

Informe Técnico del Proceso

Aprendiz: Juan Carlos Lopez Moreno

Actividad: GA2-220501096-AA2-EV01

Programa: Programación de Aplicaciones y Servicios Para la Nube

Profesor: Alvaro Esteban Betancourt Matoma

Servicio Nacional de Aprendizaje

Guadalupe, Santander

Desarrollo de Los Ejercicios

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.

<p>Algoritmo calcular_numeros</p> <p>Definir numero1, numero2, numero3, resultado Como Entero</p> <p>1' Escribir 'Ingrese el número 1'</p> <p>Leer numero1</p> <p>2' Escribir 'Ingrese el número 2'</p> <p>Leer numero2</p> <p>3' Escribir 'Ingrese el número 3'</p> <p>Leer numero3</p> <p>Si numero1 < 0 Entonces</p> <p> resultado <- numero2 + numero3</p> <p> Escribir 'El resultado es: ', resultado</p> <p>SiNo</p> <p> resultado <- numero2 * numero3</p> <p> Escribir 'El resultado es: ', resultado</p> <p>FinAlgoritmo</p>	<pre> graph TD Start([Algoritmo calcular_numeros]) --> Decl[Definir numero1, numero2, numero3, resultado Como Entero] Decl --> In1[/Ingrese el número 1/] In1 --> Num1[numero1] Num1 --> In2[/Ingrese el número 2/] In2 --> Num2[numero2] Num2 --> In3[/Ingrese el número 3/] In3 --> Num3[numero3] Num3 --> Cond{numero1 < 0} Cond -- F --> Sum[resultado <- numero2 + numero3] Sum --> Out1[/El resultado es: ', resultado/] Cond -- V --> Mult[resultado <- numero2 * numero3] Mult --> Out2[/El resultado es: ', resultado/] Out1 --> End([FinAlgoritmo]) Out2 --> End </pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<p> $\text{resultado} \leftarrow$ $\text{numero2} + \text{numero3}$ Escribir 'El resultado es: ', resultado FinSi FinAlgoritmo </p>	
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

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.

Algoritmo tres_numeros_enteros

Definir numero1, numero2, numero3, resultado Como Entero

Escribir 'Digite el numero1'

Leer numero1

Escribir 'Digite el numero2'

Leer numero2

Escribir 'Digite el numero3'

Leer numero3

Si $\text{numero1} = (\text{numero2} + \text{numero3})$ Entonces

Escribir 'La suma del numero 2 mas el numero 3 coincide con el
valor del numero 1'

SiNo

Si $\text{numero2} = (\text{numero1} + \text{numero3})$ Entonces

Escribir 'La suma del numero 1 mas el numero 3 coincide con
el valor del numero 2'

SiNo

Si $\text{numero3} = (\text{numero1} + \text{numero2})$ Entonces

Escribir 'La suma del numero 1 mas el numero 2
coincide con el valor del numero 3'

SiNo

Escribir 'Ninguno de los números coinciden con la
suma de al menos 2'

