



UAEM

Universidad Autónoma
del Estado de México

Unidad 3. Estructuras de control en diagrama de Flujo y pseudocódigo Tema Iteración o Repetitivas (mientras)



Juan Pablo Cobá Juárez Pegueros

Programación Básica

Bioingeniería Médica

Facultad de Medicina

05/09/2017

Contenido



- ¿Qué son las estructuras de control repetitivas ?
- ¿Cómo se ejecuta?
- ¿Cómo estructura la condición de repetición ?
- Tipos de estructuras de repetición
- Ciclo Mientras
- Diagrama de Flujo y Pseudo código
- Componentes
- Variable de control
- Condición
- Incremento
- Ejemplo
- Evaluación de la condición
- Pseudocódigo
- Encontrar los factores de un número
- ¿Cuál es el factor de un número?
- Lógica para encontrar todos los factores de un número
- Pseudocódigo
- Calcular la tabla de multiplicar de un numero
- Lógica para calcular la tabla de multiplicar de un numero
- Pseudocódigo
- Bibliografía



Objetivo: Aplicar las estructuras de control computacionales utilizando instrucciones sintáctica y semánticamente correctas para controlar el flujo de control en un programa computacional expresado en diagrama de flujo y/o pseudocódigo, utilizar la técnica de pruebas de escritorio para validar la propuesta de solución

Iteración o Repetitivas (mientras)

Unidad 3. Estructuras de control en diagrama de Flujo y pseudocódigo

¿Qué son las estructuras de control repetitivas ?



- Son estructuras que permiten la ejecución repetida de una secuencia de instrucciones que pueden ser:

Procesos

Impresión

decisión

lectura

¿Cómo se ejecuta?



- El número de veces que el bloque de instrucciones se ejecutará se puede especificar de manera **explícita** o a través de **una condición** que indica cuando se ejecuta de nuevo o cuando no

¿Cómo se ejecuta?



- **explícita** nosotros conocemos el valor inicial y el valor final del ciclo
- **condición** el valor que regrese la condición determina el inicio o fin del ciclo



Existen tres tipos principales de sentencias de repetición

- **Ciclo MIENTRAS**
- **Ciclo REPITE-MIENTRAS**
- **Ciclo PARA**



- El CICLO MIENTRAS ejecuta un bloque de acciones 'mientras' que una condición dada se cumpla
- Si la condición no se cumple, el bloque no se ejecuta

