



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: César Fabián Domínguez Velasco

Asignatura: Fundamentos de Programación

No. de práctica(s): 08

Integrante(s): 13_Garcia_Sanchez_Alejandro
18_Lopez_Castro_Anastacia
34_Ramirez_Rivas_Gael
39_Ruiz_Hernandez_Ruben_Antonio

No. de lista o brigada: 1A

Semestre: 2024-2

Fecha de entrega: 11 de Abril del 2024

Observaciones:

CALIFICACIÓN: _____

PRÁCTICA 08: ESTRUCTURAS DE REPETICIÓN

1) Objetivo:

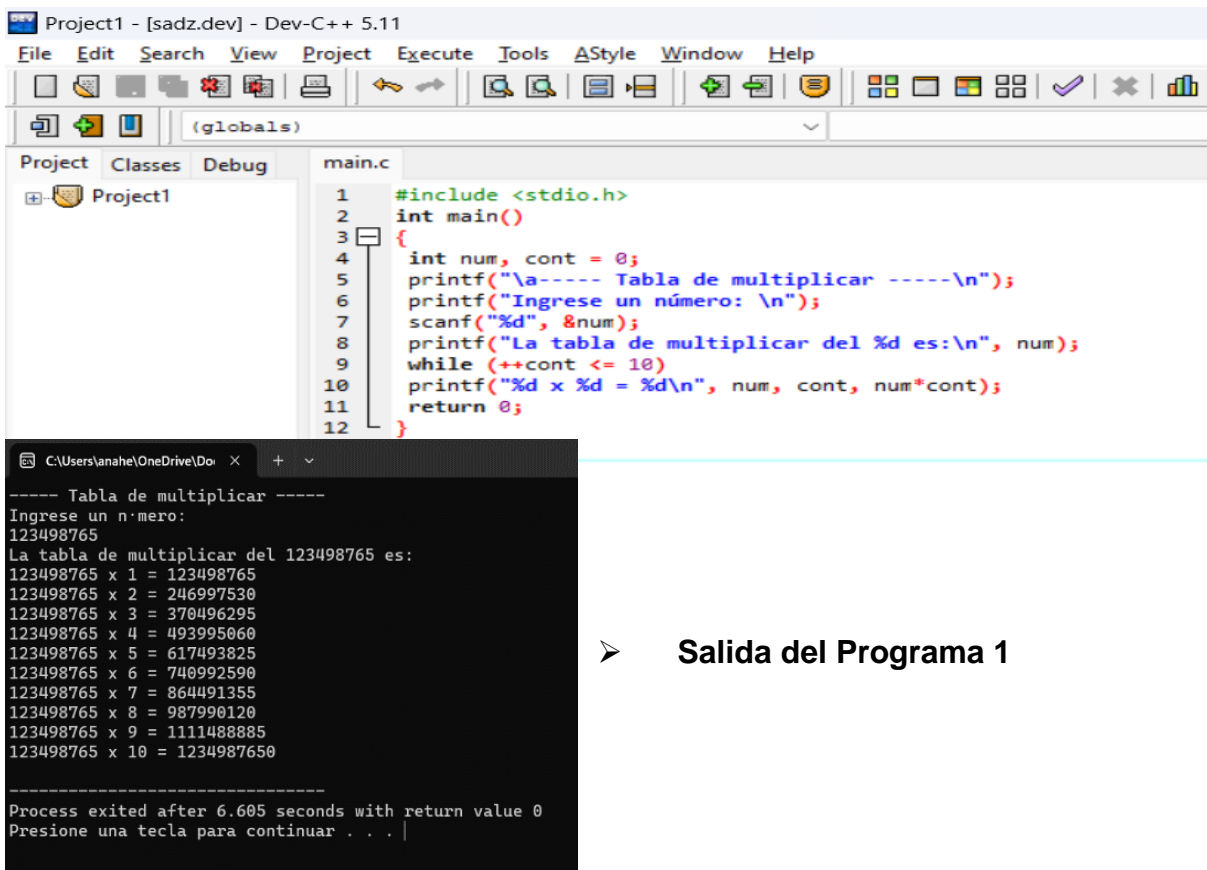
El alumno elaborará programas en C para la resolución de problemas básicos que incluyan las estructuras de repetición.