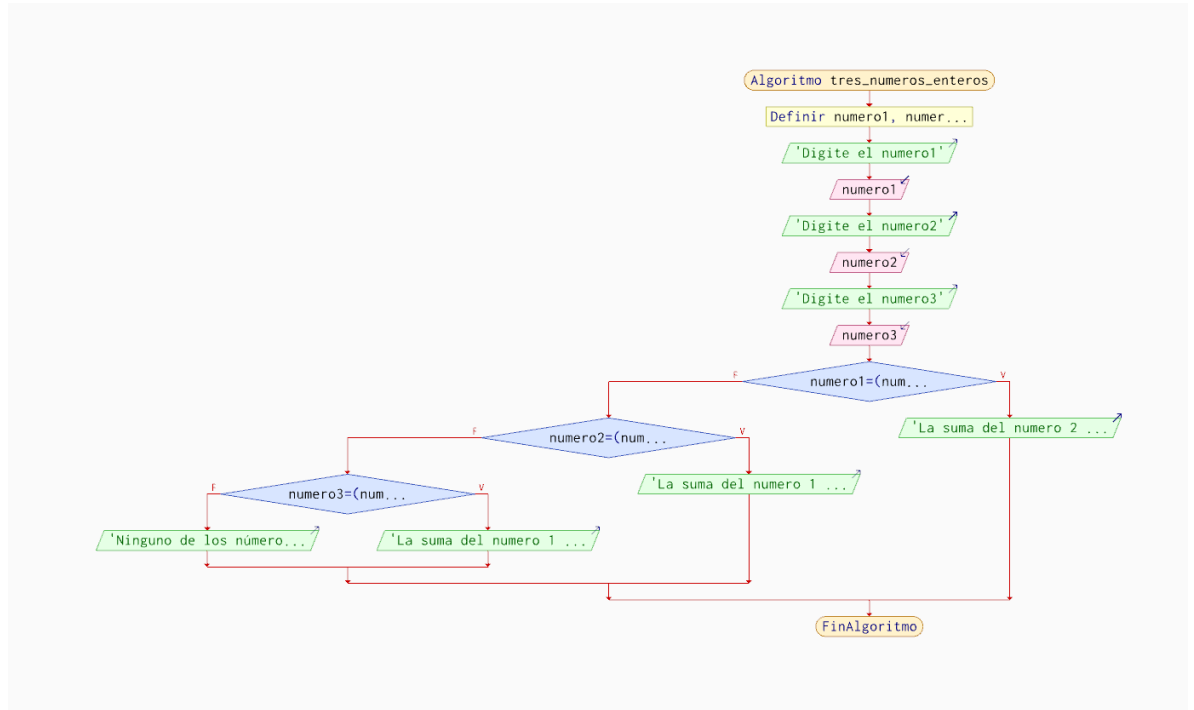
FinSi

FinSi

FinSi

FinAlgoritmo

Diagrama



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.

