



TALLER DE APLICACIÓN DISEÑO DE ALGORITMOS

EVIDENCIA: GA2-220501096-AA2

PRESENTADO POR:

**YASSER LEONARDO PACHECO CAÑIZARES
DURLEY SANDRITH GÁLVAN JIMENEZ**

CARLOS JOSÉ DELGADO GONZÁLEZ

PRESENTADO A:

EDUARDO SANCHEZ SANDOVAL

MILLERLANDY BECERRA CHAVEZ

**TECNICO EN PROGRAMACION DE APLICACIONES
PARA DISPOSITIVOS MOVILES**

FICHA: 2977832

SERVICIO NACIONAL DE APRENDIZAJE

2024



Descripción de actividad didáctica	
Nombre de la actividad	Taller de aplicación diseño de algoritmos
Objetivo de la actividad	Afianzar y aplicar los conceptos más importantes de la algoritmia a partir del desarrollo de ejercicios prácticos
Tipo de actividad sugerida	Desarrollar cada uno de los enunciados propuestos, siguiendo las estructuras de control de manera lógica
Archivo de entrega	Documento denominado ID_Ficha_PrimerNombre_PrimerApellido_Informe Tecnico Diseño de Algoritmos

Desarrollo de la actividad

Para cada uno de los siguientes problemas identifique el problema a resolver, los datos de entrada, los procesos a realizarse sobre dichos datos, los datos de salida y los resultados esperados en la solución de los problemas planteados.

A continuación, se muestra cómo se realizó un ejercicio para que sirva a manera de ejemplo para el desarrollo de los problemas propuestos.

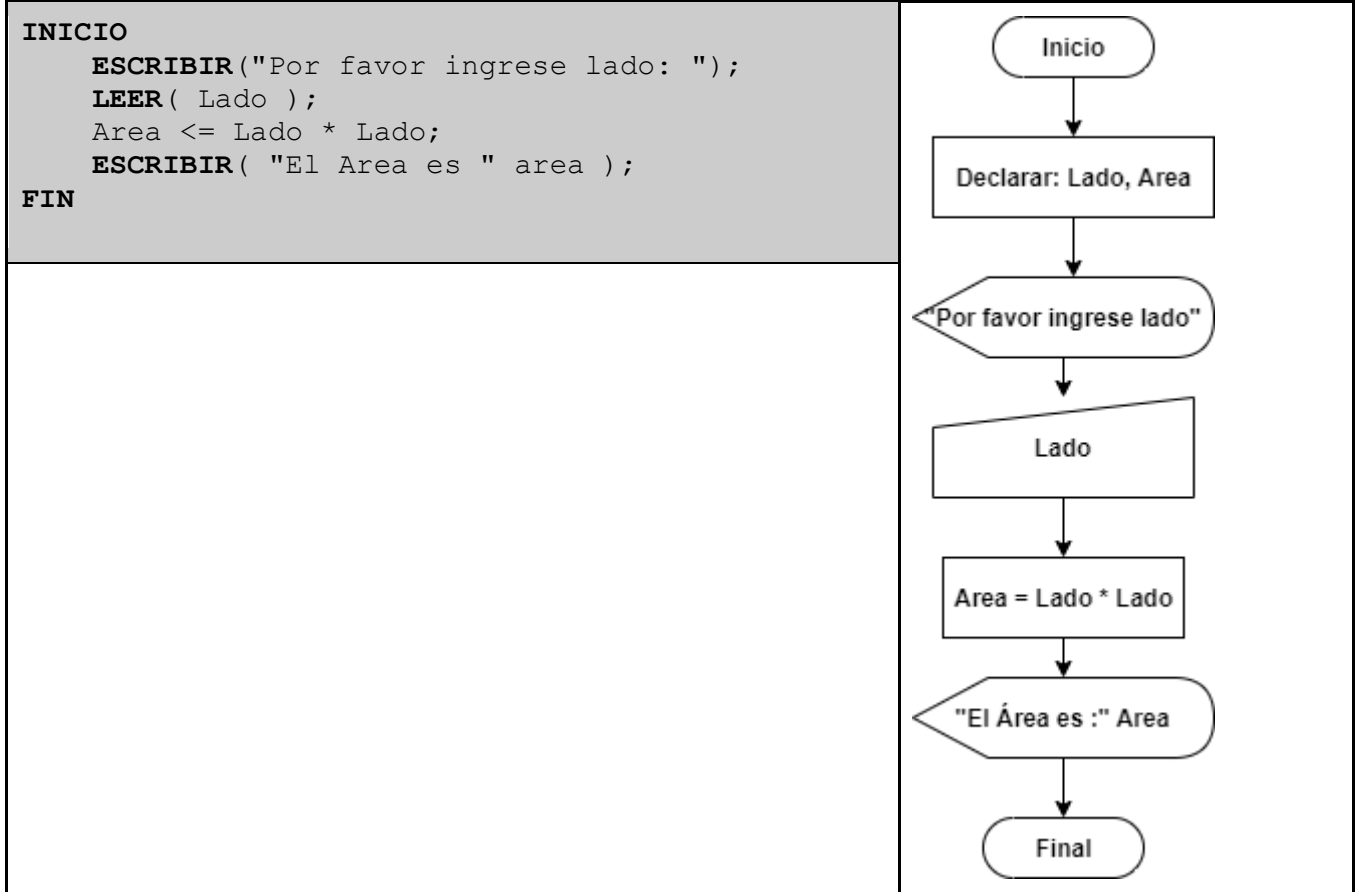
Ejemplo 1:

Hacer un algoritmo que permita el cálculo de un cuadrado y al final mostrar en la pantalla el valor del área de la figura geométrica.

Solución:

Datos		
Entrada		Identificadores
	Lado del cuadrado	Lado
Salida	Área	Área

```
ALGORITMO Calcular Area;  
VAR  
    REAL Lado;  
    REAL Area;
```



Ejemplo 2:

Hacer un algoritmo que pregunte si se desea calcular el área de un cuadrado, de un triángulo o un de un círculo y, realizar el cálculo solicitado y mostrarlo en pantalla.