Diagrama de Flujo y Pseudo código



Inicio

Diagrama de Flujo y Pseudo código



Diagrama de Flujo y Pseudo código

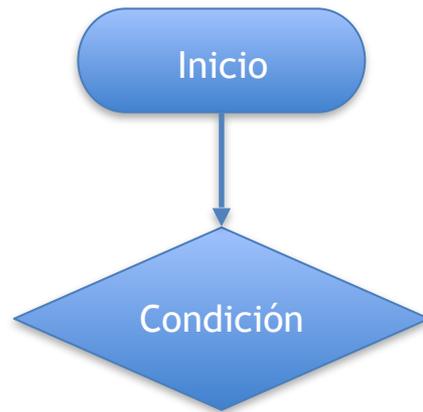


Diagrama de Flujo y Pseudo código

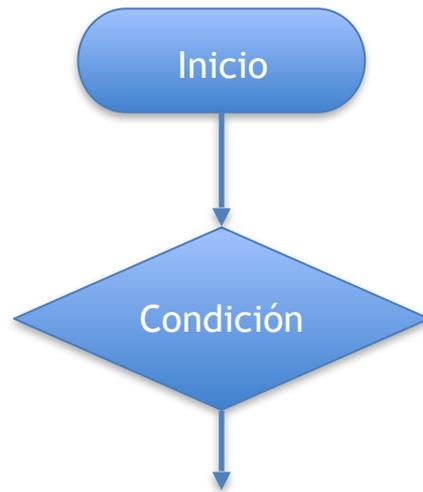


Diagrama de Flujo y Pseudo código

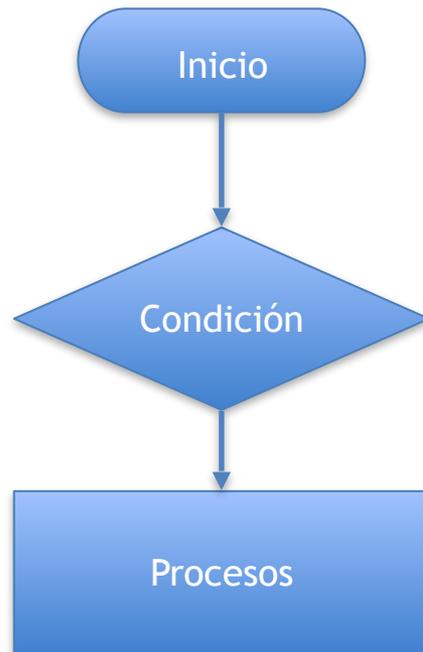


Diagrama de Flujo y Pseudo código

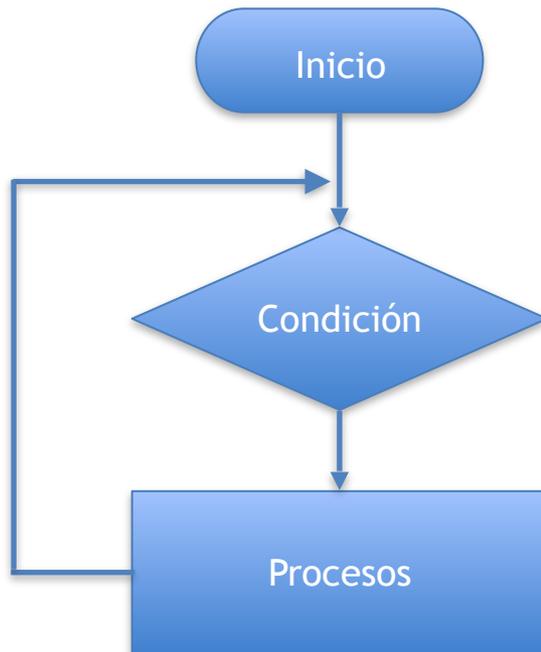


Diagrama de Flujo y Pseudo código

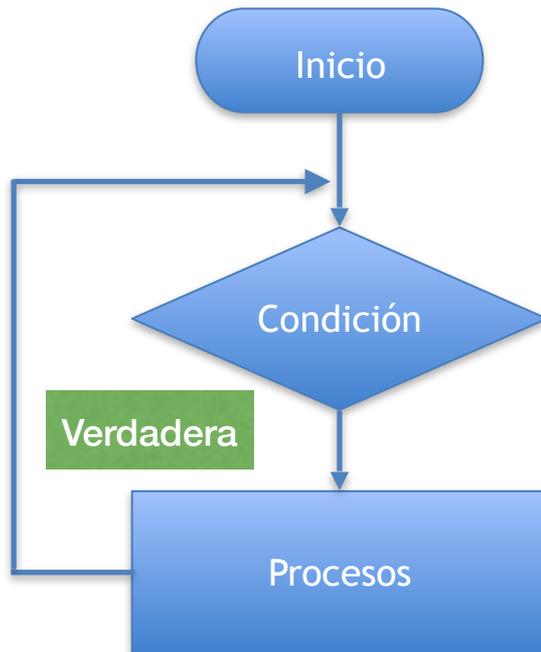


Diagrama de Flujo y Pseudo código

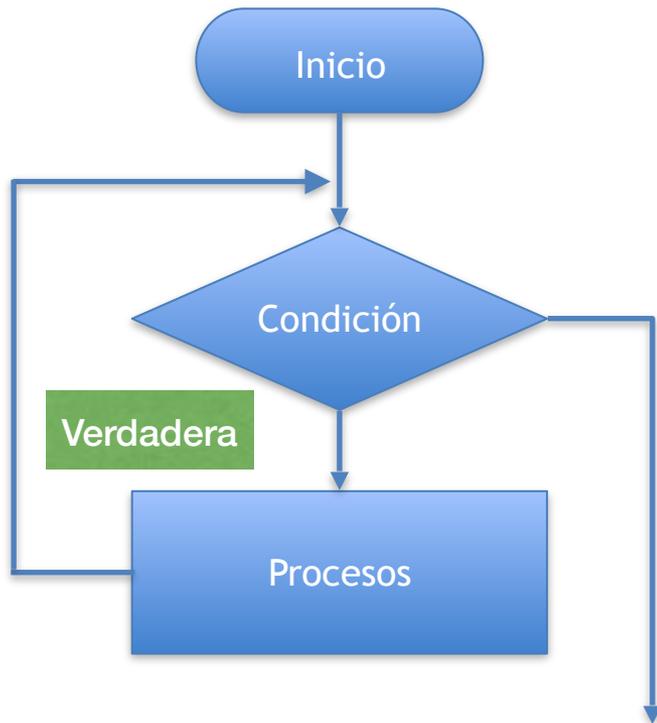


Diagrama de Flujo y Pseudo código

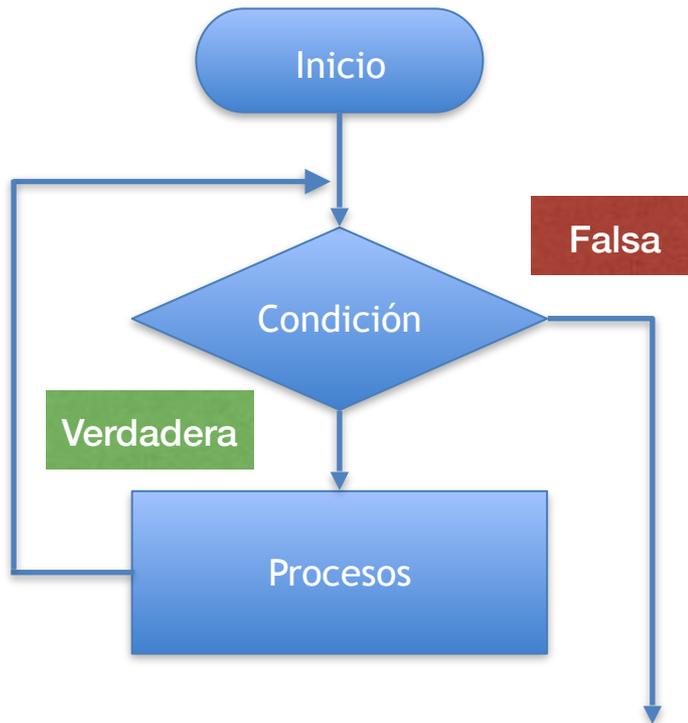


Diagrama de Flujo y Pseudo código



Diagrama de Flujo y Pseudo código



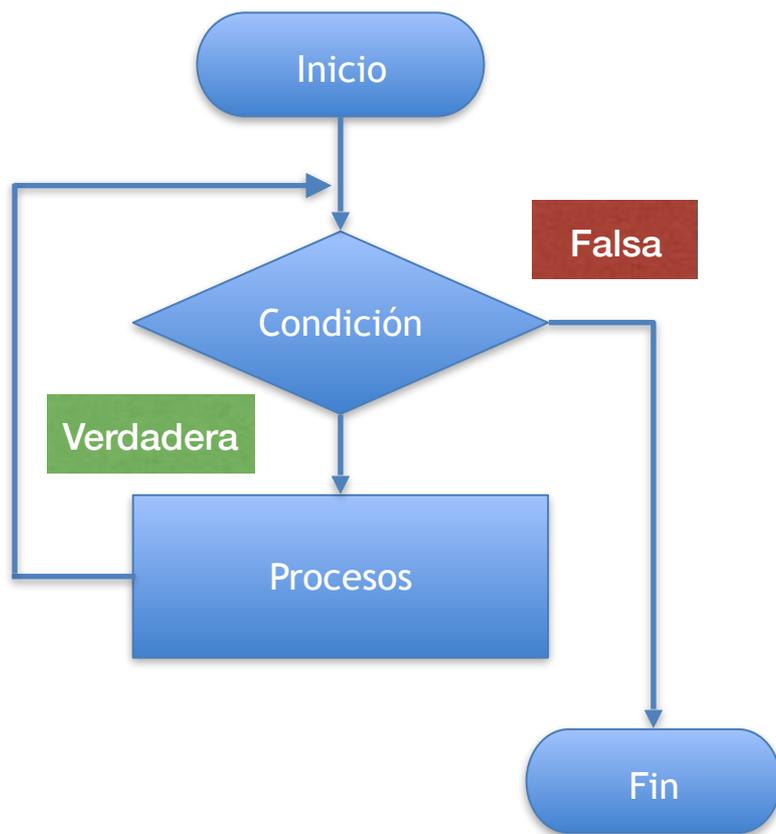


Mientras(condición)



Mientras(condición)

Bloque de
instrucciones



Mientras(condición)

Bloque de
instrucciones

Fin_mientras

Componentes



- De forma general un ciclo repetitivo mientras esta conformado de los siguientes componentes:



- De forma general un ciclo repetitivo mientras esta conformado de los siguientes componentes:
 - Variable de control de ciclo



- De forma general un ciclo repetitivo mientras esta conformado de los siguientes componentes:
 - Variable de control de ciclo
 - Condición



- De forma general un ciclo repetitivo mientras esta conformado de los siguientes componentes:
 - Variable de control de ciclo
 - Condición
 - Incremento

Variable de control



- Variable de control se utiliza para “controlar” el flujo de control del programa
- Puede visualizarse como un contador que puede incrementar o decrementar su valor

Condición



- Controla el momento en que se repite la estructura y el momento en que debe parar
- Su función es la de un interruptor que se encuentra encendido si la condición es evaluada como verdadera y se cierra de ser falsa.



