

**P.U.I. – F.C.E.y T. – UNSE**

***FUNDAMENTOS DE LA PROGRAMACIÓN (PUI)***

**TRABAJO PRÁCTICO 3**: **ARREGLOS BIDIMENSIONALES (MATRICES)**

**AÑO 2022**

# Al finalizar este práctico se espera que los alumnos logren:

´`

* Realizar la búsqueda creativa de solución/es algorítmica/s, si es que existe/n, y seleccionar la alternativa más adecuada.
* Seleccionar estructuras de datos adecuadas para la resolución de problemas.
* Emplear, según la situación problemática, algoritmos fundamentales de recorrido, búsqueda, ordenamiento y actualización.
* Verificar la solución algorítmica.

Actividad: Dados los siguientes problemas, expresar su solución mediante un algoritmo y verificar si se obtienen el/ los resultado/ s esperado/s.

# Lectura /Escritura

1. Generar la matriz UNSE (Nx4) con los datos de los libros de la Biblioteca de una Universidad: código de libro, código de autor, número de ejemplares disponibles y precio. Con la matriz UNSE (Nx4) generada se pide; mostrar todos los códigos de libro y autor si el precio es mayor a 1500.

Datos de entrada: Matriz UNSE de N filas y 4 columnas con los datos de los libros de la Biblioteca de una Universidad

Datos de salida: Código del libro y autor si el precio es mayor a 1500

Pseudocódigo:

Inicio

Leer N

Dimension UNSE[N,4]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < 4) Hacer

Leer UNSE[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

i = 0

Mientras (i < N) Hacer

Si (UNSE[i,3] > 1500) Entonces

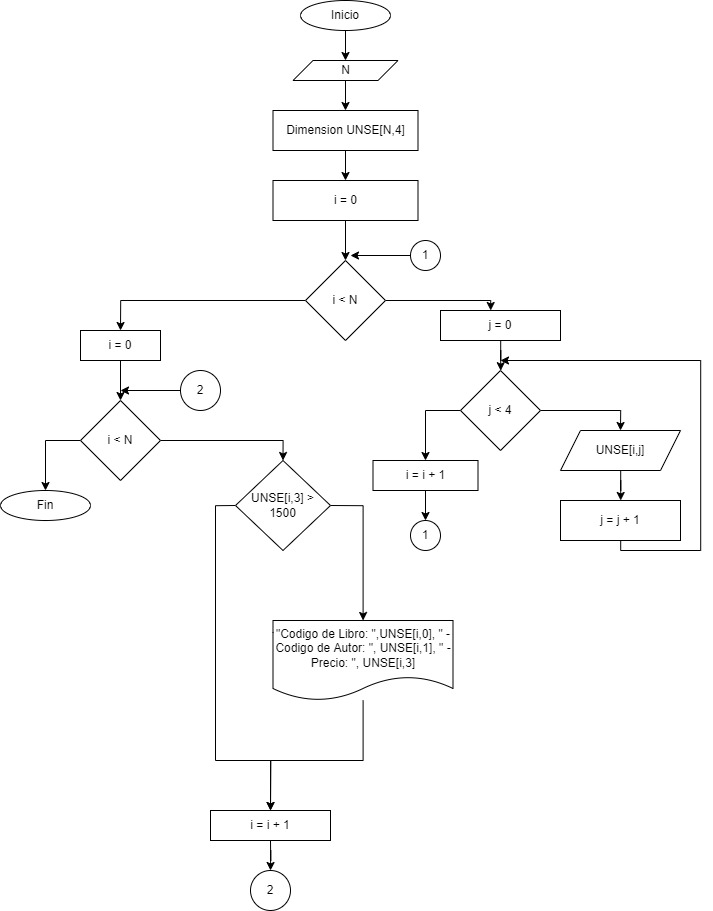
Escribir "Codigo de Libro: ",UNSE[i,0], " - Codigo de Autor: ", UNSE[i,1], " - Precio: ", UNSE[i,3]