Solución:

Datos		
Entrada	Tipo de área a calcular	Identificadores
	Lado del cuadrado	Opción
	Base del triángulo	Lado
	Altura del triángulo	Base
	Radio del círculo	Altura
Salida	Área	Radio
		Área



ALGORITMO Calcular Areas;

VAR

ENTERO opcion;

REAL lado;

REAL base;

REAL altura;

REAL radio;

REAL area;

INICIO

ESCRIBIR("1 - Área del Cuadrado.

2 - Área del Triángulo. 3 - Área del Círculo. Digite la opción: ");

LEER(opcion);

SI (opcion = 1)

ESCRIBIR("Escriba el lado del cuadrado ");

LEER(lado);

 area = lado * lado;

FINSI

SI (opcion = 2)

ESCRIBIR("Escriba la base del triángulo");

LEER(base);

ESCRIBIR("Escriba la altura del triángulo");

LEER(altura);

 area = base * altura / 2;

FINSI

SI (opcion = 3)

ESCRIBIR("Escriba el radio el círculo");

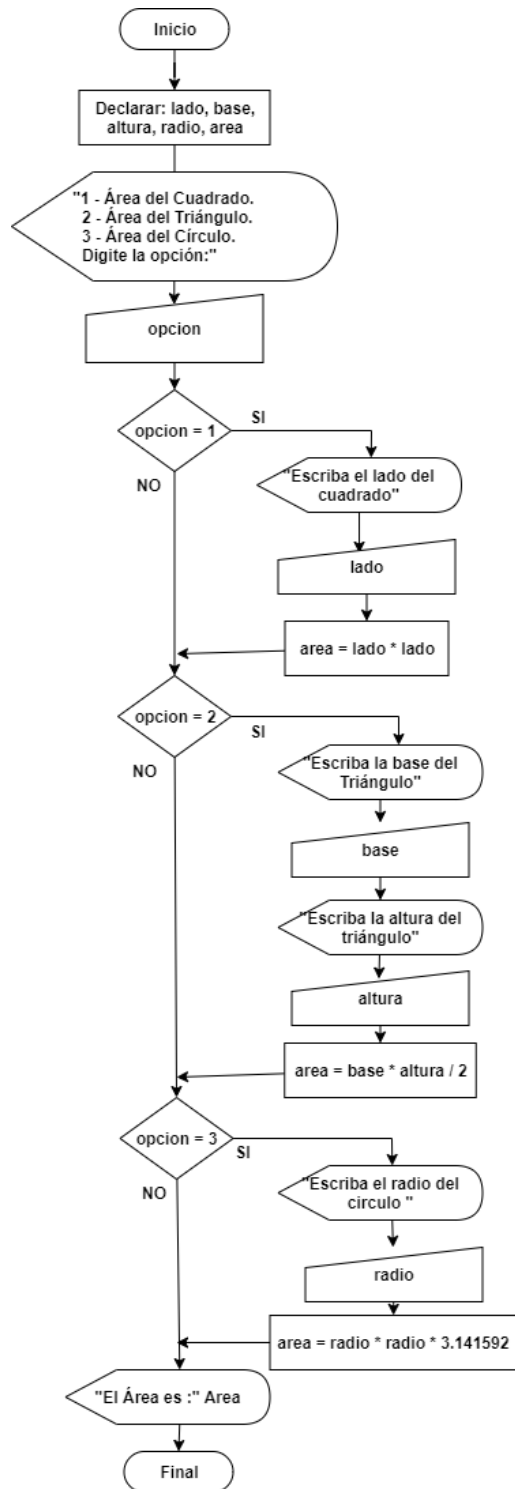
LEER(radio);

 area = radio * radio * 3.141592;

FINSI

ESCRIBIR("El área calculada es: " area);

FIN





Ejercicios a desarrollar

1. Diseñe un algoritmo en pseudocódigo y diagrama de flujo que lea tres números y, si el primero es positivo calcule el producto de los otros dos, y en otro caso, calcule la suma y muestre el resultado en pantalla.

Datos		Identificadores
Problema para resolver	Leer tres números y, si el primero es positivo calcule el producto de los otros dos, y en otro caso, calcule la suma y muestre el resultado en pantalla.	
Entrada	Numero 1 Numero 2 Numero 3 resultado	Numero 1 Numero 2 Numero 3 resultado
Salida	Producto de numero 2 y 3 o suma de números 2 +3 según la condición	Resultado

Algoritmo tres_numeros

definir numero1, numero2, numero3, resultado Como Real

ESCRIBIR ("Por favor ingrese número 1: ");

LEER numero1;

ESCRIBIR ("Por favor ingrese número 2: ");

LEER numero2;

ESCRIBIR("Por favor ingrese número 3: ");

LEER numero3;

Si (numero1 > 0)

 resultado<- numero2*numero3;