2) Introducción:

Las estructuras de repetición son las llamadas también estructuras cíclicas, iterativas o de bucles, las cuales permiten ejecutar un conjunto de instrucciones de manera repetida (o cíclica) mientras que la expresión lógica a evaluar se cumpla (sea verdadera), de igual manera en el lenguaje C existen tres estructuras de repetición: while, do - while y for.

3) Desarrollo (Capturas de pantalla de los programas en C):

Programa 1: Tablas de Multiplicar



```
Project1 - [sadz.dev] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
(globals)
Project Classes Debug main.c
Project1
1 #include <stdio.h>
2 int main()
3 {
4     int num, cont = 0;
5     printf("\a----- Tabla de multiplicar -----\\n");
6     printf("Ingrese un número: \\n");
7     scanf("%d", &num);
8     printf("La tabla de multiplicar del %d es:\\n", num);
9     while (++cont <= 10)
10     printf("%d x %d = %d\\n", num, cont, num*cont);
11     return 0;
12 }
```

```
C:\Users\anahe\OneDrive\Do... X + v
----- Tabla de multiplicar -----
Ingrese un número:
123498765
La tabla de multiplicar del 123498765 es:
123498765 x 1 = 123498765
123498765 x 2 = 246997530
123498765 x 3 = 370496295
123498765 x 4 = 493995060
123498765 x 5 = 617493825
123498765 x 6 = 740992590
123498765 x 7 = 864491355
123498765 x 8 = 987990120
123498765 x 9 = 1111488885
123498765 x 10 = 1234987650
-----
Process exited after 6.605 seconds with return value 0
Presione una tecla para continuar . . . |
```

➤ Salida del Programa 1

Programa 2: Ciclo Infinito

```
C programa1.c  C programa2.c •
C programa2.c
1  //Programa 2: Ciclo Infinito//
2
3  #include <stdio.h>
4
5  int main()
6
7  {
8
9  /* Al igual que en la estructura if-else, 0 -> falso y diferente de 0 -> verdadero.
10 El siguiente es un ciclo infinito porque la condición siempre es verdadera.
11 Así mismo, debido a que el ciclo consta de una sola línea, las llaves { } son
12 opcionales.*/
13
14 printf("Este programa permite conocer un ciclo infinito.\n");
15
16 while (100)
17 {
18
19 printf("Ciclo infinito.\nPara terminar el ciclo presione ctrl + c.\n");
20
21
22 }
23
24 return 0;
25
26 }
```

Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
^CPara terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.
Para terminar el ciclo presione ctrl + c.
Ciclo infinito.

→ Salida del Programa 2

Andorra04:Documents fp15alu39\$.

Programa 3: Promedio

```
1  #include <stdio.h>
2
3  int main ()
4
5  {
6  char op = 'n';
7  double sum = 0, calif = 0;
8  int veces = 0;
9  do
10 {
11 printf("\tSuma de calificaciones\n");
12 printf("Ingrese la calificación:\n");
13 scanf("%lf", &calif);
14 veces++;
15 sum = sum + calif;
16 printf("¿Desea sumar otra? S/N\n");
17 setbuf(stdin, NULL); // limpia el buffer del teclado
18 scanf("%c",&op);
19 getchar();
20 }
21 while (op == 'S' || op == 's');
22 printf("El promedio de las calificaciones ingresadas es: %lf\n", sum/veces);
23 return 0;
24 }
```

```

Suma de calificaciones
Ingrese la calificación:
10
¿Desea sumar otra? S/N
s

Suma de calificaciones
Ingrese la calificación:
5
¿Desea sumar otra? S/N
s

Suma de calificaciones
Ingrese la calificación:
5
¿Desea sumar otra? S/N
n
El promedio de las calificaciones ingresadas es: 6.666667