FinSi

i = i + 1

FinMientras

Fin



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **i** | **j** | **UNSE[i,j]** | **Imprimir** |
| 3 | 0 | 0 | 15 |  |
|  |  | 1 | 4 |  |
|  |  | 2 | 90 |  |
|  |  | 3 | 200 |  |
|  |  | 4 |  |  |
|  | 1 | 0 | 19 |  |
|  |  | 1 | 2 |  |
|  |  | 2 | 20 |  |
|  |  | 3 | 1700 |  |
|  |  | 4 |  |  |
|  | 2 | 0 | 25 |  |
|  |  | 1 | 8 |  |
|  |  | 2 | 50 |  |
|  |  | 3 | 500 |  |
|  |  | 4 |  |  |
|  | 3 |  |  |  |
|  | 0 |  |  |  |
|  | 1 |  |  | Codigo de Libro: 19 - Codigo de Autor: 2 - Precio: 1700 |
|  | 2 |  |  |  |

UNSE

|  |  |  |  |
| --- | --- | --- | --- |
| 15 | 4 | 90 | 200 |
| 19 | 2 | 20 | 1700 |
| 25 | 8 | 50 | 500 |

# Recorrido

1. Dada la matriz W (NxN), realizar:
2. Intercambiar los elementos de las columnas impares con los elementos de las columnas pares.

Datos de entrada: Matriz de N filas y N columnas

Datos de salida:

Pseudocodigo:

Inicio

Leer N

Dimension W[N,N]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < N) Hacer

Leer W[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < N - 1) Hacer

Si (j mod 2 <> 0) Entonces

aux = W[i,j]

W[i,j] = W[i,j+1]