Finsi



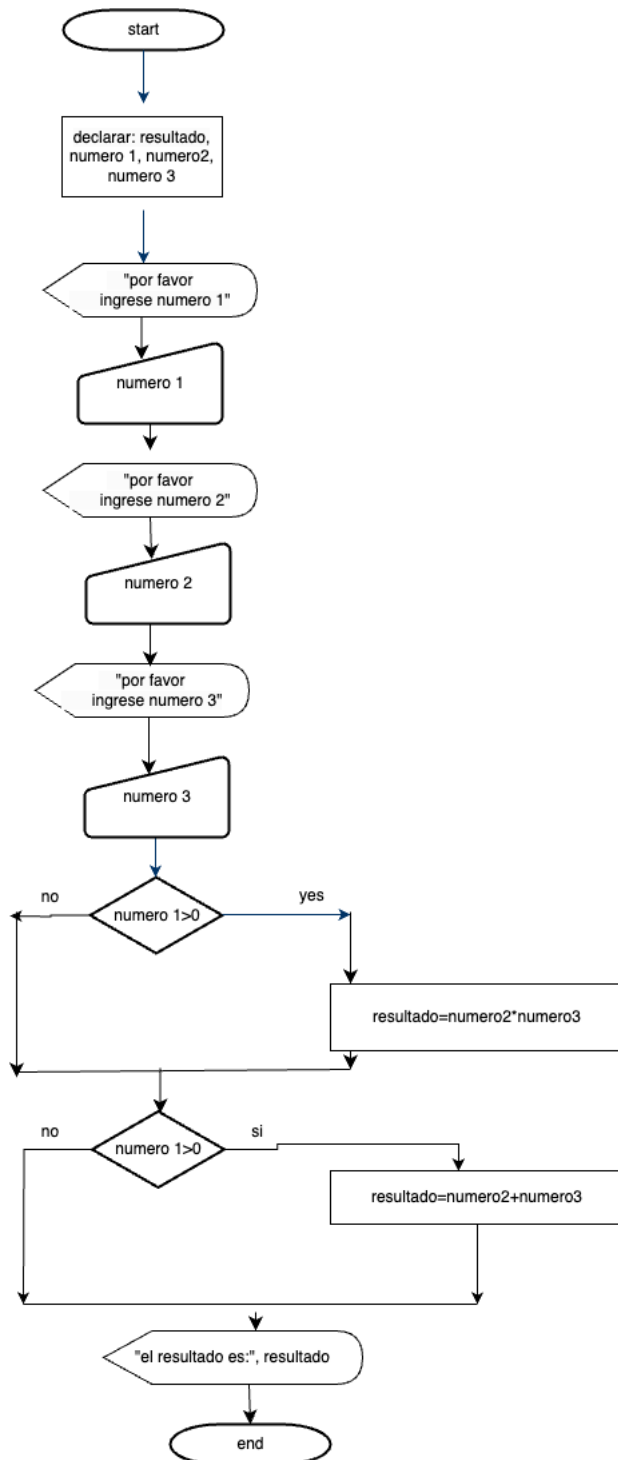
```
Si (numero1 <0)
```

```
    resultado<- numero2+numero3;
```

```
Finsi
```

```
    ESCRIBIR "el resultado es:", resultado;
```

```
FinAlgoritmo
```



2. Diseñe un algoritmo en pseudocódigo y diagrama de flujo que lea tres números enteros y decida si uno de ellos coincide con la suma de los otros dos.



Datos		Identificadores
Problema a resolver	Leer tres números enteros y decida si uno de ellos coincide con la suma de los otros dos.	
Entrada	Numero 1 Numero 2 Numero 3	Numero 1 Numero 2 Numero 3
Salida	String para decir si uno de ellos coincide con la suma de los otros dos o no	String

Algoritmo suma_tres_numeros

definir numero1,numero2,numero3 Como Real

ESCRIBIR ("Por favor ingrese número 1: ");

LEER numero1;

ESCRIBIR ("Por favor ingrese número 2: ");

LEER numero2;

ESCRIBIR("Por favor ingrese número 3: ");

LEER numero3;

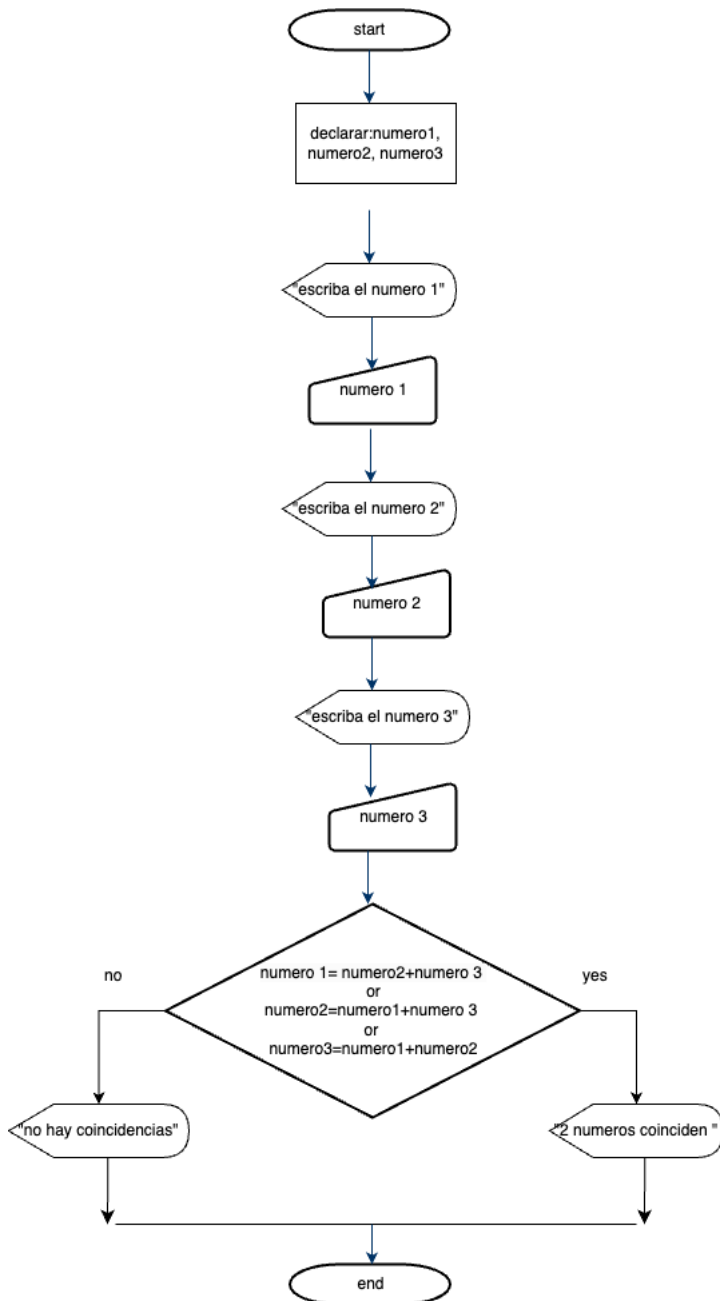
SI ((numero1=numero2+numero3)o (numero2=numero1+numero3) o
(numero3=numero1+numero2))

ESCRIBIR("2 numeros coinciden");

SINO

ESCRIBIR("no hay coincidencias")

FINSI



FinAlgoritmo



3. Diseñe un algoritmo que imprima y sume la serie de números múltiplos de 3 hasta 100, es decir, 3, 6, 9, 12, ... 99 (usar ciclos). Realizar la traza para las primeras cinco iteraciones.