```

Programa 4: Calculadora

```

main.c
1 //Programa 4: Calculadora//
2
3 #include <stdio.h>
4
5 int main ()
6
7 {
8
9     printf("Este programa permite realizar cualquiera de las 4 operaciones básicas de una calculadora:\n");
10
11     int op, uno, dos;
12
13     do
14     {
15         printf(" --- Calculadora ---\n");
16         printf("\n¿Qué desea hacer?\n");
17         printf("\n1) Sumar\n");
18         printf("\n2) Restar\n");
19         printf("\n3) Multiplicar\n");
20         printf("\n4) Dividir\n");
21         printf("\n5) Salir\n");
22         scanf("%d", &op);
23
24         switch(op)
25         {
26             case 1:
27                 printf("\nSumar\n");
28                 printf("\nIntroduzca los números a sumar separados por comas\n");
29                 scanf("%d, %d", &uno, &dos);
30                 printf("%d + %d = %d\n", uno, dos, (uno + dos));
31                 break;
32             case 2:
33                 printf("\nRestar\n");
34                 printf("\nIntroduzca los números a restar separados por comas\n");
35                 scanf("%d, %d", &uno, &dos);
36                 printf("%d - %d = %d\n", uno, dos, (uno - dos));
37                 break;
38             //case 3:
39             //case 4:
40             //case 5:
41             //break;
42         }
43     } while (op != 5);
44 }

```

```

main.c
34 scanf("%d", &dos, &uno, &dos);
35 printf("%d + %d = %d\n", uno, dos, (uno + dos));
36
37 break;
38
39 case 2:
40
41 printf("\nRestar\n");
42 printf("\nIntroduzca los números a restar separados por comas\n");
43 scanf("%d", &dos, &uno, &dos);
44 printf("%d - %d = %d\n", uno, dos, (uno - dos));
45
46 break;
47
48 case 3:
49
50 printf("\nMultiplicar\n");
51 printf("\nIntroduzca los números a multiplicar separados por comas\n");
52 scanf("%d", &dos, &uno, &dos);
53 printf("%d * %d = %d\n", uno, dos, (uno * dos));
54
55 break;
56
57 case 4:
58
59 printf("\nDividir\n");
60 printf("\nIntroduzca los números a dividir separados por comas\n");
61 scanf("%d", &dos, &uno, &dos);
62 printf("%d / %d = %.2f\n", uno, dos, ((double)uno / dos));
63
64 break;
65
66 case 5:
67
68 printf("\nSalir\n");
69
70 break;
71
72 default:
73
74 printf("\nOpción inválida.\n");
75
76 }
77 }
78
79 while (op != 5);
80
81 }

```

```

Este programa permite realizar cualquiera de las 4 operaciones básicas de una calculadora:

--- Calculadora ---

¿Qué desea hacer?

1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
1

Sumar

Introduzca los números a sumar separados por comas
45, 78
45 + 78 = 123

--- Calculadora ---

¿Qué desea hacer?

1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
2

Restar

Introduzca los números a restar separados por comas
100, 777
100 - 777 = -677

--- Calculadora ---

¿Qué desea hacer?

1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
3

Multiplicar