- Su función es modificar el valor de la variable de control, lo que permite modificar el valor dentro de la condición en cada iteración.



- El problema es calcular la suma de los números naturales desde 1 hasta n .
- **Inicio**
- Entero $i \leftarrow 1$,
- Entero $n \leftarrow 0$, Suma $\leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$
- Suma = Suma + i
- $i = i + 1$
- **Fin mientras**
- Imprimir “el valor de la suma es ”, Suma
- **Fin**



- El problema es calcular la suma de los números naturales desde 1 hasta n .
- **Inicio**
- Entero $i \leftarrow 1$, ←
- Entero $n \leftarrow 0$, Suma $\leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$
- Suma = Suma + i
- $i = i + 1$
- **Fin mientras**
- Imprimir “el valor de la suma es ”, Suma
- **Fin**



- El problema es calcular la suma de los números naturales desde 1 hasta n .
- **Inicio**
- Entero $i \leftarrow 1$, ← Variable de control
- Entero $n \leftarrow 0$, $Suma \leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$
- $Suma = Suma + i$
- $i = i + 1$
- **Fin mientras**
- Imprimir “el valor de la suma es ”, $Suma$
- **Fin**



- El problema es calcular la suma de los números naturales desde 1 hasta n.
- **Inicio**
- Entero $i \leftarrow 1$, ← Variable de control
- Entero $n \leftarrow 0$, $Suma \leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$ ←
- $Suma = Suma + i$
- $i = i + 1$
- **Fin mientras**
- Imprimir “el valor de la suma es ”, Suma
- **Fin**



- El problema es calcular la suma de los números naturales desde 1 hasta n .
- **Inicio**
- Entero $i \leftarrow 1$, ← Variable de control
- Entero $n \leftarrow 0$, $\text{Suma} \leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$ ← Condición
- $\text{Suma} = \text{Suma} + i$
- $i = i + 1$
- **Fin mientras**
- Imprimir “el valor de la suma es ”, Suma
- **Fin**



- El problema es calcular la suma de los números naturales desde 1 hasta n .
- **Inicio**
- Entero $i \leftarrow 1$, ← Variable de control
- Entero $n \leftarrow 0$, $\text{Suma} \leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$ ← Condición
- $\text{Suma} = \text{Suma} + i$
- $i = i + 1$ ←
- **Fin mientras**
- Imprimir “el valor de la suma es ”, Suma
- **Fin**



- El problema es calcular la suma de los números naturales desde 1 hasta n .
- **Inicio**
- Entero $i \leftarrow 1$, ← Variable de control
- Entero $n \leftarrow 0$, $\text{Suma} \leftarrow 0$
- Leer n
- **MIENTRAS** $i \leq n$ ← Condición
- $\text{Suma} = \text{Suma} + i$
- $i = i + 1$ ← Incremento
- **Fin mientras**
- Imprimir “el valor de la suma es ”, Suma
- **Fin**

Ejemplo



- Crear un algoritmo que solicite un numero al usuario, el programa debe finalizar solo si el usuario introduce un número positivo.

Ejemplo



- Establecer la entrada de datos
- Se requiere leer un numero

Ejemplo

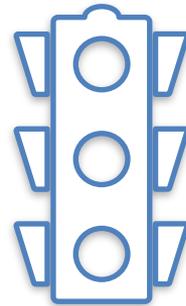


- Establecer la condición
- $(numero \leq 0)$

Ejemplo



- Establecer la condición
- ($\text{numero} \leq 0$)

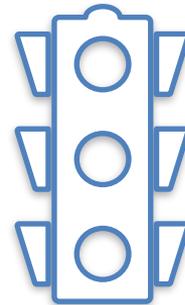


Ejemplo



- Establecer la condición
- $(\text{numero} \leq 0)$

Número	$\text{numero} \leq 0$	Evaluación
--------	------------------------	------------