Datos		Identificadores
Problema a resolver	imprimir y sumar la serie de números múltiplos de 3 hasta 100	
Entrada	Suma contador numero	Suma contador num
Salida	números múltiplos de 3 y su suma.	Suma números

Algoritmo múltiplos_de_tres

```
definir suma,contador,num Como Entero
suma<-0
contador<-0
num<-1
continuar <- Verdadero

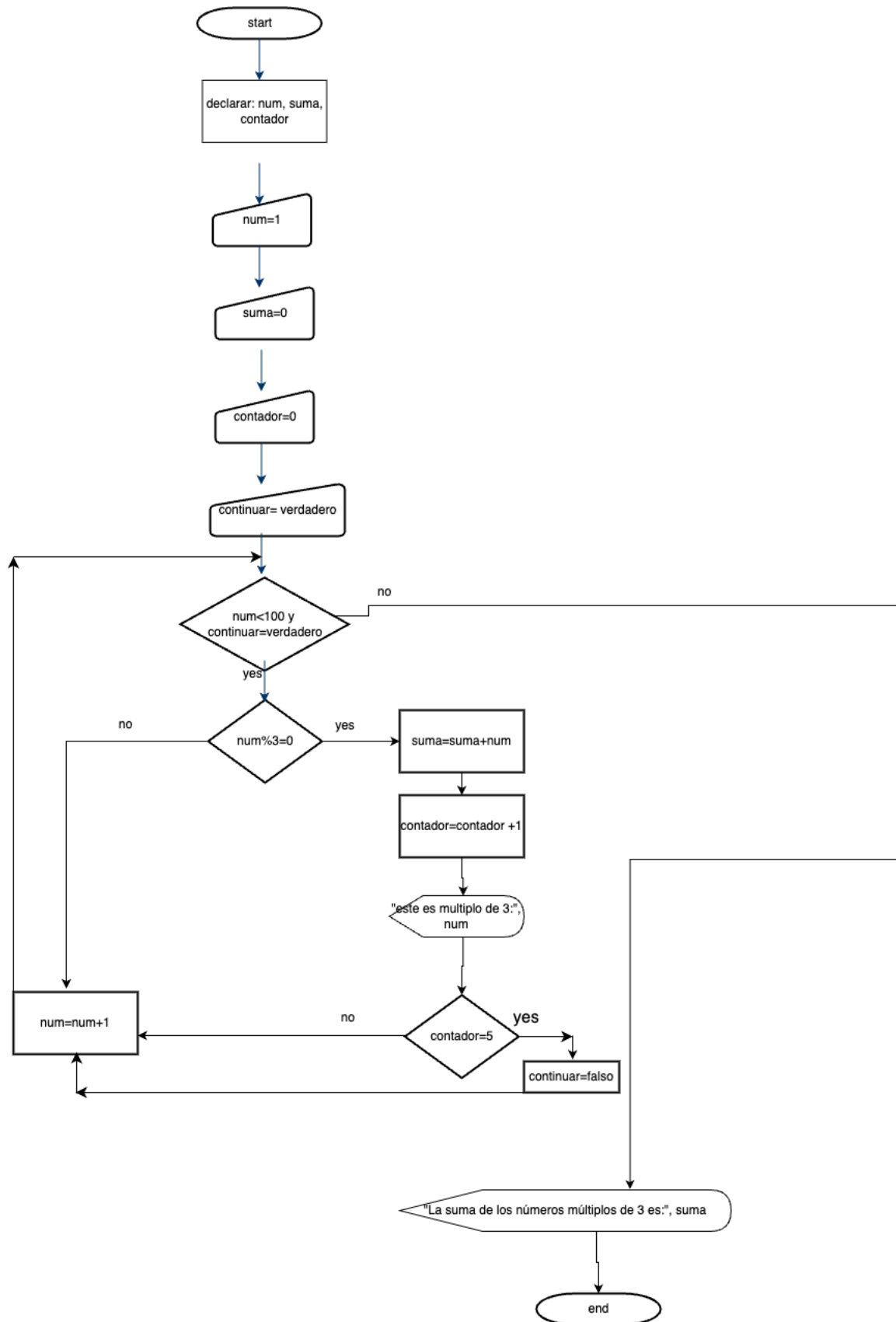
mientras (num<100) y continuar Hacer
    si num%3=0
        suma<-suma+num
        contador<-contador+1
        Escribir "este es multiplo de 3: ", num
        si contador=5 Entonces
            continuar <- Falso

        FinSi

    FinSi

    FinSi
    num<-num+1
FinMientras
escribir "la suma de los numero múltiplos de 3 es:",suma
```

FinAlgoritmo





4. Diseñe un algoritmo que presenta en pantalla todas las potencias enteras de 2 que sean menores o iguales que 100 (usar ciclos).