W[i,j+1] = aux

FinSi

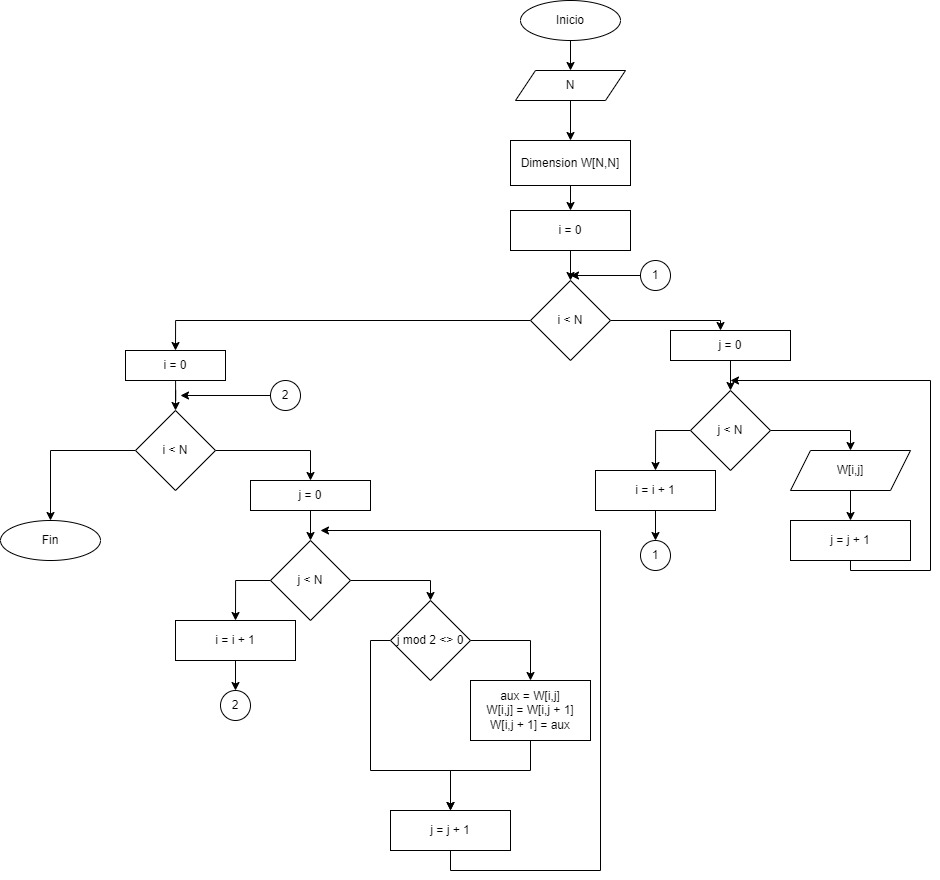
j = j + 1

FinMientras

i = i + 1

FinMientras

Fin



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **N** | **i** | **j** | **W[i,j]** | **aux** | **W[i,j + 1]** | **Imprimir** |
| 3 | 0 | 0 | 25 |  |  |  |
|  |  | 1 | 30 |  |  |  |
|  |  | 2 | 10 |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 1 | 0 | 57 |  |  |  |
|  |  | 1 | 66 |  |  |  |
|  |  | 2 | 70 |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 2 | 0 | 22 |  |  |  |
|  |  | 1 | 23 |  |  |  |
|  |  | 2 | 30 |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 3 |  |  |  |  |  |
|  | 0 | 0 |  |  |  |  |
|  |  | 1 | 10 | 30 | 30 |  |
|  |  | 2 |  |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 1 | 0 |  |  |  |  |
|  |  | 1 | 70 | 66 | 66 |  |
|  |  | 2 |  |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 2 | 0 |  |  |  |  |
|  |  | 1 | 30 | 23 | 23 |  |
|  |  | 2 |  |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 3 |  |  |  |  |  |

W

|  |  |  |
| --- | --- | --- |
| 25 | 30 | 10 |
| 57 | 66 | 70 |
| 22 | 23 | 30 |

Matriz W resultante al cambiar columnas impares por pares

|  |  |  |
| --- | --- | --- |
| 25 | 10 | 30 |
| 57 | 70 | 66 |
| 22 | 30 | 23 |

1. Dadas dos matrices cuadradas A y B, de NxN elementos cada una, se pide:
2. A partir de los datos de la matriz A(NxN) generar un vector V con los elementos de la diagonal secundaria que sean pares.

Datos de entrada: Matriz Cuadrada A

Datos de salida:

Pseudocodigo:

Inicio

Leer N

Dimension A[N,N]

Dimension V[N\*N]

i = 0

Mientras (i < N) Hacer

j <- 0

Mientras (j < N) Hacer

Leer A[i,j]

j = j + 1

FinMientras

i <- i + 1

FinMientras

i = 0

N2 = 0

Mientras (i < N) Hacer

Si(A[i, N - 1 - i] mod 2 = 0) Entonces

V[N2] = A[i, N - 1 - i]

