7
8      //Mensaje de bienvenida
9      printf("\n\n\t\t\tSuma de los primeros n números\n\n",au);
10
11     //Solicitar el numero de los elementos a sumar
12     printf("\n¿Cuántos números deseas sumar?",sp,aa,au);
13     scanf("%d",&n);
14
15     //Sumar los n numeros
16     res=0;
17     for(int i=1;i<=n;i++)
18     {
19         res=res+i;
20     }
21
22     //Mostrar el resultado
23     printf("La suma de los primeros %d números es: %d\n",n,au,res);
24
25     return 0;
26 }
27
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>gauss.exe
```

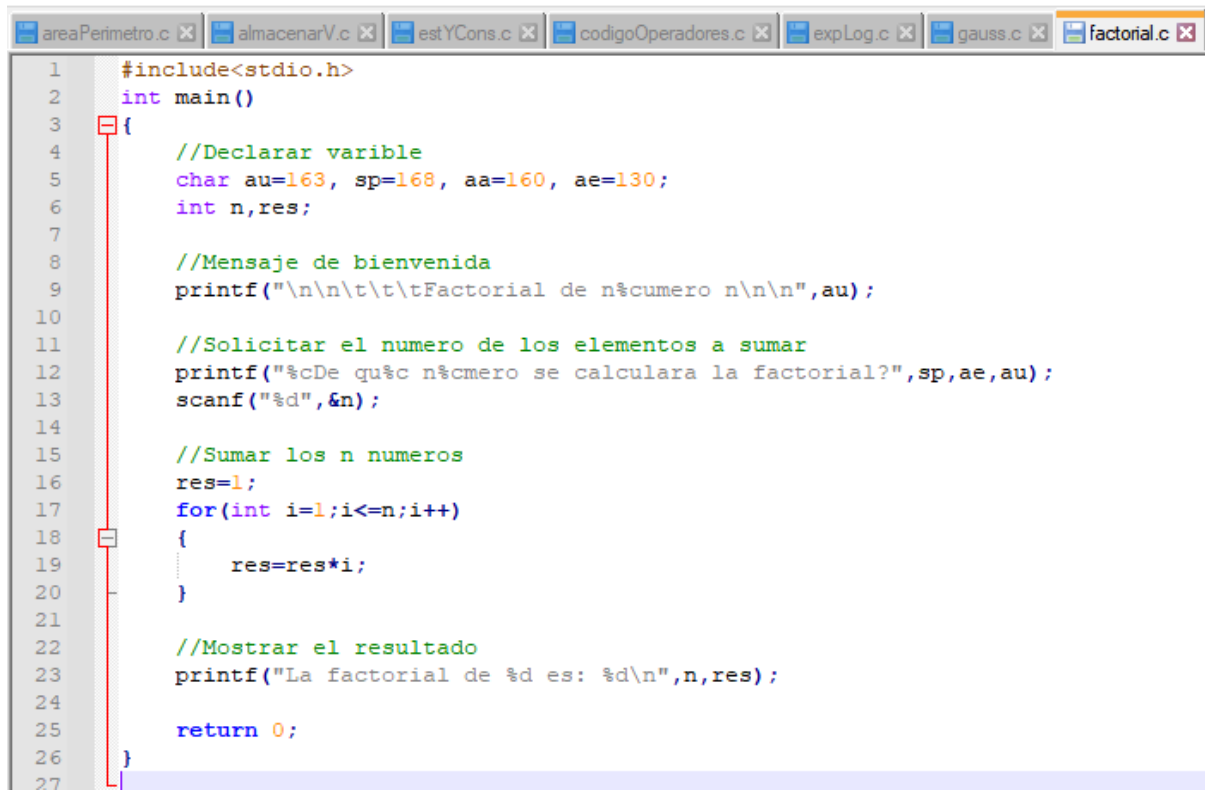
```
Suma de los primeros n números
```