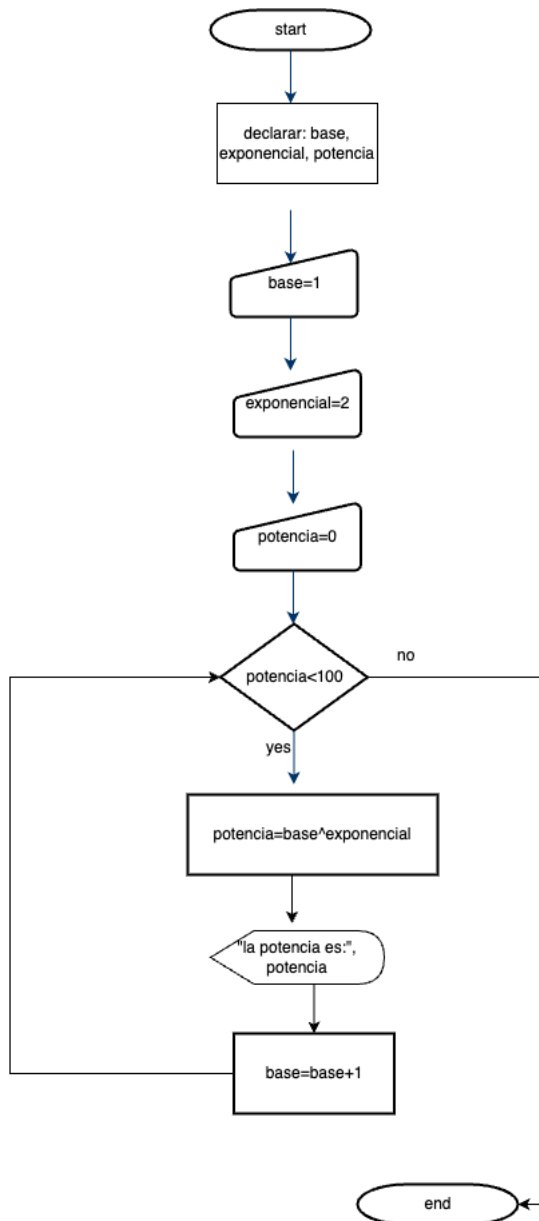
Datos		Identificadores
Problema a resolver	Presentar en pantalla todas las potencias enteras de 2 que sean menores o iguales que 100.	
Entrada	Base exponencial potencia	Base exponencial potencia
Salida	potencias enteras de 2 que sean menores o iguales que 100	String statement (con la lista de potencias)

Algoritmo potencias

```
definir base,exponencial,potencia Como Entero  
base<-1  
exponencial<-2  
potencia<-0
```

```
mientras (potencia<100)  
    potencia<-base^exponencial  
    escribir "la potencia es:",potencia  
    base<-base+1  
FinMientras
```

FinAlgoritmo





5. Diseñe un algoritmo que sume los números pares comprendidos entre 50 y 200, inclusive.

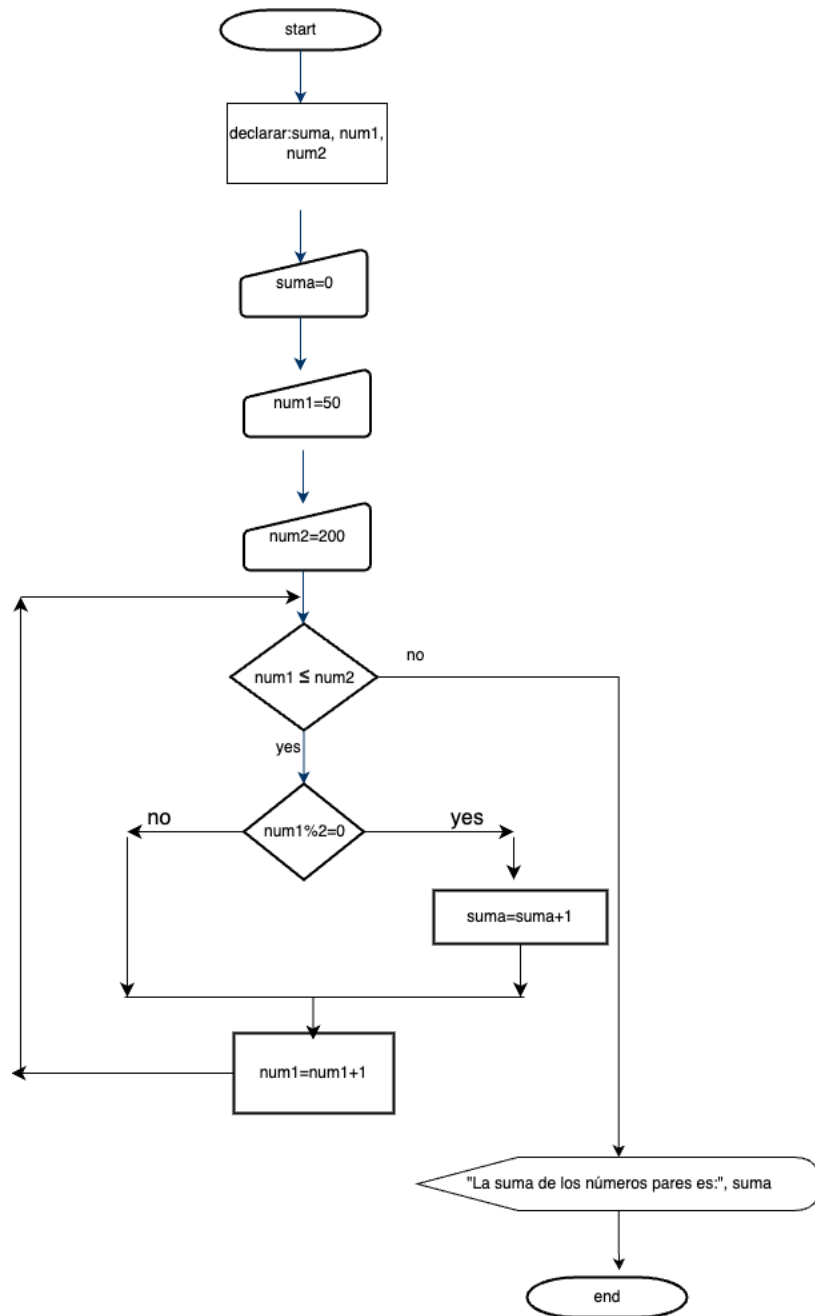
Datos		Identificadores
Problema a resolver	Sumar los números pares comprendidos entre 50 y 200.	
Entrada	Suma num1 sum2	Suma Num1 num2
Salida	Suma de números pares	suma

Algoritmo suma_pares

```
definir suma, num1, num2 Como Entero
suma<-0
num1<-50
num2<-200

mientras (num1<=num2)
    si num1%2=0
        suma<-suma+num1
    FinSi
    num1<-num1+1
FinMientras
Escribir "la suma de los numero pares es :",suma
```

FinAlgoritmo





6. Una temperatura Celsius (centígrados) puede ser convertida a una temperatura equivalente Fahrenheit, de acuerdo con la siguiente fórmula:

$$F = \left(\frac{9}{5}\right)C + 32$$

Diseñe un algoritmo que lea la temperatura en grados Celsius y la escriba en Fahrenheit.