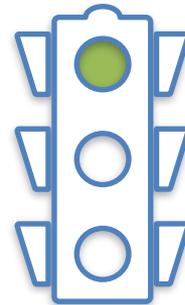


Ejemplo



- Establecer la condición
- $(numero \leq 0)$

Número	$numero \leq 0$	Evaluación
-1	$-1 \leq 0$	VERDADERO



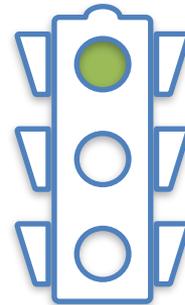
Se repite

Ejemplo



- Establecer la condición
- $(\text{numero} \leq 0)$

Número	$\text{numero} \leq 0$	Evaluación
-1	$-1 \leq 0$	VERDADERO
0	$0 \leq 0$	VERDADERO



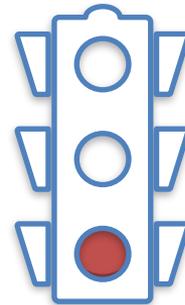
Se repite

Ejemplo



- Establecer la condición
- $(\text{numero} \leq 0)$

Número	$\text{numero} \leq 0$	Evaluación
-1	$-1 \leq 0$	VERDADERO
0	$0 \leq 0$	VERDADERO
10	$10 \leq 0$	FALSO



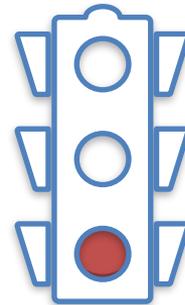
Se detiene

Ejemplo



- Establecer la condición
- $(\text{numero} \leq 0)$

Número	$\text{numero} \leq 0$	Evaluación
-1	$-1 \leq 0$	VERDADERO
0	$0 \leq 0$	VERDADERO
10	$10 \leq 0$	FALSO
4	$4 \leq 0$	FALSO



Se detiene

Ejemplo



- Establecer el bloque de instrucciones que se debe repetir
- Imprimir Error
- Imprimir “introduce un número positivo”
- Leer numero

Ejemplo



- Establecer el bloque de instrucciones que se debe repetir
- Imprimir Error
- Imprimir “introduce un número positivo”
- Leer numero



- **Inicio**
- Entero numero $\leftarrow 0$,
- Leer numero
- **MIENTRAS** numero ≤ 0
- Imprimir “error”
- Imprimir “escribe un numero entero ”
- Leer numero
- **Fin mientras**
- Imprimir “el valor es correcto ”
- **Fin**



UAEM | Universidad Autónoma
del Estado de México

Ejercicios



Encontrar los factores de un número



Escriba un algoritmo que permita ingresar un número del usuario y encuentre todos los factores del número dado usando el ciclo de repetición **Mientras** que permita encontrar factores de un número

¿Cuál es el factor de un número?



El factor de cualquier número es un número entero que divide exactamente el número en un número entero sin dejar ningún resto.

Por ejemplo: 2 es un factor de 6 porque 2 divide 6 exactamente dejando ningún residuo.

Lógica para encontrar todos los factores de un número



Paso a paso la lógica descriptiva para encontrar todos los factores de un número.

1. Introduzca el número del usuario. Guárdelo en la variable num
2. Ejecutar el ciclo mientras i sea menor o igual que num. La condición del ciclo debe ser similar a ($i \leq \text{num}$)
3. Para cada iteración dentro de ciclo compruebe si la variable de control del ciclo i es un factor de num o no. Para comprobar el factor, verificamos la divisibilidad del número realizando la división del módulo, es decir, si $(\text{num} \% i == 0)$ i es un factor de num.

Si i es un factor de num, entonces imprime el valor de i .

4. Incrementar i en 1
5. Establecer el fin del ciclo



INICIO

ENTERO $i \leftarrow -0$, num;

IMPRIMIR “introduce un numero”

LEER num

MIENTRAS($i \leq \text{num}$)

Si $\text{num} \% i = 0$

IMPRIMIR i

Fin si

$i \leftarrow i + 1$;

Fin MIENTRAS

FIN

Calcular la tabla de multiplicar de un numero



Escriba un algoritmo que permita introducir un número al usuario y calcule la tabla de multiplicación del número dado usando el ciclo **Mientras** .