Algoritmo series_de_3

Definir resultado Como Entero

resultado <- 0

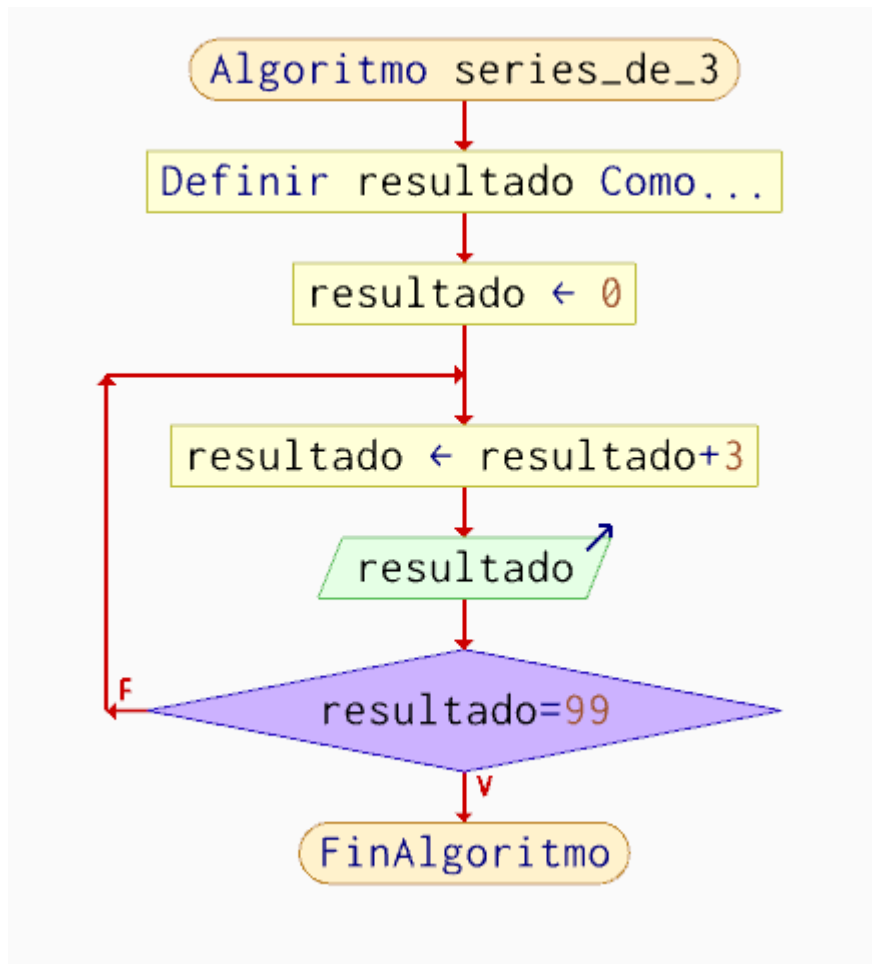
Repetir

 resultado <- resultado + 3

 Escribir resultado

Hasta Que resultado = 99

FinAlgoritmo



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

Algoritmo potencias_de_2

Definir potencia Como Entero

potencia <- 1

Repetir

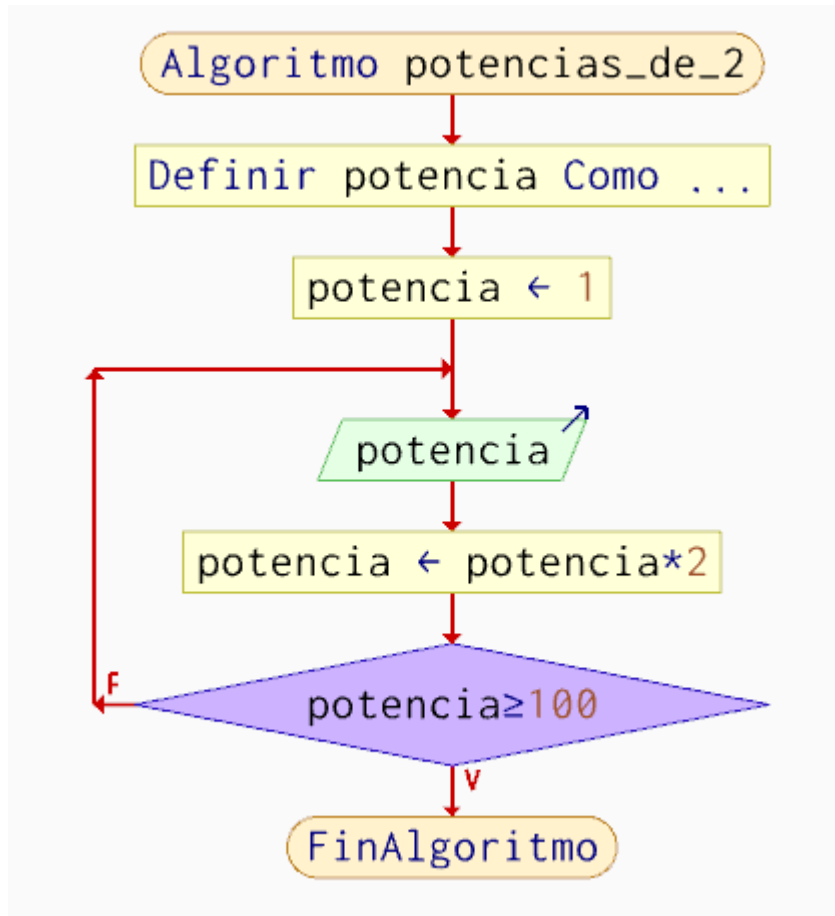
 Escribir potencia

 potencia <- potencia * 2