Datos		Identificadores
Problema a resolver	Convertir una temperatura en grados Celsius a Fahrenheit.	
Entrada	Grados Celsius	Celsius
Salida	Temperatura en Fahrenheit	resultado

Algoritmo temperatura

definir celsius, resultado Como Real

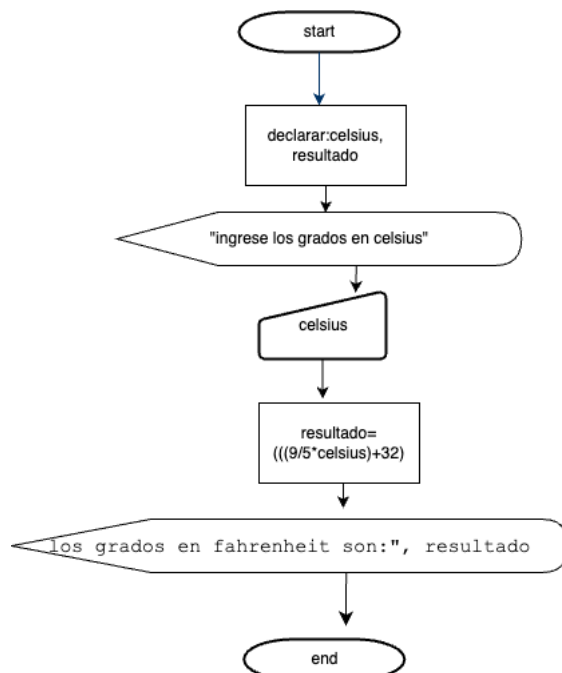
Escribir "ingrese los grados en celsius"

Leer celsius

resultado<- (((9/5)*celsius)+32)

escribir " los grados en fahrenheit son:", resultado

FinAlgoritmo





7. Diseñe un algoritmo que lea la hora de un día de notación de 24 horas y la respuesta en notación de 12 horas, por ejemplo, si la entrada es 13, la salida será 1 p.m.

Datos		Identificadores
Problema a resolver	Convertir la hora de notación de 24 horas a notación de 12 horas.	
Entrada	Hora en notación 24	hora
Salida	Hora en notación 12	hora

Algoritmo hora_

definir hora Como Entero

Escribir "ingrese la hora "

Leer hora

si hora<0 o hora >=24

Escribir "hora no valida, vuelva a intentar."

SiNo

si hora<12

si hora<-0

escribir "12 a.m"

SiNo

Escribir hora,"a.m"

FinSi

SiNo

si hora<-12

Escribir "12 p.m"

SiNo

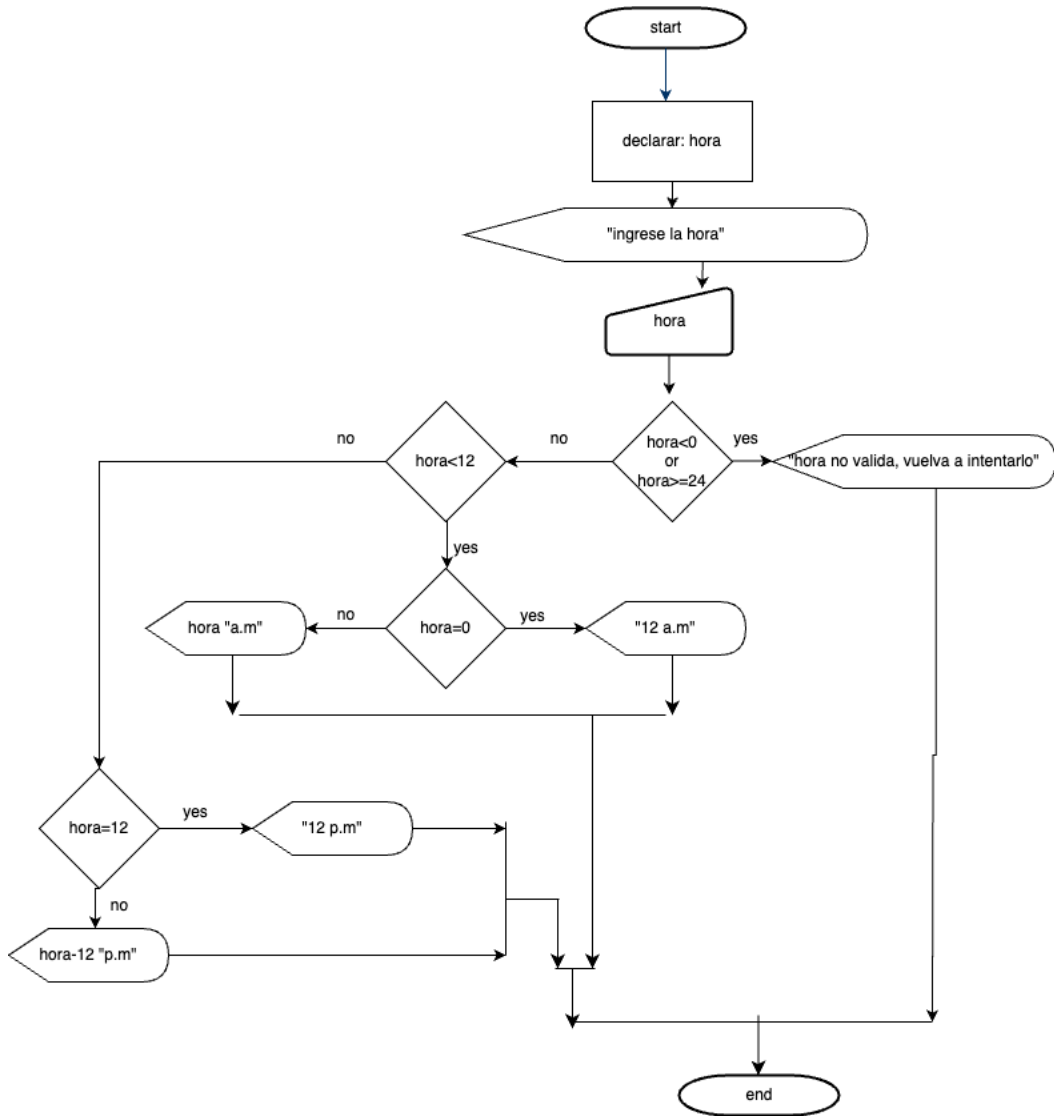
escribir hora-12, "p.m"

FinSi

FinSi

FinSi

FinAlgoritmo





8. Diseñe un algoritmo en pseudocódigo para crear un vector de cinco elementos de cadenas de caracteres, inicializa el vector con datos leídos por el teclado, copie los elementos del vector en otro vector, pero en orden inverso y, muéstrelo por la pantalla.