```
¿Cuántos números deseas sumar?5
```

```
La suma de los primeros 5 números es: 15
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>
```

Ejercicio 4



The image shows a screenshot of a C code editor with multiple tabs open at the top: 'areaPerimetro.c', 'almacenarV.c', 'estYCons.c', 'codigoOperadores.c', 'expLog.c', 'gauss.c', and 'factorial.c'. The 'factorial.c' tab is active. The code is as follows:

```
1  #include<stdio.h>
2  int main()
3  {
4      //Declarar variable
5      char au=163, sp=168, aa=160, ae=130;
6      int n,res;
7
8      //Mensaje de bienvenida
9      printf("\n\n\t\t\tFactorial de n%cumero n\n\n",au);
10
11     //Solicitar el numero de los elementos a sumar
12     printf("%cDe qu%c n%cmero se calculara la factorial?",sp,ae,au);
13     scanf("%d",&n);
14
15     //Sumar los n numeros
16     res=1;
17     for(int i=1;i<=n;i++)
18     {
19         res=res*i;
20     }
21
22     //Mostrar el resultado
23     printf("La factorial de %d es: %d\n",n,res);
24
25     return 0;
26 }
27
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorial.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?2
```

```
La factorial de 2 es: 2
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorial.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?3
```

```
La factorial de 3 es: 6
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorial.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?4
```

```
La factorial de 4 es: 24
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorial.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?5
```

```
La factorial de 5 es: 120
```

Ciclo while


```
areaPerimetro.c x almacenarV.c x estYCons.c x codigoOperadores.c x expLog.c x gauss.c x factorial.c x gaussWhile.c x
1  #include<stdio.h>
2  int main()
3  {
4      //Declarar variable
5      char au=163, sp=168, aa=160;
6      int i,n,res;
7
8      //Mensaje de bienvenida
9      printf("\n\n\t\tSuma de los primeros n números\n\n",au);
10
11     //Solicitar el numero de los elementos a sumar
12     printf("%cCuántos números deseas sumar?",sp,aa,au);
13     scanf("%d",&n);
14
15     //Sumar los n numeros
16     res=0;
17     i=1;
18     while(i<=n)
19     {
20         res=res+i;
21         i++;
22     }
23
24     //Mostrar el resultado
25     printf("La suma de los primeros %d números es: %d\n",n,au,res);
26
27     return 0;
28 }
29
```

C:\Users\twins\Desktop\Lenguaje C\Ejemplos>gaussWhile.exe

```
Suma de los primeros n números

¿Cuántos números deseas sumar?2
La suma de los primeros 2 números es: 3
```

C:\Users\twins\Desktop\Lenguaje C\Ejemplos>gaussWhile.exe

```
Suma de los primeros n números

¿Cuántos números deseas sumar?3
La suma de los primeros 3 números es: 6
```

C:\Users\twins\Desktop\Lenguaje C\Ejemplos>gaussWhile.exe

```
Suma de los primeros n números

¿Cuántos números deseas sumar?4
La suma de los primeros 4 números es: 10
```

Ejercicio 5

```
areaPerimetro.c x expLog.c x gauss.c x factorial.c x gaussWhile.c x factorialWhile.c x
1  #include<stdio.h>
2  int main()
3  {
4      //Declarar variable
5      char au=163, sp=168, aa=160, ae=130;
6      int i,n,res;
7
8      //Mensaje de bienvenida
9      printf("\n\n\t\t\tFactorial de ncurnero n\n\n",au);
10
11     //Solicitar el numero de los elementos a sumar
12     printf("%cDe quc ncurnero se calculara la factorial?",sp,ae,au);
13     scanf("%d",&n);
14
15     //Sumar los n numeros
16     res=1;
17     i=1;
18     while(i<=n)
19     {
20         res=res*i;
21         i++;
22     }
23
24     //Mostrar el resultado
25     printf("La factorial de %d es: %d\n",n,res);
26
27     return 0;
28 }
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorialWhile.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?2
```