Hasta Que potencia ≥ 100

FinAlgoritmo

Diagrama



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

Algoritmo numeros_pares_rango

Para i <- 50 Hasta 200 Hacer

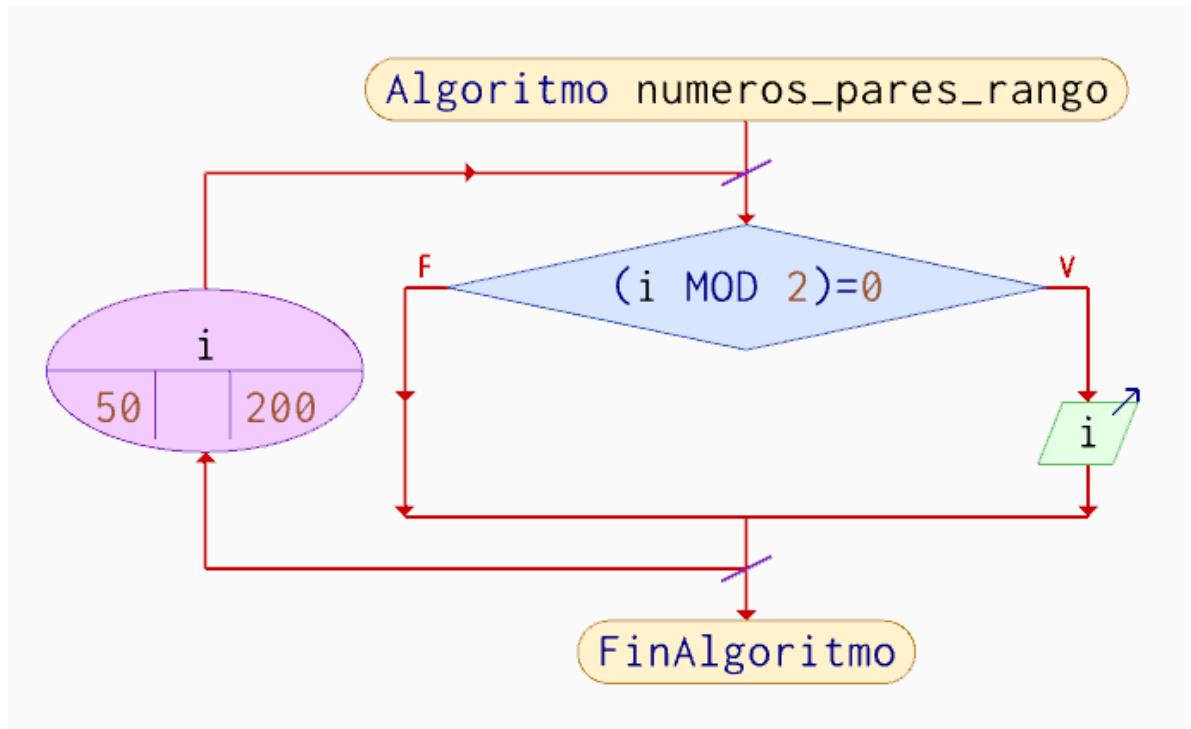
Si $(i \% 2) = 0$ Entonces

Escribir i

Fin Si

Fin Para

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.

Algoritmo celcius_to_fahrenheit

Definir fahrenheit, celsius Como Entero

Escribir "Ingrese los grados en celsius C°"