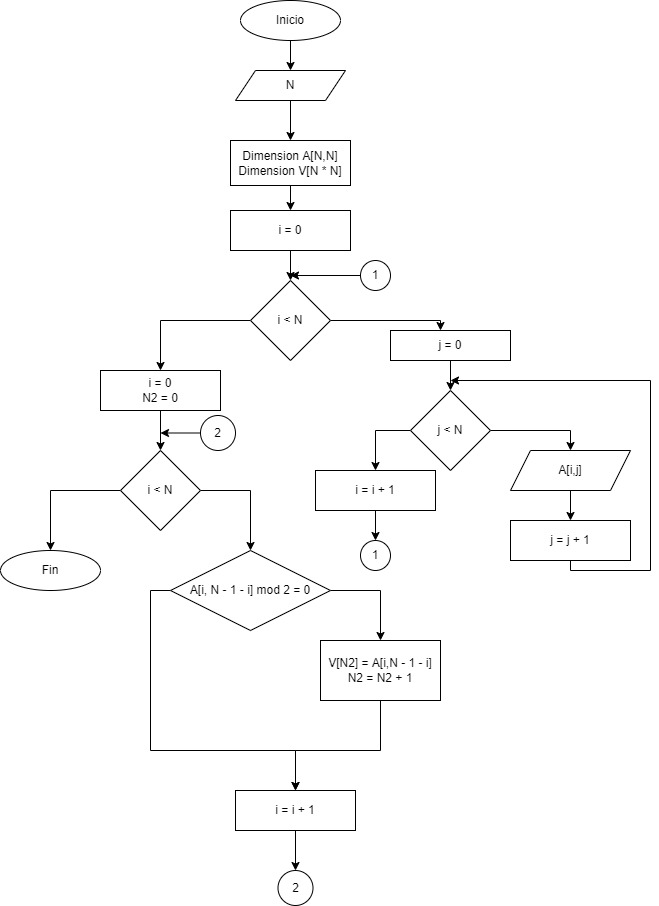
N2 = N2 + 1

FinSi

i = i + 1

FinMientras

Fin



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **N** | **i** | **j** | **A[i,j]** | **N2** | **V[N2]** | **Imprimir** |
| 3 | 0 | 0 | 20 |  |  |  |
|  |  | 1 | 5 |  |  |  |
|  |  | 2 | 10 |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 1 | 0 | 14 |  |  |  |
|  |  | 1 | 12 |  |  |  |
|  |  | 2 | 23 |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 2 | 0 | 55 |  |  |  |
|  |  | 1 | 41 |  |  |  |
|  |  | 2 | 3 |  |  |  |
|  |  | 3 |  |  |  |  |
|  | 3 |  |  |  |  |  |
|  | 0 |  |  | 0 | 10 |  |
|  | 1 |  |  | 1 | 12 |  |
|  | 2 |  |  |  |  |  |
|  | 3 |  |  |  |  |  |

A

|  |  |  |
| --- | --- | --- |
| 20 | 5 | 10 |
| 14 | 12 | 23 |
| 55 | 41 | 3 |

V

|  |  |
| --- | --- |
| 10 | 12 |

# Búsqueda

1. Mostrar los valores primos de la matriz P(NxM).

Datos de entrada: Matriz P de N filas y M columnas

Datos de salida: Matriz P

Pseudocodigo:

Inicio

Leer N

Leer M

Dimension P[N,M]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < M) Hacer

Leer P[i,j]

j = j + 1

FinMientras

i <- i + 1

FinMientras

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < M) Hacer

x = 1

c = 0

Mientras (x<=P[i,j]) Hacer

Si (P[i,j] MOD (x) = 0) Entonces

c = c + 1

FinSi

x = x + 1

FinMientras

Si (c = 2) Entonces

Escribir "El numero " P[i,j] " es primo"