```
La factorial de 2 es: 2
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorialWhile.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?3
```

```
La factorial de 3 es: 6
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorialWhile.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?4
```

```
La factorial de 4 es: 24
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>factorialWhile.exe
```

```
Factorial de número n
```

```
¿De qué número se calculara la factorial?5
```

```
La factorial de 5 es: 120
```

Ciclo do-while

```
areaPerimetro.c x expLog.c x gauss.c x factorial.c x menuSalir.c x
1  #include<stdio.h>
2  int main()
3  {
4      //Declarar variables
5      int op;
6      char aa=160, ae=130, ai=161, ao=162, au=163;
7
8      //Mensaje de bienvenida
9      printf("\n\n\t\t\tMen%c de figuras :)\n\n",au);
10
11     do
12     {
13         //Mostrar el menu
14         printf("1)Tri%cngulo\n2) Rect%cngulo\n3) C%crculo\n4) Salir\n",aa,aa,ai);
15
16         //Solicitar la opcion
17         printf("Elige una opci%cn: ",ao);
18         scanf("%d",&op);
19
20         switch(op)
21         {
22             case 1:
23                 printf("Elegiste Tri%cngulo\n\n",aa);
24                 break;
25             case 2:
26                 printf("Elegiste Rect%cngulo\n\n",aa);
27                 break;
28             case 3:
29                 printf("Elegiste C%crculo\n\n",ai);
30                 break;
31             case 4:
32                 printf("Elegiste Salir\n\n");
33             default:
34                 printf("Opci%cn no v%clida!!!\n\n",ao,aa);
35         }
36     } while(op!=4);
37     printf("Gracias por usar nuestro programa :)\n");
38     return 0;
39 }
40
```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>menuSalir.exe
```

```
Menú de figuras :)
```

```
1)Triángulo
2) Rectángulo
3) Círculo
4) Salir
Elige una opción: 1
Elegiste Triángulo
```

```
1)Triángulo
2) Rectángulo
3) Círculo
4) Salir
Elige una opción: 2
Elegiste Rectángulo
```

```
1)Triángulo
2) Rectángulo
3) Círculo
4) Salir
Elige una opción: 3
Elegiste Círculo
```

```
1)Triángulo
2) Rectángulo
3) Círculo
4) Salir
Elige una opción: 5
Opción no válida!!!
```

```
1)Triángulo
2) Rectángulo
3) Círculo
4) Salir
Elige una opción: 4
Elegiste Salir
```

```
Opción no válida!!!
```

```
Gracias por usar nuestro programa :)
```