```

```

¿Qué desea hacer?

1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
3

Multiplicar

Introduzca los números a multiplicar separados por comas
46, 22
46 * 22 = 1012

--- Calculadora ---

¿Qué desea hacer?

1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
4

Dividir

Introduzca los números a dividir separados por comas
2345, 5
2345 / 5 = 469.00

--- Calculadora ---

¿Qué desea hacer?

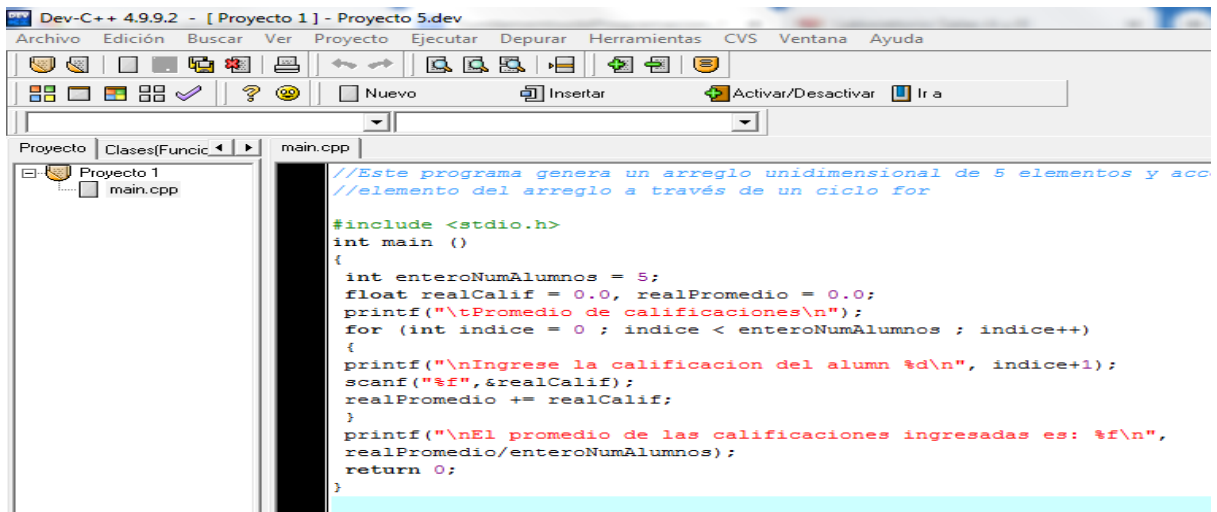
1) Sumar
2) Restar
3) Multiplicar
4) Dividir
5) Salir
5

Salir

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

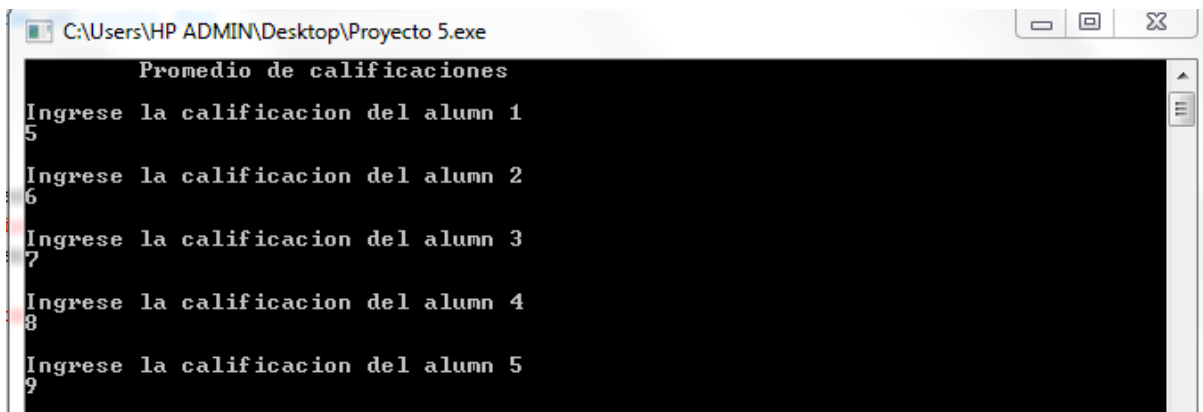
```

Programa 5: Calificaciones



```
//Este programa genera un arreglo unidimensional de 5 elementos y accede a cada elemento del arreglo a través de un ciclo for

#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 calificacion 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;
}
```



```
C:\Users\HP ADMIN\Desktop\Proyecto 5.exe

Promedio de calificaciones
Ingrese la calificacion del alumn 1
5
Ingrese la calificacion del alumn 2
6
Ingrese la calificacion del alumn 3
7
Ingrese la calificacion del alumn 4
8
Ingrese la calificacion del alumn 5
9
```

Ejercicio Adicional 1: Arreglo de Tamaño 10

main.c

```

1  /* Elabora un programa en C, donde declares un arreglo de enteros de tamaño 10
2  (declara una constante usando la directiva #DEFINE, tal como los últimos ejemplos en la práctica).
3  Mediante scanf, llena el arreglo de forma iterativa con números (enteros negativos y positivos).
4  Por último, imprime en reversa dicho arreglo, elevando al cuadrado cada uno de los valores de los
5  índices del arreglo principal*/
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 #define TAMANO 10
11
12 int arregloEnteros[TAMANO];
13 int i = 0;
14 int main()
15 {
16
17     printf("Este programa permite declarar un arreglo de tamaño 10.\n");
18
19     for (i = 0; i < TAMANO; i++)
20     {
21         printf("\nIngresa un número: ");
22         scanf("%d", &arregloEnteros[i]); //No se pone & porque es paso de datos o referencia
23     }
24
25     //Imprimir el arreglo para verificar que está lleno.
26
27     for (i = TAMANO - 1; i >= 0; i--)
28     {
29         printf("%d ", arregloEnteros[i] * arregloEnteros[i]);
30     }
31
32     return 0;
33 }