FinSi

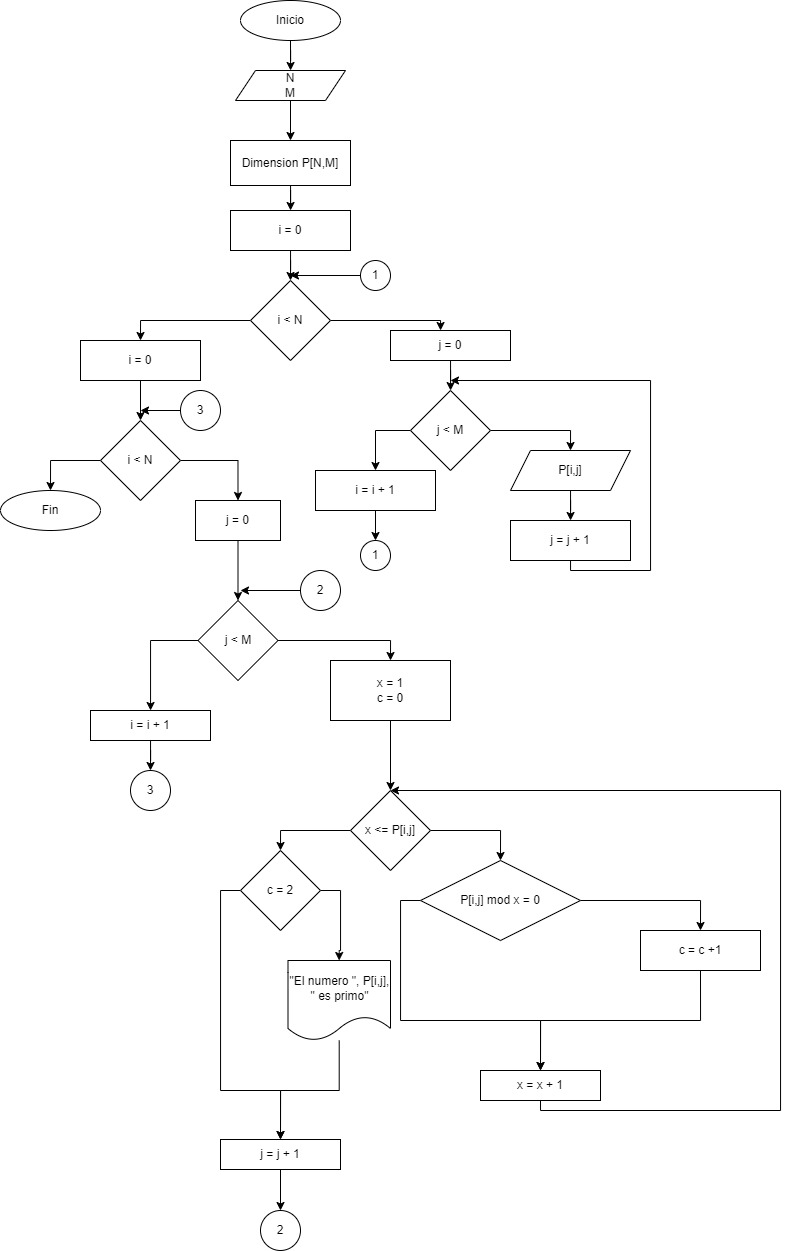
j = j + 1

FinMientras

i = i + 1

FinMientras

Fin



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **N** | **M** | **i** | **j** | **P[i,j]** | **x** | **c** | **Imprimir** |
| 3 | 4 | 0 | 0 | 15 |  |  |  |
|  |  |  | 1 | 20 |  |  |  |
|  |  |  | 2 | 40 |  |  |  |
|  |  |  | 3 | 55 |  |  |  |
|  |  |  | 4 |  |  |  |  |
|  |  | 1 | 0 | 13 |  |  |  |
|  |  |  | 1 | 10 |  |  |  |
|  |  |  | 2 | 22 |  |  |  |
|  |  |  | 3 | 36 |  |  |  |
|  |  |  | 4 |  |  |  |  |
|  |  | 2 | 0 | 24 |  |  |  |
|  |  |  | 1 | 78 |  |  |  |
|  |  |  | 2 | 4 |  |  |  |
|  |  |  | 3 | 8 |  |  |  |
|  |  |  | 4 |  |  |  |  |
|  |  | 0 | 0 |  | 1 | 0 |  |
|  |  |  |  |  | 2 | 1 |  |
|  |  | 1 | 0 |  | 1 | 0 |  |
|  |  |  |  |  | 2 | 1 |  |
|  |  |  |  |  |  | 2 | El numero 13 es primo |

P

|  |  |  |  |
| --- | --- | --- | --- |
| 15 | 20 | 40 | 55 |
| 13 | 10 | 22 | 36 |
| 24 | 78 | 4 | 8 |

1. Dada la matriz NUM(NxM) indicar si un determinado valor existe en la diagonal principal, que se encuentra ordenada en forma ascendente: Si no existe mostrar el mensaje “el número buscado no existe”.

Datos de entrada: Matriz NUM[N,M]

Datos de salida: Mostrar “el número buscado no existe”

Pseudocodigo:

Inicio

Leer N

Leer M

Dimension NUM[N,M]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < M) Hacer

Leer NUM[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

i = 0

Escribir "Ingresa el numero a buscar en la diagonal principal: "

Leer W

M = trunc((i + N) / 2)

Mientras (i < N & i < M & W <> NUM[M,M]) Hacer

Si (W < NUM[M,M]) Entonces

N = M - 1

SiNo

i = M + 1

FinSi

M = trunc((i + N) / 2)

