



Carátula para entrega de prácticas

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): 08

Integrante(s): Juan Manuel Cuellar Orbezo

No. de lista o brigada: 07

Semestre: 2024 - 2

Fecha de entrega: 09 Abril 2024

Observaciones:

CALIFICACIÓN: _____

Práctica 08 - Estructuras de repetición

Objetivo:

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

Estructuras de repetición: Resumen

Esta tabla resume las estructuras de repetición básicas (for, while, do while) en programación, pseudocódigo y código fuente. Permiten ejecutar una sección de código una n cantidad de veces.

Estructura	Pseudocódigo	Codificación
for	Para (i desde 1 hasta x con paso y) Instrucciones fin para	for (i = 0; i < x; i++) { } }
while	Mientras (exp lógica) Instrucciones fin mientras	while (exp log) { } }
do while	Hacer Instrucción Repetir mientras	do { } while (exp log)

Programa 1

Este código imprime la tabla de multiplicar de un número ingresado por el usuario. Esto se logra con el loop while, siempre y cuando nuestro contador sea menor o igual a 10.

Si se cambia en este caso ++cont por cont++ en el código, el programa imprimirá la tabla de multiplicar de manera incorrecta. ++cont incrementa el valor de cont antes de usarlo en la expresión. cont++ incrementa el valor de cont después de usarlo en la expresión.

```
#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 <= 10)
    printf("%d x %d = %d\n", num, cont, num*cont); return 0;
```

```
}
```

Programa 2

Este código permite calcular el promedio de varias calificaciones.

Solicita al usuario que ingrese calificaciones una por una. Acumula las calificaciones y cuenta la cantidad de calificaciones ingresadas, finaliza cuando el usuario ingresa una letra diferente a 'S' o 's'. Calcula y muestra el promedio de las calificaciones ingresadas.

```
#include <stdio.h>
int main ()
{
    char op = 'n';
    double sum = 0, calif = 0;
    int veces = 0;
    do
    {
        printf("\tSuma de calificaciones\n\n");
        printf("Ingrese la calificación:\n");
        scanf("%lf", &calif); //Pide un numero al usuario

        veces++;
        sum = sum + calif;

        printf("¿Desea sumar otra? S/N\n");
        setbuf(stdin, NULL); // limpia el buffer del teclado
        // Es una manera de limpiar "la memoria del programa para poder
        volver a usar scanf"
        scanf("%c",&op);
        getchar();
    }
    while (op == 'S' || op == 's');
    printf("El promedio de las %d califs. es:
    %.11f\n",veces,sum/veces);
    return 0;
}
```

Programa 3

Este código permite realizar operaciones básicas aritméticas (suma, resta, multiplicación y división) entre dos números ingresados por el usuario.

Presenta un menú con las opciones disponibles, se selecciona la operación que desea realizar. Solicita al usuario que ingrese dos números separados por comas. Realiza la

operación y muestra el resultado. Repite el proceso hasta que el usuario selecciona la opción "Salir".

```
#include <stdio.h>
int main ()
{
    int op, uno, dos;
    do //se agrega aquí el do porque es el principio de la parte del
    programa que queremos que se ejecute antes de revisar la condición
    {

        printf("\n--- 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\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); //agregamos aquí while porque es la parte donde acaba
el "loop" que queremos que se ejecute por lo menos una vez, cabe
resaltar que while es la única estructura que debe tener ;
}

```

Programa 4

Este código permite calcular el promedio de las calificaciones de un grupo de alumnos. Se solicita la cantidad de alumnos, Permite ingresar las calificaciones de cada alumno de forma individual; calcula el promedio general de las calificaciones. Pregunta al usuario si desea calcular el promedio de otro grupo de alumnos y repite el proceso hasta que el usuario indica que no desea continuar.

El ciclo do-while se utiliza para repetir el proceso hasta que la variable opc cambia e indica que no se desea continuar.

```

#include <stdio.h>
int main ()
{
    int NoAl, opc;
    float Cal = 0.0, Prm = 0.0, Glb = 0.0;

    printf("\t***Promedio de calificaciones***");

    do
    {
        printf("\n\nHola profesor! Cuántos alumnos tiene?\n");
        scanf("%d",&NoAl);
        printf("\n");

        for (int idx = 0 ; idx < NoAl ; idx++)
        {
            printf("\nIngresa la calificación del alumn %d\n\t", idx+1);
            scanf("%f",&Cal);
            Prm += Cal;
        }
    }
}

```

```

    }

    Glb = Prm/NoAl;
    printf("\nEl promedio de los %d alumnxs es: %.2f\n",NoAl, Glb);

    printf("\n\nProfesor, desea calcular otro grupo?");
    printf("\n    [Sí=1]/[No=0]\t");
    scanf("%d",&opc);

    Cal = 0.0, Prm = 0.0, Glb = 0.0;

}

while(opc==1);
}