```

Console

↺

```

1  Este programa permite declarar un arreglo de tamaño 10.
2
3  Ingresa un número: -5
4  -5
5
6  Ingresa un número: -56
7  -56
8
9  Ingresa un número: 100
10 100
11 Ingresa un número: 20
12 20
13 Ingresa un número: 55
14 55
15 Ingresa un número: 43
16 43
17 Ingresa un número: -39
18 -39
19 Ingresa un número: -67
20 -67
21 Ingresa un número: -2
22 -2
23
24 Ingresa un número: -34
25 -34
26 4 4489 1521 1849 3025 400 10000 3136 25 25
27

```

Activar Windows
Ve a Configuración para activar Windows.

4) Conclusiones (Individuales):

 **García Sánchez Alejandro:**

La ejecución de esta práctica fue muy importante y necesaria, ya que, me permitió conocer las diferentes estructuras de control, así como su uso, en qué casos podemos aplicarlas para una mejor optimización del programa y presentación. Al realizar el programa 5 empleando la estructura de repetición “for” me permitió capturar una serie de datos determinados, sin estar solicitando los continuamente, simplemente haciendo uso de una iteración con un fin determinado.

 **López Castro Anastacia:**

Mejore significativamente mi comprensión y destreza en el uso de las estructuras de repetición en lenguaje C. A través de la práctica constante y la creación de programas, he consolidado mi dominio de las estructuras iterativas como while, do-while y for. Además, en esta práctica al ser menos los programas a realizar, es más fácil centrarse en cada uno de ellos.

 **Ramírez Rivas Gael:**

En esta práctica se realizaron actividades las cuales consistieron en realizar programas en lenguaje C basados en el uso de estructuras de repetición con el fin de aprender a utilizar las funciones iterativas para realizar una tarea determinada. Las actividades se realizaron sin contratiempos utilizando las distintas estructuras de repetición según el caso cumpliendo los objetivos planteados al inicio.

 **Ruíz Hernández Rubén Antonio:**

Logré perfeccionar, es decir comprender el uso de las estructuras de repetición mediante el empleo de construcción de programas del lenguaje C de medio nivel, sin embargo, por esta ocasión el profe auxiliar que nos impartió la práctica compartió programas

con dichas estructuras iterativas (while, do - while y for), creando videojuegos fantásticos, de los cuales nos pueden servir de apoyo a corto plazo para nuestro proyecto final de semestre.

5) Retroalimentación (Equipo):

Se cumplieron al pie de la letra los objetivos de esta práctica, ya que claramente cada uno tenía su propio camino, es decir, para el caso de la estructura “while” nos permite crear un ciclo puesto que al ejecutar el programa determina su simple una condición, del mismo modo reforzamos que para el “for” se utiliza cuando se indica el número de veces que debe tener un ciclo. Adicionalmente, el profesor de la asignatura planteó un problema donde nuestro supervisor nos dio una pista de elaborar un programa en C, pues declaramos un arreglo de enteros de tamaño 10, ya que realizamos el uso de la directiva “#DEFINE” y mediante un “scanf”, que llenó el arreglo de forma iterativa con números (enteros negativos y positivos), de igual manera, como salida imprimimos en reversa dicho arreglo, elevando al cuadrado cada uno de los valores de los índices del arreglo principal.

6) Fuentes en APA:

- ❖ Laboratorio Salas A y B. (s.f.). *Manual de Prácticas de la Asignatura Fundamentos de Programación (Guía práctica de estudio 08: Estructuras de Repetición, pág. 128 - 136)*. Recuperado el 05 de abril del 2024, de Laboratorio de Computación Salas A y B: <http://lcp02.fi-b.unam.mx/>
- ❖ RuyAntonio. (s.f.). *GitHub - RuyAntonio/practica8_fdp: Práctica 08: Estructuras de Repetición*, de GitHub. Recuperado el 05 de abril del 2024, de GitHub: https://github.com/RuyAntonio/practica8_fdp.git