



FUNDAMENTOS DE PROGRAMACIÓN

Unidad 4

Instructor: Luz Mery Cárcamo

ESTRUCTURAS CICLICAS

Conjunto de instrucciones que se repiten un numero finito de veces.

Cada repetición se llama una iteración.

Toda instrucción tiene asociada una condición que es la que va a determinar si se ejecuta la instrucción y hasta cuando.

Existen tres tipos de estructuras repetitivas, los cuales se utilizan de acuerdo a la situación que se presente. Estos son:

Mientras

Para

Haga

Contador: Variable cuyo valor inicial es 0 y durante el proceso bajo alguna condición se incrementa en 1.

Cont=0

cont=Cont+1

Acumulador: Es una variable que almacena cantidades de variables. Se utiliza para efectuar *sumas sucesivas*. Se inicializa en 0.

acum=0

acum=variable+acum

Interruptores , conmutadores bandera o switches, flag, indicadores

Conocidas como interruptores, switch, flags o conmutadores, son variables que pueden tomar solamente dos valores durante la ejecución del programa, los cuales pueden ser 0 ó 1, o bien los valores booleanos True o False. Se les suele llamar interruptores porque cuando toman los valores 0 ó 1 están simulando un interruptor abierto/cerrado o encendido/apagado. Puede tener las siguientes funciones:

- Salir de un ciclo de iteración variable.
- Comunicar información de una parte a otra parte del diagrama para variar la secuencia de la ejecución.

Por lo general la bandera inicia con el valor 0 y solamente se encenderá si la situación esperada se presenta.

Operadores de Incremento y Decremento

Ejemplo	Nombre	
++a	Prefijo de incremento	Incrementa a en 1, luego se usa el nuevo valor de a en la expresión en la que aparece
--a	Prefijo de decremento	Decrementa a en 1, luego se usa el nuevo valor de a en la expresión en la que aparece
a++	Postfijo de incremento	Usa el valor de actual de a en la expresión en la que a aparece, y luego incrementa a a en 1
a--	Postfijo de decremento	Usa el valor de actual de a en la expresión en la que a aparece, y luego decrementa a a en 1

Instrucción mientras

Forma general:

MIENTRAS (**condición**) Hacer

Instrucciones

Fin mientras

La **Condición** puede ser una expresión relacional o expresión lógica

Cuando la computadora encuentra la expresión mientras evalúa la condición. El resultado de evaluar la condición puede ser verdadero o falso.

Si el resultado es verdadero, se ejecutaran las instrucciones que están dentro del ciclo mientras, cuando termina de ejecutar las instrucciones y encuentra el delimitador fin del mientras, lo que le indica es que regrese a evaluar la condición y así se repite hasta que la condición sea falsa.

Ejemplo: Imprima los cinco primeros números naturales(1,2,3,4,5). Usando contador

INICIO

entero $n=0$

MIENTRAS ($n<5$) hacer

$n=n+1$

IMPRIMA n

Fin mientras

Fin

Imprima la suma de los cinco primeros números naturales(1,2,3,4,5). Usando acumulador

INICIO

entero n=0

real acum=0

MIENTRAS (n<5) hacer

n=n+1

acum=acum+n

Fin mientras

Imprima acum

Fin

- Ciclo infinito de números el cual al ingresar o, este ciclo se detendrá.
Imprimir e producto de todos los números

INICIO

real num, mul=1

int centi=0;

LEA “Ingrese el primer numero”, num

MIENTRAS (num<>centi) hacer

mul=num*mul

num=num+1

LEA “Ingrese el siguiente numero”, num

Fin mientras

IMPRIMA “El producto de los números es:”, mul

Fin

Instrucción Para

Forma general:

PARA (inicio; final; incremento)

 Instrucciones

Fin para

Ejemplo: Elaborar un algoritmo en el cual se da un entero n y se imprime todos los números enteros desde 1 hasta n

INICIO

 entero n,i

 LEA “Ingrese valor de n”, n

 For (i=1; i<=n;i++)

 IMPRIMA i

 Fin para

FIN

Instrucción Haga

Forma general:

HAGA

Conjunto de instrucciones que se ejecutan mientras la condición es verdadera

MIENTRAS Condición

muestre los números del 1 al 100

INICIO

entero $i=1$

IMPRIMA “NUMEROS DEL 1-100”

HAGA

IMPRIMA i

$i++$

MIENTRAS ($i \leq 100$)

FIN

TALLER

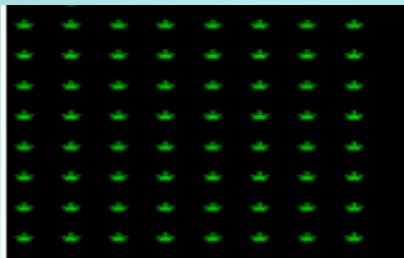
1. Imprima los números del 1 al 100 de 2 en dos
2. Calcular el promedio de notas definitivas de un grupo de estudiantes. Ingrese nombre, y definitiva. El programa se detiene cuando el usuario no desee ingresar mas registros.
3. Sumar los números entre el 3 y el 30
4. Imprima y sume la serie de números múltiplos de 3 en el rango cerrado de 3 a 99
5. Cálculo del factorial de un número realizado con los tres tipos de ciclos: Mientras, Haga y para:

6. Leer números y muestra su suma hasta que el usuario pulse S para terminar
7. Determinar si un numero es primo o no
8. Genere la tabla de multiplicar de un numero ingresado por el usuario ej: el usuario ingresa 2 debería quedar:
 $2 * 1$ hasta llegar a $2 * 10 =$
9. Genere las tablas de multiplicar de un número ingresado por el usuario así: si ingresa 2. se generan dos tablas la del 1 y la del 2 cada tabla hasta el 9
10. Realice un algoritmo que permita calcular y dar como salida el promedio general de una sección, tomando en cuenta que esta compuesta por 25 estudiantes y que se tiene la nota definitiva de cada uno de ellos.
11. Elaborar un algoritmo en el cual se da un entero n y se imprime todos los números enteros desde n hasta 1
12. Imprimir por pantalla 5 líneas de caracteres

13. Imprimir los 100 primeros números de Fibonacci
14. Diseñe un algoritmo que lea un entero n y calcule hasta el n-ésimo término de la sucesión de Fibonacci. Cada término de la sucesión se obtiene sumando los dos términos anteriores. Los dos primeros términos de la sucesión son el número uno. 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89,...

15. Por ejemplo, si $n = 6$ el algoritmo debe mostrar: 1, 1, 2, 3, 5, 8
16. Ingresa números del 1 al 100, hasta encontrar algún número mayor que 10.
17. Haga un algoritmo que imprima los términos de la siguiente serie:
 $\frac{1}{2}, \frac{3}{4}, \frac{5}{6}, \frac{7}{8}$
18. Leer cantidad indeterminada de números enteros positivos e imprima el promedio de todos los números. El algoritmo termina cuando se ingrese un número múltiplo de 6
19. Imprimir hola cada vez que un número sea menor a 5
20. Mostrar los números del 0-9 con haga mientras
21. Escriba un algoritmo que lea n números reales y muestre el promedio de los números leídos.
22. Escriba un algoritmo que lea n números reales y muestre el número que es menor y el número que es mayor.

23. Diseñe un algoritmo que lea un entero n y muestre un cuadrado formado por n filas con n asteriscos. Por ejemplo, si su algoritmo lee el número 8, este debe mostrar por pantalla el siguiente patrón de asteriscos:

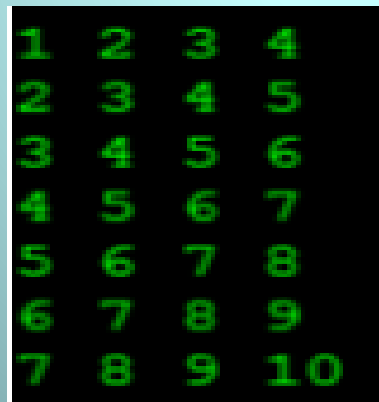


24. Diseñe un algoritmo que lea los nombres y las notas de n estudiantes y muestre: a) La cantidad de estudiantes ganaron. b) El porcentaje de estudiantes que perdieron. c) El nombre del estudiante con la nota más alta. d) La nota más alta. e) La nota más baja. f) La nota promedio de los estudiantes que ganaron. g) ¿Cuántos estudiantes obtuvieron la nota más alta? h) ¿Cuales estudiantes obtuvieron la nota más alta, nota mas alta 5?

25. Escriba un algoritmo para cada uno de los siguientes patrones de asteriscos. Por ejemplo, para el patrón de asteriscos B), lea un entero n y muestre n asteriscos en la primera fila, $n-1$ asteriscos en la segunda fila y así sucesivamente.



26. Diseñe un algoritmo que muestre la siguiente secuencia de números:



27. Diseñe un algoritmo que lea un número entero n y, luego, solicite los nombres, el sexo y las edades de n personas. Calcule y muestre: a) La cantidad de hombres b) La cantidad de mujeres c) El promedio de edad de las mujeres y el promedio de edad de los hombres d) El nombre y el sexo de la persona de mayor edad e) La mujer de menor edad f) El hombre de menor edad
28. Diseñe un algoritmo que lea las notas de n estudiantes y muestre cuantos estudiantes tienen una nota entre 3.5 y 4.5
29. Diseñe un algoritmo que lea números enteros hasta que encuentre uno que cumpla las siguientes condiciones: a. Múltiplo de 2 b. No múltiplo de 5 c. Mayor que 100 d. Menor que 10.000
30. Diseñe un algoritmo que muestre los números primos menores o iguales que un número leído por teclado.