Ciclo do-while (submenu)

```
areaPerimetro.c x expLog.c x gauss.c x factorial.c x menuSalir.c x submenu.c x
1  #include<stdio.h>
2  int main()
3  {
4      //Declarar variables
5      int op,op2;
6      char aa=l60, ae=l30, ai=l61, ao=l62, au=l63;
7
8      //Mensaje de bienvenida
9      printf("\n\n\t\tMen%c de figuras :)\n\n",au);
10
11     do
12     {
13         //Mostrar el menu
14         printf("1)Tri%cngulo\n2) Rect%cngulo\n3) C%crculo\n4) Salir\n",aa,aa,ai);
15
16         //Solicitar la opcion
17         printf("Elige una opci%c\n: ",ao);
18         scanf("%d",&op);
19
20         switch(op)
21         {
22             case 1:
23                 printf("Elegiste Tri%cngulo\n\n",aa);
24                 do
25                 {
26                     printf("1) %c%c\n2) per%cmetro\n3) salir\n",aa,ai);
27                     printf("Elige una opci%c\n: ",ao);
28                     scanf("%d",&op2);
29                     switch(op2)
30                     {
31                         case 1:
32                             printf("Elegiste el %c%c\n\n",aa);
33                             break;
34                         case 2:
35                             printf("Elegiste per%cmetro\n\n",ai);
36                             break;
37                         case 3:
38                             printf("Elegiste salir\n\n");
39                             break;
40                         default:
41                             printf("Opci%c\n no v%cclida!!!\n\n",ao,aa);
42                     }
43                 }while(op2!=3);
44                 break;
45             case 2:
46                 printf("Elegiste Rect%cngulo\n\n",aa);
47                 do
48                 {
49                     printf("1) %c%c\n2) per%cmetro\n3) salir\n",aa,ai);
50                     printf("Elige una opci%c\n: ",ao);
51                     scanf("%d",&op2);
52                     switch(op2)
53                     {
54                         case 1:
55                             printf("Elegiste el %c%c\n\n",aa);
56                             break;
57                         case 2:
58                             printf("Elegiste per%cmetro\n\n",ai);
59                             break;
60                         case 3:
61                             printf("Elegiste salir\n\n");
62                             break;
63                         default:
64                             printf("Opci%c\n no v%cclida!!!\n\n",ao,aa);
65                     }
66                 }while(op2!=3);
67                 break;
68             case 3:
69                 printf("Elegiste C%crculo\n\n",aa);
70                 do
71                 {
72                     printf("1) %c%c\n2) per%cmetro\n3) salir\n",aa,ai);
73                     printf("Elige una opci%c\n: ",ao);
74                     scanf("%d",&op2);
75                     switch(op2)
76                     {
77                         case 1:
78                             printf("Elegiste el %c%c\n\n",aa);
79                             break;
80                         case 2:
81                             printf("Elegiste per%cmetro\n\n",ai);
82                             break;
83                         case 3:
84                             printf("Elegiste salir\n\n");
85                             break;
86                         default:
87                             printf("Opci%c\n no v%cclida!!!\n\n",ao,aa);
88                     }
89                 }while(op2!=3);
90                 break;
91             case 4:
92                 printf("Elegiste Salir\n\n");
93                 break;
94             default:
95                 printf("Opci%c\n no v%cclida!!!\n\n",ao,aa);
96                 break;
97         }
98     }while(op!=4);
99 }
```

```

57
58         case 2:
59             printf("Elegiste per%cmetro\n\n",ai);
60             break;
61         case 3:
62             printf("Elegiste salir\n\n");
63             break;
64         default:
65             printf("Opci%cn no v%clida!!!\n\n",ao,aa);
66     }
67     }while(op2!=3);
68     break;
69 case 3:
70     printf("Elegiste C%ccrulo\n\n",ai);
71     do
72     {
73         printf("1) %screa\n2) per%cmetro\n3) salir\n",aa,ai);
74         printf("Elige una opci%cn: ",ao);
75         scanf("%d",&op2);
76         switch(op2)
77         {
78             case 1:
79                 printf("Elegiste el %screa\n\n",aa);
80                 break;
81             case 2:
82                 printf("Elegiste per%cmetro\n\n",ai);
83                 break;
84             case 3:
85                 printf("Elegiste salir\n\n");
86                 break;
87             default:
88                 printf("Opci%cn no v%clida!!!\n\n",ao,aa);
89         }
90     }while(op2!=3);
91     break;
92 case 4:
93     printf("Elegiste Salir\n\n");
94
95     default:
96         printf("Opci%cn no v%clida!!!\n\n",ao,aa);
97 }
98 while(op!=4);
99 printf("Gracias por usar nuestro programa :)\n\n");
100 return 0;
101 }
102