```

Programa 5

Este código permite al usuario ingresar 5 valores enteros en un arreglo y luego los muestra uno a uno. El primer ciclo while recorre el arreglo y permite al usuario ingresar un valor para cada elemento. El segundo ciclo while recorre el arreglo y muestra el valor de cada elemento en la pantalla.

```

#include <stdio.h>
#define MAX 5
int main ()
{
    int arreglo[MAX], cont;
    cont = 0;

    while (cont<MAX)
    {
        printf("Ingrese el valor %d del arreglo: ", cont+1);
        scanf("%i", &arreglo[cont]);
        cont++;
    }

    printf("El valor ingresado para cada elemento del arreglo es:\n[");
    cont = 0;

    while (cont<MAX)
    {
        printf("\t%d", arreglo[cont]);
        cont++;
    }
}

```

```
printf("\t]\n");  
return 0;  
}
```

Tarea

Código A

Este código permite al usuario elegir entre tres tipos de bucles (for, while y do-while) para contar números del 1 al 10.

- Presenta un menú al usuario con las opciones disponibles.
- Permite al usuario seleccionar el tipo de bucle que desea utilizar.
- Ejecuta el bucle seleccionado, mostrando cada número del 1 al 10 en una nueva línea.
- Pregunta al usuario si desea volver a contar números, permitiéndole repetir el proceso.
- Se despide del usuario cuando este selecciona la opción "NO" para salir.

```
#include <stdio.h>  
int main ()  
{  
    int opc, con;  
    int exit = 1;  
  
    printf("\t***Contadores***");  
  
    do  
    {  
        printf("\n\nHola! Qué tipo de loop quieres usar?\n");  
        printf("\t[1] for\n\t[2] while\n\t[3] do while\n");  
        scanf("%d",&opc);  
        printf("\n");  
  
        con = 1;  
  
        switch (opc)  
        {  
            case 1:  
                printf("***Ciclo FOR***\n");  
  
                for (con=1; con<11; ++con)  
                {  
                    printf("\n\tNúmero %d",con);  
                }  
            }  
        }  
    }  
}
```

```

        printf("\n");
        break;

    case 2:
        printf("***Ciclo WHILE***\n");

        while (con<11)
        {
            printf("\n\tNúmero %d",con);
            ++con;
        }
        printf("\n");
        break;

        break;
    case 3:
        printf("***Ciclo DO WHILE***\n");

        do
        {
            printf("\n\tNúmero %d",con);
            ++con;
        }
        while (con<11);

        printf("\n");
        break;

        break;
}

printf("\n\nEso fue divertido! Quieres contar de nuevo?\n");
printf("\t[1] SI / [0] NO\t\t");
scanf("%d",&exit);

}

while(exit==1);

printf("\n\nAdios! :D");

}

```

Código B

Este código es prácticamente igual al programa 4. Sin embargo aquí hay un cambio de ciclo, de for a while. Nótese la importancia de declarar nuestro contador antes del ciclo y de incluir el incremento dentro de las instrucciones del mismo.


```

#include <stdio.h>
int main ()
{
    int NoAl, opc, idx;
    float Cal = 0.0, Prm = 0.0, Glb = 0.0;

    printf("\t***Promedio de calificaciones***");

    do
    {
        printf("\n\nHola profesor! Cuántos alumnos tiene?\t");
        scanf("%d",&NoAl);
        printf("\n");
        idx = 0;

        while (idx < NoAl)
        {
            printf("\nIngrese la calificación del alumn %d\n\t", idx+1);
            scanf("%f",&Cal);
            Prm += Cal;
            idx++;
        }

        Glb = Prm/NoAl;
        printf("\nEl promedio de los %d alumnxs es: %.2f\n",NoAl, Glb);

        printf("\n\nProfesor, desea calcular otro grupo?");
        printf("\n    [Sí=1]/[No=0]\t");
        scanf("%d",&opc);

        Cal = 0.0, Prm = 0.0, Glb = 0.0;

    }

    while(opc==1);
}

return 0;
}

```

GitHub

A continuación, la liga al repositorio de GitHub - https://github.com/CuellarJM/practica8_fdp

Conclusión

Esta práctica se enfocó en las estructuras iterativas en el lenguaje C los procesos iterativos son importantes puesto que ayudan a eficientar los algoritmos, crear mucho menos líneas de código y poder repetir unas instrucciones una n cantidad de veces. Todos los pasos que hemos estado repasando durante este mes de clases han sido aplicados para este tipo de ejercicios. El dominar los diferentes tipos de loops y entender cómo funcionan sus particularidades será muy importante para programas más avanzados en el curso.

Bibliografía

El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Edu

GitHub, Inc. (2024). Hello world. GitHub Docs.
<https://docs.github.com/en/get-started/start-your-journey/hello-world>

Gookin, D. (2004). *C for dummies ; 2nd edition*. John Wiley & Sons.

Solano, J. A., García, E. E., & Montaña, L. S. (2022, February 21). Manual de prácticas del laboratorio de Fundamentos de programación. CDMX; Facultad de Ingeniería.