Datos		Identificadores
Problema a resolver	Crear un vector de cinco elementos de cadenas de caracteres, copiar los elementos del vector en otro vector en orden inverso y mostrarlo por pantalla.	
Entrada	5 elementos	vector
Salida	Vector invertido	Copia_vector

Algoritmo vector_

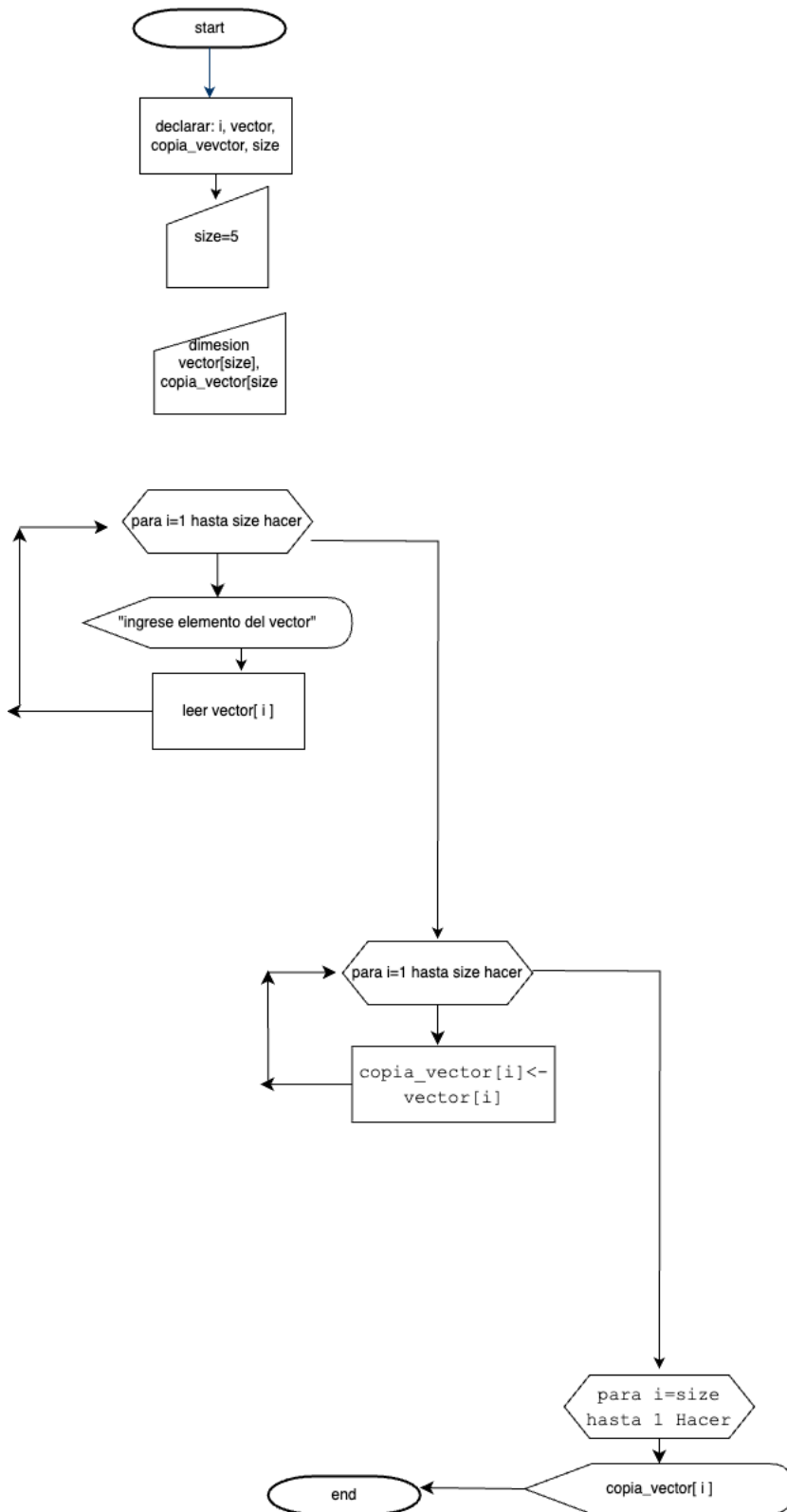
```
definir i, size Como entero
definir vector, copia_vector Como caracter
size<-5
Dimension vector[size], copia_vector[size];

para i <-1 hasta size Hacer
    Escribir "ingrese elemento del vector"
    leer vector[i]
FinPara

para i <-1 hasta size Hacer
    copia_vector[i]<-vector[i]
FinPara

para i<-size hasta 1 Hacer
    Escribir copia_vector[i]
FinPara
```

FinAlgoritmo





9. Diseñe un algoritmo que lea por el teclado las cinco notas obtenidas por un alumno (comprendidas entre 0 y 10). A continuación, debe mostrar todas las notas, la nota promedio, la nota más alta que ha sacado y la menor.