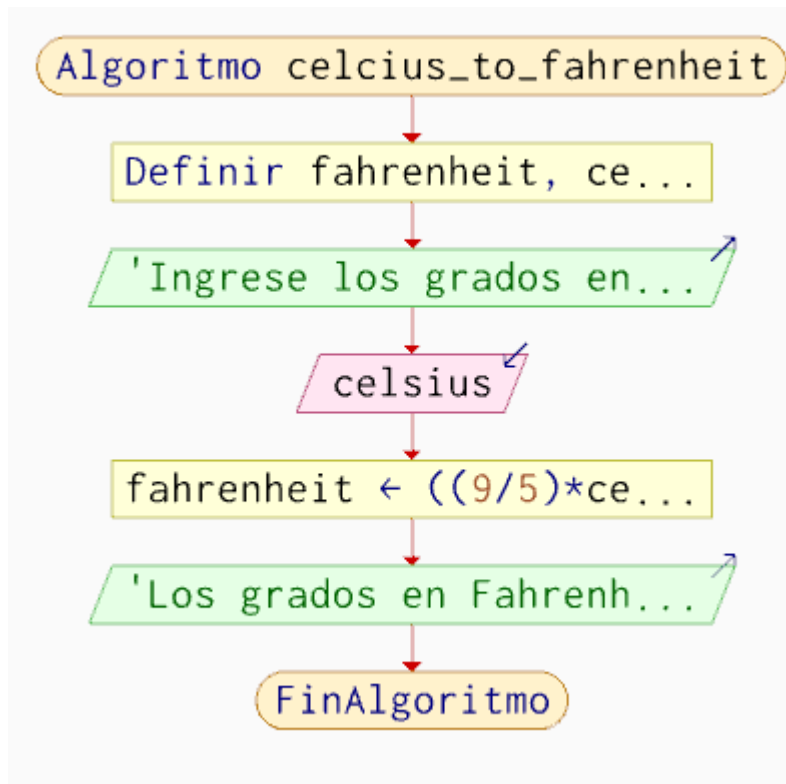
Leer celsius

$fahrenheit \leftarrow [(9 / 5) * celsius] + 32$

Escribir "Los grados en Fahrenheit son: " fahrenheit

FinAlgoritmo

Diagrama



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.

Algoritmo format_24_to_12

Definir hora_24, hora_12 Como Entero

Definir sufijo Como Caracter

Escribir "Ingresa la hora correspondiente en formato 24 (0 - 24)"

Leer hora_24

Si hora_24 < 0 o hora_24 > 24 Entonces

Escribir "Fuera del rango por favor digite el número correctamente"

SiNo

Si hora_24 = 0 entonces

hora_12 <- 12

sufijo <- "a.m"

Sino

Si hora_24 < 12 Entonces

hora_12 <- hora_24

sufijo <- "a.m."

Sino

Si hora_24 = 12 entonces

hora_12 <- 12

sufijo <- "p.m."

SiNo

hora_12 <- hora_24 % 12

sufijo <- "p.m."

