

	
Facultad de Ingeniería	Laboratorio de docencia

Laboratorios de computación

**Profesor:** Saavedraa Hernández Honorato

**Asignatura:** Fundamentos de programación

**Grupo:** 01

**No de Práctica(s):** 9 “Estructuras de repetición”

**Integrante(s):** Luis Salinas Ludwig

**Semestre:** 2018-1

**Fecha de entrega:** 30/10/2017

**Observaciones:**

CALIFICACIÓN: \_\_\_\_\_

salas A y B

## Objetivo

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

## Desarrollo

Para el desarrollo de esta práctica usamos como herramienta a CodeBlocks, con ello comenzamos a ver las estructuras de control repetitivas que anteriormente fueron abordadas en pseudocódigo donde el nombre es distinto, pero en la forma de la sintaxis es algo parecida y comenzamos por:

### Estructura de control repetitivas while

Esta estructura lo que hace primero es validar la expresión lógica y si esta se cumple podremos ejecutar el bloque de instrucciones que se encuentra dentro de nuestra estructura (la cual ésta delimitada por llaves {}). De lo contrario continua su flujo normal sin realizar ninguna acción. La sintaxis de While es la siguiente:

while (expresión lógica)

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

Proseguimos con la Estructura de control repetitivas do-while.

Ésta es una estructura cíclica la cual ejecuta el bloque que se encuentra dentro de las llaves, donde evalúa lógicamente de una a ene veces hasta que termine su proceso. La 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);

Continuamos con la Estructura de repetición for.

Esta estructura ejecuta 3 acciones antes o después de ejecutar el bloque. La 1° es la inicialización, donde definimos las variables e inicializamos sus valores. La 2° es una expresión lógica la cual es evaluada y dependerá si se cumple o no para ejecutar el programa. La 3° consta de operaciones que se realizan ejecutarse el bloque. Hay que mencionar que las tres acciones son opcionales cada una de ellas. La sintaxis es la siguiente:

for (inicialización; expresión\_lógica; operaciones por iteración)

{

/\*

Bloque de código

a ejecutar

\*/

}

Define

Define permite definir constante o literales, las cuales son llamadas constantes simbólicas. La sintaxis es la siguiente:

#define<nombre><valor>

Break

Esta es la posibilidad para abandonar el ciclo, ya que esta proporciona una salida anticipada de la estructura de repetición. El break provoca que el ciclo termine inmediatamente.

Continue

Esta provoca que inicie la siguiente interacción del ciclo de repetición que la contiene.

Conclusiones

Las estructuras de repetición son de gran ayuda para poder desarrollar programas desde los muy básicos hasta más complicados ya que estos se ejecutan cuando necesitamos tener un ciclo repetitivo que al ejecutarse se evalúen expresiones lógicas, para así darle la solución al problema que resolverá nuestro programa. Estas estructuras de control repetitivas son un poco confusas desde mi punto de vista, pero pienso que, al entender mejor el funcionamiento de estas y como corren,

la comprensión será mejor y podremos seguir realizando programas con estas estructuras de control.