



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Solano Meza Angel Daniel

Matrícula: 372453

Maestro: Pedro Núñez Yépiz

Actividad No. : 6

Tema - Unidad : Ciclos y Validación de Entradas - Unidad 1

Ensenada Baja California a 19 de Septiembre del 2023



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

En esta practica aprendimos las diferencias al utilizar los diferentes tipos de ciclos, como utilizarlos de la forma mas optima y como poder trabajar con banderas cualquier ciclo. Tambien aprendimos a validar cualquier entrada que el usuario otorga mediante el uso de cadenas de texto, para que nuestro programa no crashee por error del usuario.

2. COMPETENCIA

Se identifican las diferencias entre los 3 ciclos de C.
Se utilizan las banderas de forma indirecta y directa.
Se logran validar todas las entradas del usuario.

3. FUNDAMENTOS

El ciclo **for** es al que más se recurre cuando se requiere realizar operaciones secuenciales, en donde se conoce el número de iteraciones o la condición a comprobar.

La sentencia de control **while** se encarga de repetir un bloque de código mientras se cumpla una condición. El bloque de código se debe de encontrar entre llaves, excepto si es una sola línea.

Cuando necesitamos que un ciclo se ejecute por lo menos una vez, es necesaria esta sentencia. **do-while**

Estos mismos se denominan ciclos determinados, mixtos e indeterminados .

Mas información aqui:

https://drive.google.com/file/d/1u7x-8WZm26xZsZLZGErrg8KZI0qOF_rE/view?usp=sharing



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

4. PROCEDIMIENTO

PARTE 1

- 1.- Programa que pregunte la cantidad de veces que desea que se realice el programa de fibonacci
- 2.- Programa que pida un número y desplegar la salida de factorial de un número dado
- 3.- Programa que pida un número y desplegar la cantidad de dígitos que tiene el número

PARTE 2

- 1.- El profesor de una materia desea conocer la cantidad de sus alumnos que no tienen derecho al examen de nivelación. Diseñe un programa en C que lea las calificaciones obtenidas en las 5 unidades por cada uno de los 40 alumnos y escriba la cantidad de ellos que no tienen derecho al examen de nivelación (Los promedios menores de 50 no tiene derecho a examen de nivelación)
- 2.- Realiza una función en C que imprima las tablas de multiplicar del 1 al 10
3. Realizar función en C para un programa que sirva para leer n cantidad de números dentro de un rango dado por el usuario, desplegar la suma de los números y la media aritmética de los números válidos dentro del rango.
- 4- En los cabos la embarcación finisterra que tiene fondo de cristal, solo sale a navegar con un máximo de 10 turistas o un máximo de 700 kilos de pasajeros. (preguntar el peso a cada turista) con un máximo de 15% de sobrepeso. desplegar el promedio de peso de los turistas y cual de los 2 condiciones se cumplió.
- 5.- Un alumno solo puede cursar la misma asignatura en un máximo de 3 veces, si el alumno reprueba durante sus intentos deberá repetir la materia, y si en su tercera ocasión no aprueba se le dará de baja académica. Elabore una función donde basada en sus 3 exámenes parciales calcular el promedio y basado en su promedio final, se deberá enviar mensaje al alumno de repetir materia, aprobado o baja temporal



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

5. RESULTADOS Y CONCLUSIONES

Las estructuras de control repetitivas y el uso de banderas nos permiten realizar una operación hasta que se cumplan una de varias condiciones que puedan detener el ciclo. Esto nos permitirá realizar procesos repetitivos de distintas formas. De igual forma el validar las entradas del usuario nos permitirá que nuestros programas ya no crasheen más por culpa de las entradas defectuosas del usuario.

```
while (band)
{
    printf("TURISTA %d: \n", ++turistas);
    peso = Validar("Ingresa tu peso: \n", 0, 805);
    ac += peso;
    if (peso >= 805) //Si se llega al limite de peso sale
    {
        band = 0;
    }
    if (turistas == 10) //Si se llega a los 10 turistas sale
    {
        band = 0;
    }
}
```

6. ANEXOS

Archivo anexo.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

7. REFERENCIAS

Diseño de algoritmos y su codificación en lenguaje C

Corona, M.A. y Ancona, M.A. (2011)..

España: McGraw-Hill.

ISBN: 9786071505712

Programación estructurada a fondo: implementación de algoritmos en C

:Pearson Educación.Sznajdleder, P. A. (2017)..

Buenos Aires,Argentina: Alfaomega

Como programar en C/C++

H.M. Deitel/ P.J. Deitel

Segunda edición

Editorial: Prentice Hall.

ISBN:9688804711

Programación en C.Metodología, estructura de datos y objetos

Joyanes, L. y Zahonero, I. (2001)..

España:McGraw-Hill.

ISBN: 8448130138