Lógica para calcular la tabla de multiplicar de un número



1. Establecer i como la variable de control e iniciarla en 1
2. Inicializar otra variable para almacenar producto, es decir, $\text{producto} = 1$
3. Introduzca el número del usuario. Guárdelo en la variable num
4. Mientras la condición $(i \leq 10)$ sea verdadera Multiplique num por i y guarde el resultado en la variable producto
5. Imprime el valor de la variable producto .
6. Incrementa la variable de control $i \leftarrow i + 1$
7. Establecer el fin del ciclo



INICIO

ENTERO $i \leftarrow -1$, num, producto;

IMPRIMIR “introduce un numero”

LEER num

MIENTRAS($i \leq 10$)

 producto \leftarrow num * i

 IMPRIMIR producto

$i \leftarrow i + 1$;

FIN MIENTRAS

FIN



UAEM | Universidad Autónoma
del Estado de México

Gracias !!!



Bibliografía



- Cairo, O., (2008), Metodología de la programación: Algoritmos, Diagramas de Flujo y programas, México, Alfaomega.
Norton, P., (2006), Introducción a la computación, México, McGraw-Hill.
- Joyanes, A. L., (2008), Fundamentos de Programación. Algoritmos, Estructuras de Datos y Objetos, Madrid, McGraw-Hill.
Ramírez, F., (2007), Introducción a la programación: Algoritmos y su implementación en VB.NET, C#, JAVA Y C++, México, Alfaomega.
- Rodríguez, B. L., Fernández A. M. y Joyanes A. L., (2003), Fundamentos de programación. Libro de problemas. España, McGrawHill.
Criado, Ma. A., (2006), Programación en lenguajes estructurados, México, Alfaomega.
- López, R. L., (2004), Programación estructurada. Un enfoque algorítmico, México, Alfaomega.

ALGORITMOS Y PROGRAMACIÓN BÁSICA

Diapositivas

Material Didáctico sólo visión

Nombre del material:

Iteración o Repetitivas - mientras.

Contenido

Presentación.....	2
Objetivo de la asignatura	3
Guion explicativo para el uso del material.....	3
Orden de las diapositivas.....	3

Juan Pablo Cobá Juárez Pegueros
Facultad de Ingeniería
Bioingeniería Médica

Presentación

Hoy en día es de gran importancia en el mundo de la Bioingeniería Médica formar profesionales con un alto sentido de responsabilidad, de ética y vocación de servicio, y con las competencias y aprendizajes para diseñar y construir tecnología para solucionar problemas y necesidades específicas en los campos de intervención e investigación médica, libre de riesgos sanitarios

Por lo anterior los alumnos deben conocer, analizar y resolver mediante algoritmos y programación problemas asociados con los sistemas de la bioingeniería.

Objetivo general de la unidad de aprendizaje

Solucionar problemas mediante el desarrollo de algoritmos en pseudocódigo, verificando su funcionalidad mediante la prueba de escritorio.

Guion explicativo para el uso del material didáctico

Las unidades del programa son las siguientes:

UNIDAD 1. ARQUITECTURA DE COMPUTADORAS Y RESOLUCIÓN DE PROBLEMAS MEDIANTE ALGORITMOS
UNIDAD 2. TIPOS, OPERADORES Y EXPRESIONES
UNIDAD 3. ESTRUCTURAS DE CONTROL EN DIAGRAMA DE FLUJO Y PSEUDOCÓDIGO
UNIDAD 4. ARREGLOS
UNIDAD 5. INTRODUCCIÓN A LA PROGRAMACIÓN MODULAR

El material se encuentra enfocado exclusivamente a la Unidad 3, específicamente en el tema 3 Iteración o Repetitivas mientras.

Este material está dirigido a toda persona interesada en el tema, pero específicamente a los alumnos de la unidad de aprendizaje Algoritmos y Programación Básica.

El uso de este material es sencillo, ya que sólo contiene imágenes e ideas centrales del tema, que facilitan la concentración del alumno.

Orden de las diapositivas

1. ¿Qué son las estructuras de control repetitivas ?
2. ¿Cómo se ejecuta?
3. ¿Cómo estructura la condición de repetición ?
4. Tipos de estructuras de repetición
5. Ciclo Mientras
6. Diagrama de Flujo y Pseudo código
7. Componentes
8. Variable de control
9. Condición
10. Incremento
11. Ejemplo
12. Evaluación de la condición
13. Pseudocódigo
14. Encontrar los factores de un número
15. ¿Cuál es el factor de un número?
16. Lógica para encontrar todos los factores de un número
17. Pseudocódigo
18. Calcular la tabla de multiplicar de un numero
19. Lógica para calcular la tabla de multiplicar de un numero
20. Pseudocódigo
21. Bibliografía