Datos		Identificadores
Problema a resolver	Leer cinco notas obtenidas por un alumno y mostrar todas las notas, el promedio, la nota más alta y la nota más baja.	
Entrada	5 notas	notas
Salida	Nota promedio todas las notas nota menor nota mayor	Promedio menor alta

Algoritmo notas_

 definir alta, menor, notas, contador Como entero
 definir promedio Como real

 contador<-1

 alta<- 0

 promedio<- 0

 menor<- 10

 mientras contador <=5 hacer

 escribir "ingrese la nota", contador

 leer notas

 si notas>0 y notas<=10 Entonces

 promedio<- (promedio+notas)

 si notas>alta Entonces

 alta<-notas

 FinSi

 si notas<menor Entonces

 menor<-notas

 FinSi

 contador<-contador+1

 SiNo

 Escribir "ingrese una nota entre 0 y

10"

 FinSi

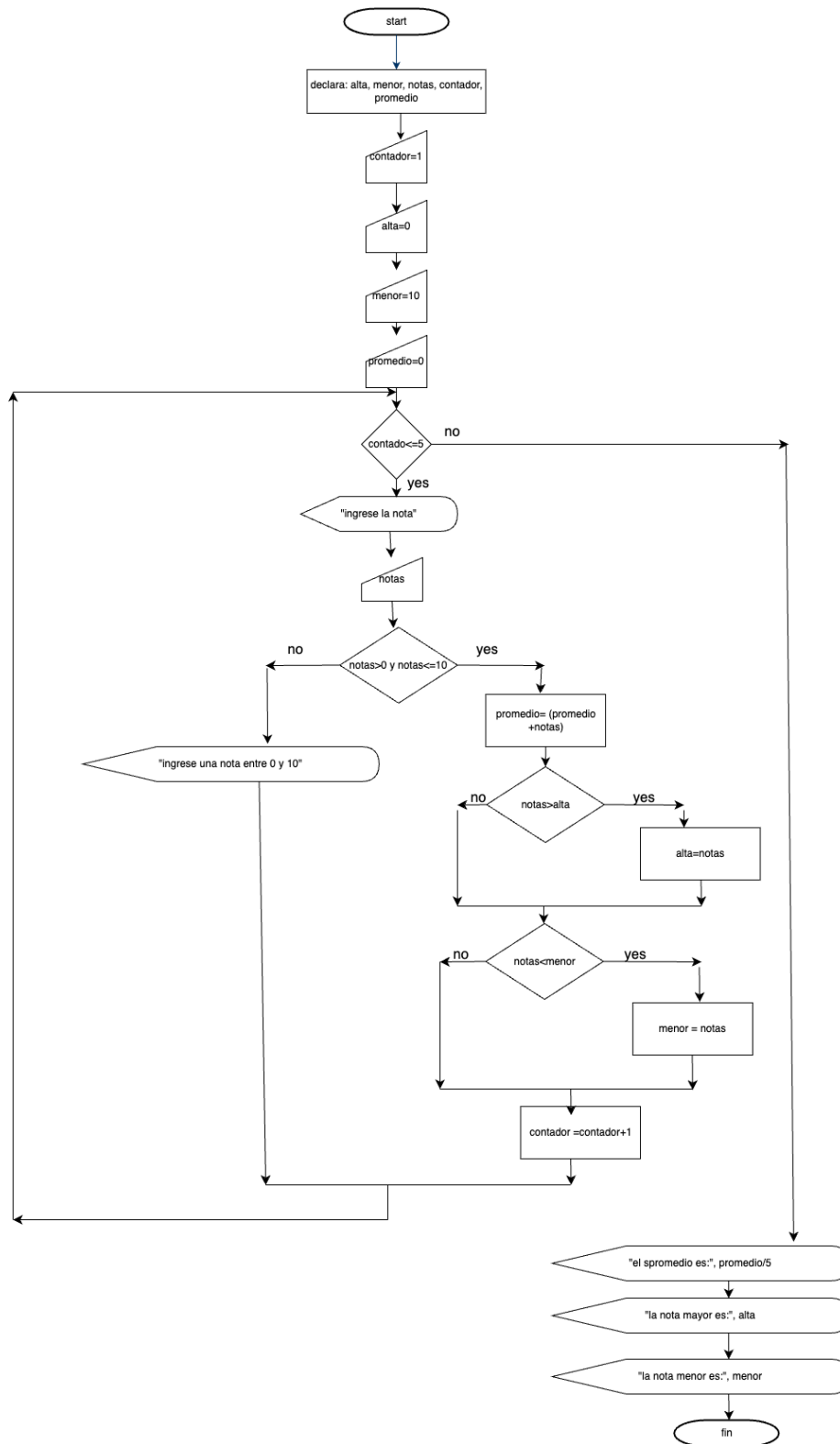
FinMientras

escribir "el promedio es:", promedio/5

escribir "la nota mayor es:", alta

escribir "la nota menor es:", menor

FinAlgoritmo





10. Diseñe el algoritmo correspondiente a un programa:

Cree una tabla bidimensional de longitud 3x4 y nómbrala "matriz".

Cargue la tabla con valores numéricos enteros.

Sume todos los elementos de cada fila, visualizando los resultados en la pantalla.

Sume todos los elementos de cada columna y muestre los resultados en la pantalla.

Datos		Identificadores
Problema a resolver	Crear una tabla bidimensional de longitud 3x4, cargarla con valores numéricos enteros, sumar todos los elementos de cada fila y de cada columna, y mostrar los resultados en pantalla.	
Entrada	Números para la matriz	matriz
Salida	Suma de las filas y suma de las columnas	Suma_filas suma_columnas

ALGORITMO matriz;

VAR

ENTERO matriz;

ENTERO suma_filas;

ENTERO suma_columnas;

ENTERO i;

ENTERO j;

INICIO

para i desde 0 hasta 2 :

para j desde 0 hasta 3:

escribir "escriba el valor para la posición

matriz[i][j]"

para i desde 0 hasta 2:

Suma_filas[i]=0

Para j desde 0 hasta 3:

Suma_filas[i]=suma_filas[i]+matriz[i][j]

finpara

escribir "la suma de la fila es", suma_filas[i]

para j desde 0 hasta 2:

Suma_columnas[j]=0

para i desde 0 hasta 3:

Suma_columnas[j]=suma_columnas[j]+matriz[i][j]

Finpara

finpara

Print "la suma de la columna es", suma_columnas[j]

FIN