```

```
C:\Users\twins\Desktop\Lenguaje C\Ejemplos>submenu.exe
```

```
Menú de figuras :)
```

```
1)Triángulo  
2) Rectángulo  
3) Círculo  
4) Salir  
Elige una opción: 1  
Elegiste Triángulo
```

```
1) área  
2) perímetro  
3) salir  
Elige una opción: 1  
Elegiste el área
```

```
1) área  
2) perímetro  
3) salir  
Elige una opción: 2  
Elegiste perímetro
```

```
1) área  
2) perímetro  
3) salir  
Elige una opción: 3  
Elegiste salir
```

```
1)Triángulo  
2) Rectángulo  
3) Círculo  
4) Salir  
Elige una opción: 4  
Elegiste Salir
```

```
Opción no válida!!!
```

```
Gracias por usar nuestro programa :)
```


Tarea 5. Calculadora con opción de salir

```
areaPerimetro.c expLog.c gauss.c calculadoraOpc.c gaussWhile.c submenu.c factorial.c calculadoraEnC.c menuSalir.c factorialWhile.c
1 #include<stdio.h>
2 int main()
3 {
4     //Mensaje de bienvenida
5     printf("\n\n\t\tBienvenido a mi Calculadora en C :)\n\n");
6
7     //Declarar variables a utilizar
8     int num1,num2,sumRes, restaRes,mRes,facRes,gaussRes,op,op2,n,i;
9     char aa=160, ae=130, ai=161, ao=162, au=163, sp=168;
10
11
12
13     do
14     {
15         //Mostrar el menu
16         printf("\n1)Suma\n2) Resta\n3) Multiplicaci\u00f3n\n4) Divisi\u00f3n\n5) Modulo\n6) Factorial\n7) Suma de los primeros n n\u00fameros\n8) salir\n\n",ao,ao,au);
17
18         //Solicitar la opcion
19         printf("Elige una opci\u00f3n: ",ao);
20         scanf("%d",&op);
21
22         switch(op)
23         {
24             case 1:
25                 printf("Elegiste Suma\n\nElige una opci\u00f3n\n\n",ao);
26                 do
27                 {
28                     printf("\n1) Realizar la operaci\u00f3n\n2) Salir\n\n",ao);
29                     printf("Elige una opci\u00f3n: ",ao);
30                     scanf("%d",&op2);
31                     switch(op2)
32                     {
33                         case 1:
34                             printf("Dame 2 n\u00fameros separados por coma para sus operaciones: \n",au);
35                             scanf("%i,%i",&num1,&num2);
36                             sumRes=num1+num2;
37                             printf("La suma de %d y %d es %d \n\n",num1,num2,sumRes);
38                             break;
39                         case 2:
40                             printf("Elegiste Salir\n\n");
41                             break;
42                         default:
43                             printf("Opci\u00f3n no v\u00e1lida!!!\n\n",ao,aa);
44                     }
45                 }while(op2!=2);
46                 break;
47             case 2:
48                 printf("Elegiste Resta\n\nElige una opci\u00f3n\n\n",ao);
49                 do
50                 {
51                     printf("\n1) Realizar la operaci\u00f3n\n2) Salir\n\n",ao);
52                     printf("Elige una opci\u00f3n: ",ao);
53                     scanf("%d",&op2);
54                     switch(op2)
55                     {
56                         case 1:
57                             //...
```

```

56 {
57     case 1:
58         printf("Dame 2 nùmeros separados por coma para sus operaciones: ",au);
59         scanf("%i,%i",&num1,&num2);
60         restaRes=num1-num2;
61         printf("La resta de %d y %d es %d \n\n",num1,num2,restaRes);
62         break;
63     case 2:
64         printf("Elegiste salir\n\n");
65         break;
66     default:
67         printf("Opciòn no vùclida!!!\n\n",ao,aa);
68     }
69     }while(op2!=2);
70     break;
71 case 3:
72     printf("Elegiste Multiplicaciòn\n\nElige una opsiòn\n\n",ao,ao);
73     do
74     {
75         printf("1) Realizar la operaciòn2) Salir\n",ao);
76         printf("Elige una opsiòn: ",ao);
77         scanf("%d",&op2);
78         switch(op2)
79         {
80             case 1:
81                 printf("Dame 2 nùmeros separados por coma para sus operaciones: ",au);
82                 scanf("%i,%i",&num1,&num2);
83                 mRes=num1*num2;
84                 printf("La multiplicacion de %d y %d es %d \n\n",num1,num2,mRes);
85                 break;
86             case 2:
87                 printf("Elegiste salir\n\n");
88                 break;
89             default:
90                 printf("Opciòn no vùclida!!!\n\n",ao,aa);
91             }
92         }while(op2!=2);
93         break;
94 case 4:
95     printf("Elegiste Divisiòn\n\nElige una opsiòn\n\n",ao,ao);
96     do
97     {
98         printf("1) Realizar la operaciòn2) Salir\n",ao);
99         printf("Elige una opsiòn: ",ao);
100         scanf("%d",&op2);
101         switch(op2)
102         {
103             case 1:
104                 printf("Dame 2 nùmeros separados por coma para sus operaciones: ",au);
105                 scanf("%i,%i",&num1,&num2);
106                 float divRes;
107
108                 if (num2==0)
109                 {
110                     printf("La division entre %d y %d es indefinida");
111                 }
112                 else