FinSi

FinSi

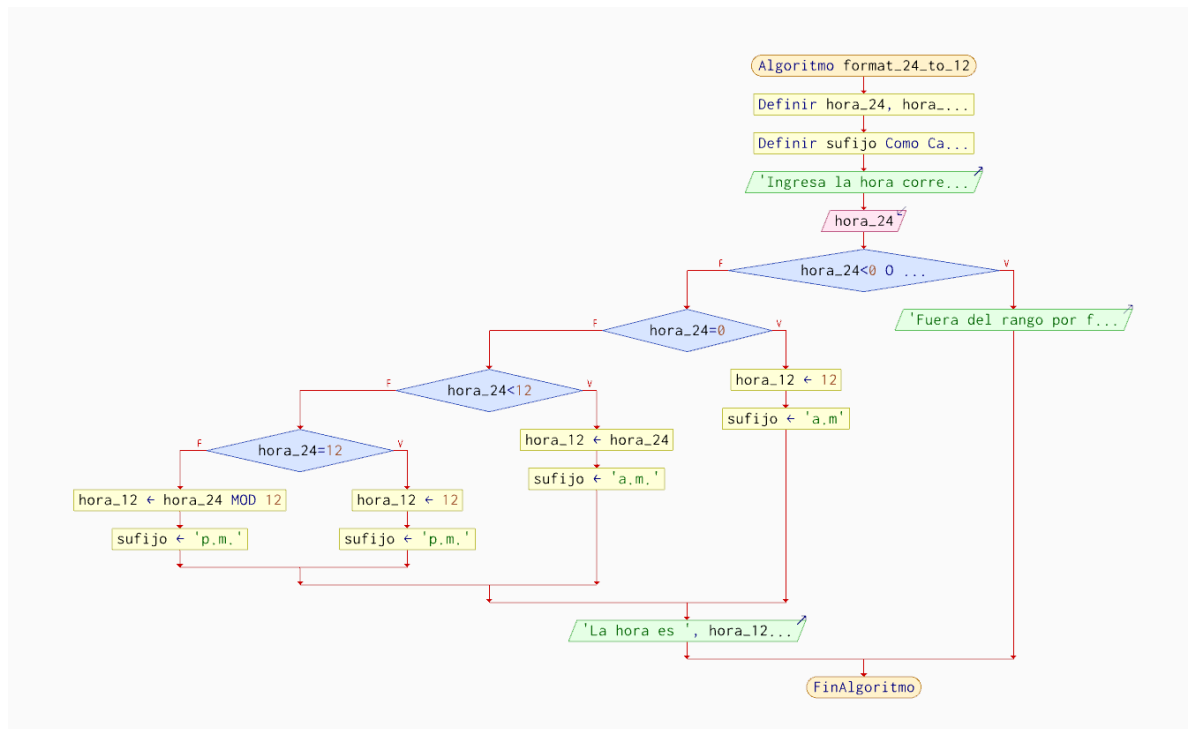
FinSi

Escribir "La hora es ", hora_12, " " sufijo

FinSi

FinAlgoritmo

Diagrama



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.

Algoritmo InvertirVectorCadenas

```
// Declaración de variables
```

```
Definir vectorOriginal, vectorInvertido Como Cadena
```

```
Definir i Como Entero
```

```
// Dimensionar los vectores para 5 elementos
```

```
Dimension vectorOriginal[5]
```

```
Dimension vectorInvertido[5]
```

```
// Inicialización: Leer elementos del vector original
```

```
Escribir "=== INGRESO DE DATOS ==="
```

```
Para i <- 1 Hasta 5 Hacer
```

```
    Escribir "Ingrese la cadena ", i, ": "
```

```
    Leer vectorOriginal[i]
```

```
FinPara
```

```
// Proceso: Copiar elementos en orden inverso
```

```
Para i <- 1 Hasta 5 Hacer
```

```
    vectorInvertido[i] <- vectorOriginal[6-i]
```

```
FinPara
```

```
// Salida: Mostrar vector original
```

Escribir ""

Escribir "=== VECTOR ORIGINAL ==="

Para i <- 1 Hasta 5 Hacer

 Escribir "Posición ", i, ": ", vectorOriginal[i]

FinPara

// Salida: Mostrar vector invertido

Escribir ""

Escribir "=== VECTOR INVERTIDO ==="

