



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Alejandro E. Pimentel Alarcón

Profesor:

Fundamentos de Programación

Asignatura:

3; Bloque 135

Grupo:

09

No de Práctica(s):

Bazaldúa Morales Vanessa

Integrante(s):

*No. de Equipo de
cómputo*

52-Tunez

06; 8166

No. de Lista o

2020-1

Semestre:

14/oct/19

Fecha de entrega:

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*.

Acerca de las estructuras de repetición:

Y a continuación se muestran algunas estructuras de repetición básicas, que ya hemos visto anteriormente pero que analizaremos nuevamente, sin embargo, ya no veremos tanta teoría, ahora será más directo a realizar el programa.

WHILE

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

DEFINE

El *define* es una palabra clave que se utiliza para declarar un nombre especial con un significado. Es muy parecido a una variable, con la diferencia de que no se puede cambiar a lo largo del programa.

```
#define MAX 5
```

DO-WHILE

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

FOR

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

Actividades:

- ◆ Hacer un programa que pida un número y muestre su tabla de multiplicar (hasta el 10).

```
~/Escritorio/tablas.c - Sublim  
File Edit Selection Find View Goto Too  
numero.c tablas.c  
1 #include <stdio.h>  
2  
3 int main(){  
4     int a,c,r;  
5  
6     scanf("%i",&a);  
7  
8     c=10;  
9     do{  
10        r=a*c;  
11  
12        printf("%i x %i = %i\n",a,c,r);  
13        c=c-1;  
14    }  
15    while (c!=0);  
16    return 0;  
17 }
```

```
Archivo Editar Ver Buscar Terminal Ayuda  
vanessa@Titan:~$ cd Escritorio  
vanessa@Titan:~/Escritorio$ gcc tablas.c -o tablas  
vanessa@Titan:~/Escritorio$ ./tablas  
4  
4 x 10 = 40  
4 x 9 = 36  
4 x 8 = 32  
4 x 7 = 28  
4 x 6 = 24  
4 x 5 = 20  
4 x 4 = 16  
4 x 3 = 12  
4 x 2 = 8  
4 x 1 = 4  
vanessa@Titan:~/Escritorio$ ./tablas  
9  
9 x 10 = 90  
9 x 9 = 81  
9 x 8 = 72  
9 x 7 = 63  
9 x 6 = 54  
9 x 5 = 45  
9 x 4 = 36  
9 x 3 = 27  
9 x 2 = 18  
9 x 1 = 9
```

- ◆ Hacer un programa que pida y lea 10 números y muestre su suma y su promedio.

```
~/Escritorio/primo.c - Sublime Text (UNREGISTERED)  
File Edit Selection Find View Goto Tools Project Preferen  
numero.c tablas.c x promedio.c x  
1 #include <stdio.h>  
2 #include <stdlib.h>  
3  
4 int main(){  
5     int x=1,numero,contador=0;  
6     printf("Escribe un numero:\n");  
7     scanf("%i",&numero);  
8  
9     while(x<=numero){  
10        if(numero%x==0){  
11            contador++;  
12        }  
13        x++;  
14    }  
15  
16    if(contador>2){  
17        printf("El numero no es primo\n");  
18    }  
19    else{  
20        printf("El numero es primo\n");  
21    }  
22    return 0;  
23 }
```

```
vanessa@Titan: ~/Escritorio  
Archivo Editar Ver Buscar Terminal Ayuda  
vanessa@Titan:~$ cd Escritorio  
vanessa@Titan:~/Escritorio$ ./promedio  
bash: ./promedio: No existe el archivo o el directorio  
vanessa@Titan:~/Escritorio$ ls  
1 3 ensayorúbrica.docx primo promedio.c  
2 8 FP_2020-1_8166 primo.c 'tips para el ensayo.pptx'  
vanessa@Titan:~/Escritorio$ gcc promedio.c -o promedio  
vanessa@Titan:~/Escritorio$ ./promedio  
Escribe tus 10 numeros  
1 2 3 4 5 6 7 8 9 1  
El resultado de tu suma es: 46.00  
El promedio de la suma es: 4.60  
vanessa@Titan:~/Escritorio$ ./promedio  
Escribe tus 10 numeros  
2 5 7 9 6 4 6 8 2 7  
El resultado de tu suma es: 56.00  
El promedio de la suma es: 5.60  
vanessa@Titan:~/Escritorio$ ./promedio  
Escribe tus 10 numeros  
2.3 5.6 1.2 2 3.5 1.4 8 6.1 4.3 2.1  
El resultado de tu suma es: 36.50  
El promedio de la suma es: 3.65  
vanessa@Titan:~/Escritorio$
```

- ◆ Hacer un programa que pida un número e indique si es primo o no.

```
~/Escritorio/primo.c - Sublime Text (UNREGISTE
File Edit Selection Find View Goto Tools Project Prefe
numero.c tablas.c promedio.c
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     int x=1,numero,contador=0;
6     printf("Escribe un numero:\n");
7     scanf("%i",&numero);
8
9     while(x<=numero){
10         if(numero%x==0){
11             contador++;
12         }
13         x++;
14     }
15
16     if(contador>2){
17         printf("El numero no es primo\n");
18     }
19     else{
20         printf("El numero es primo\n");
21     }
22     return 0;
23 }
```

```
vanessa@Titan: ~/Escritorio
Archivo Editar Ver Buscar Terminal Ayuda
Escribe un numero:
3
El numero es primovanessa@Titan:~/Escritorio$ gcc primo.c -o primo
vanessa@Titan:~/Escritorio$ ./primo
Escribe un numero:
47
El numero es primo
vanessa@Titan:~/Escritorio$ ./primo
Escribe un numero:
50
El numero no es primo
vanessa@Titan:~/Escritorio$ ./primo
Escribe un numero:
13
El numero es primo
vanessa@Titan:~/Escritorio$ ./primo
Escribe un numero:
2
El numero es primo
vanessa@Titan:~/Escritorio$ ./primo
Escribe un numero:
8
El numero no es primo
vanessa@Titan:~/Escritorio$
```

Conclusión:

Para concluir, las estructuras de repetición nos ayudan a crear bucles o ciclos que se estén repitiendo, por supuesto, con ciertas condiciones para que se termine y nos de el resultado al que queremos llegar.

Son muy útiles, ya que, hacen más práctico el programa sin necesidad de escribir tanto. No obstante, debemos saber utilizarlos correctamente y saber en específico para que nos sirve cada uno, porque sino podemos cometer errores y el programa ya no funcionará.

Pero fuera de eso, estas estructuras nos ayudan demasiado, y creo que puede ser para varios programas.