```

```

112         else
113         {
114             divRes=((float)num1/(float)num2);
115             printf("La division de %d y %d es %.2f \n\n",num1,num2,divRes);
116         }
117         break;
118     case 2:
119         printf("Elegiste salir\n\n");
120         break;
121     default:
122         printf("Opci%cn no v%clida!!!\n\n",ao,aa);
123     }
124 }while(op2!=2);
125 break;
126 case 5:
127     printf("Elegiste Modulo\n\nElige una opci%cn\n\n",ao);
128     do
129     {
130         printf("1) Realizar la operaci%cn\n2) Salir\n",ao);
131         printf("Elige una opci%cn: ",ao);
132         scanf("%d",&op2);
133         switch(op2)
134         {
135             case 1:
136                 printf("Dame 2 n%cmmeros separados por coma para sus operaciones: ",au);
137                 scanf("%i,%i",&num1,&num2);
138                 float modRes;
139                 if (num2==0)
140                 {
141                     printf("No existe el modulo");
142                 }
143                 else
144                 {
145                     modRes=(num1%num2);
146                     printf("El modulo de %d y %d es %.2f \n\n",num1,num2,modRes);
147                 }
148                 break;
149             case 2:
150                 printf("Elegiste salir\n\n");
151                 break;
152             default:
153                 printf("Opci%cn no v%clida!!!\n\n",ao,aa);
154         }
155     }while(op2!=2);
156     break;
157 case 6:
158     printf("Elegiste Factorial\n\nElige una opci%cn\n\n",ao);
159     do
160     {
161         printf("1) Realizar la operaci%cn\n2) Salir\n",ao);
162         printf("Elige una opci%cn: ",ao);
163         scanf("%d",&op2);
164         switch(op2)
165         {
166             case 1:
167                 printf("%cDe qu%c n%cmmero se calculara la factorial?",sp,ae,au);
168                 scanf("%d",&n);

```

Conclusion

Gracias a la practica se pudo crear un menu de salida que imprime lo que se requiere para la situacion, con el ejercicio se creo opciones para el usuario si requiere hacer una cierta operación, salir de hacer el tipo de operación o hacerla otra vez, o si se quiere realizar otro tipo de reaccion, con la practica se logro que todo esto se cumpliera con usando problemas aritmeticos basicos.

Referencias

- Nakayama Cervantes, A., Castañeda Perdomo, M., Solano Gálvez, J. A., García Cano, E. E., Sandoval Montaña, L., & Arteaga Ricci, T. I. (2018, 6 abril). Manual de prácticas del laboratorio de Fundamentos de programación (p.135-147). Laboratorio de Computación Salas A y B. http://odin.fib.unam.mx/salac/practicasFP/MADO-17_FP.pdf