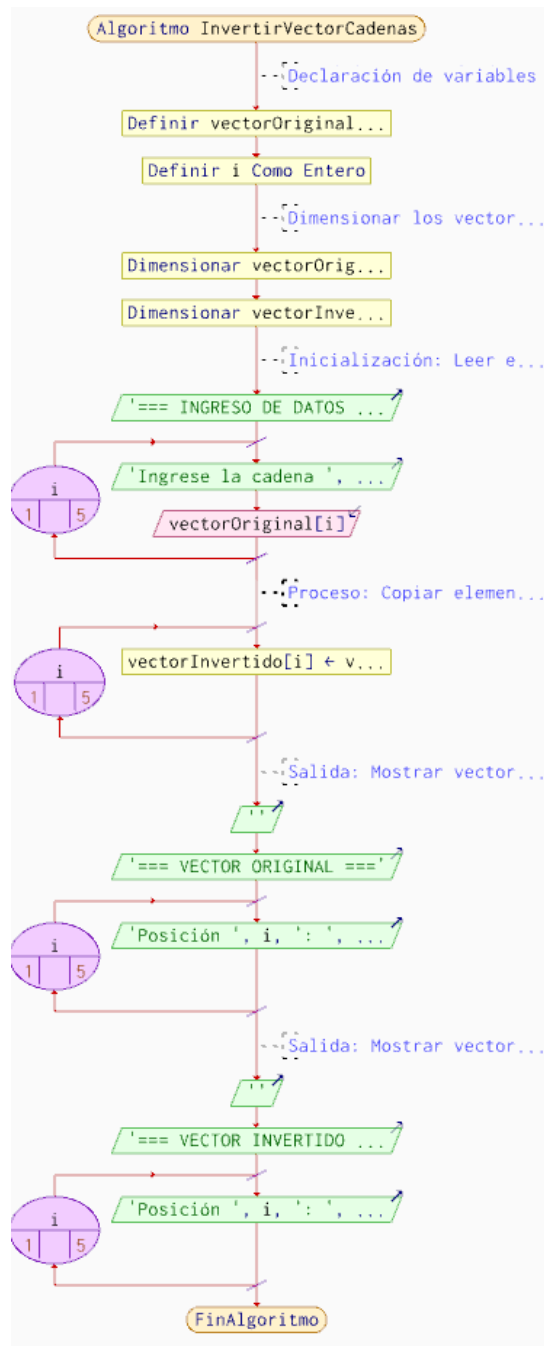
Para i <- 1 Hasta 5 Hacer

 Escribir "Posición ", i, ": ", vectorInvertido[i]

FinPara

FinAlgoritmo

Diagrama de Flujo



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.

Algoritmo notas_alumno

```
// Definir las variables
```

```
Definir i, notas, suma, promedio, notaAlta, notaBaja Como Real
```

```
// Dimensionamos en un array las notas
```

```
Definir cantidad como entero
```

```
cantidad <- 5
```

```
Dimensionar notas[cantidad]
```

```
// Aplicamos un bucle for para ingresar las notas
```

```
Para i <- 1 Hasta 5 Hacer
```

```
    Escribir "Ingrese la nota"
```

```
    Leer notas[i]
```

```
FinPara
```

```
// Aplicamos otro bucle para sumar las notas
```

```
Para i <- 1 hasta 5 hacer
```

```
    suma <- suma + notas[i]
```

```
FinPara
```

```
// Calculo del promedio

promedio <- suma / cantidad

Escribir "El promedio es: " promedio


// Calculo de la nota mas alta

// Inicializar variables

notaBaja <- 5.1

notaAlta <- 0

Para i <- 1 hasta 5 Hacer

    notaActual <- notas[i]


    Si notaActual > notaAlta Entonces

        notaAlta <- notaActual

    FinSi


    Si notaActual < notaBaja Entonces

        notaBaja <- notaActual

    FinSi

FinPara


Escribir "Esta es la nota mas alta: ", notaAlta

Escribir "Esta es la nota mas baja: ", notaBaja

FinAlgoritmo
```



```

graph TD
    Start([Algoritmo notas_almuno]) --> Def1[Definir i, notas, suma...]
    Def1 --> Def2[Definir cantidad Como...]
    Def2 --> Cant[cantidad ← 2]
    Cant --> Dim[Dimensionar notas(cant...]
    Dim --> For[Aplicamos un bucle for...]
    For --> While1((while i <= cantidad))
    While1 --> Ingresar[Ingresar la nota]
    Ingresar --> Notas[notas[i]]
    Notas --> Suma[suma ← suma+notas[i]]
    Suma --> IncI[i ← i+1]
    IncI --> While1
    While1 --> CalcProm[Calculo del promedio]
    CalcProm --> Prom[promedio ← suma/cantidad]
    Prom --> PrintProm[El promedio es: ' pr...]
    PrintProm --> CalcNota[Calculo de la nota mas...]
    CalcNota --> Init[Inicializar variables]
    Init --> NotaBaja[notaBaja ← 0]
    NotaBaja --> NotaAlta[notaAlta ← 10]
    NotaAlta --> While2((while i <= cantidad))
    While2 --> Actual[notasActual ← notas[i]]
    Actual --> If1{notasActual > notaAlta}
    If1 --> ActualAlta[notaAlta ← notasActual]
    ActualAlta --> If2{notasActual < notaBaja}
    If2 --> ActualBaja[notaBaja ← notasActual]
    ActualBaja --> IncI2[i ← i+1]
    IncI2 --> While2
    While2 --> PrintBaja[Esta es la nota mas b...]
    PrintBaja --> PrintAlta[Esta es la nota mas a...]
    PrintAlta --> End([FinAlgoritmo])
  
```

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.