FinMientras

Si(W = NUM[M,M]) Entonces

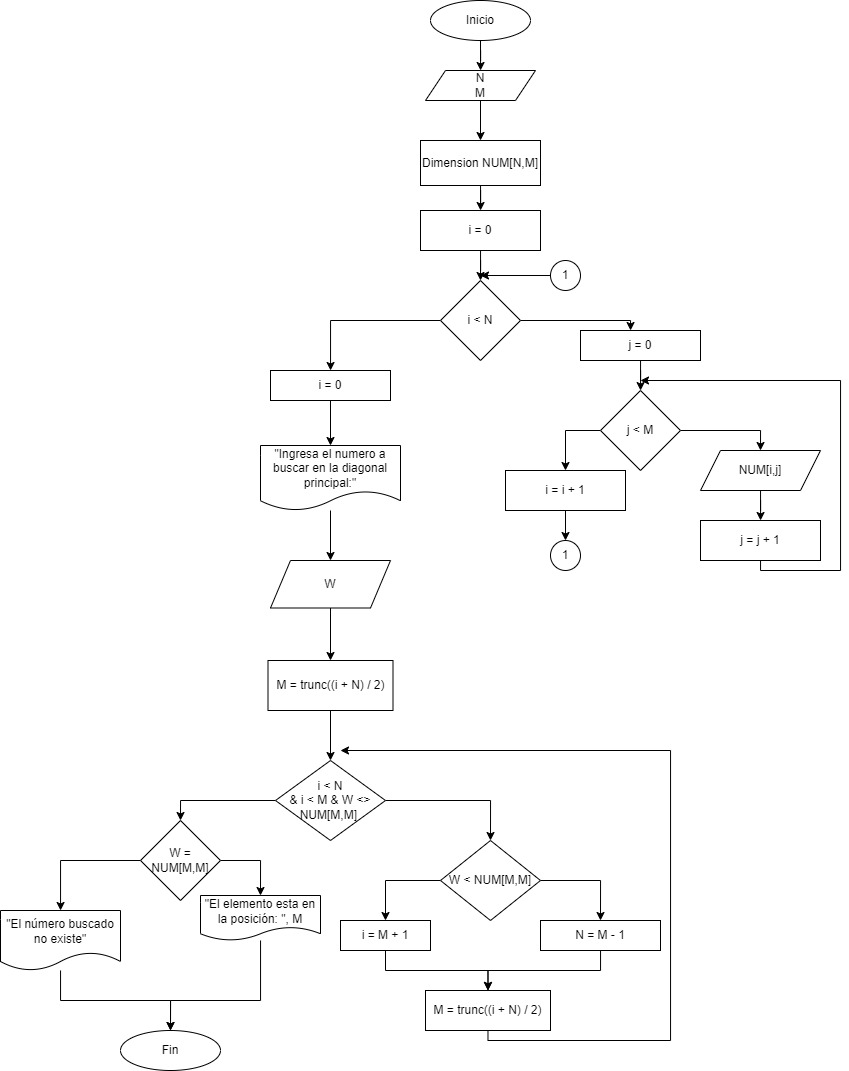
Escribir "El elemento esta en la posición: ", M

SiNo

Escribir "El número buscado no existe"

FinSi

Fin



1. Dada la matriz PROD (15x8) indicar la cantidad de valores mayores a 120 y los números pares en las columnas impares. Si no existen mostrar el mensaje “valores no encontrados”.

Datos de entrada: Matriz PROD[15,8]

Datos de salida: Mostrar “Valores no encontrados”

Pseudocodigo:

Inicio

Dimension PROD[15,8]

i = 0

Mientras (i < 15) Hacer

j = 0

Mientras (j < 8) Hacer

Leer PROD[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

i = 0

c120 = 0

cp = 0

Mientras (i < 15) Hacer

j = 0

Mientras (j < 8) Hacer

Si(PROD[i,j] > 120) Entonces

c120 = c120 + 1

FinSi

Si (j mod 2 <> 0 & PROD[i,j] mod 2 = 0) Entonces

cp = cp + 1

FinSi

j = j + 1

FinMientras

i = i + 1

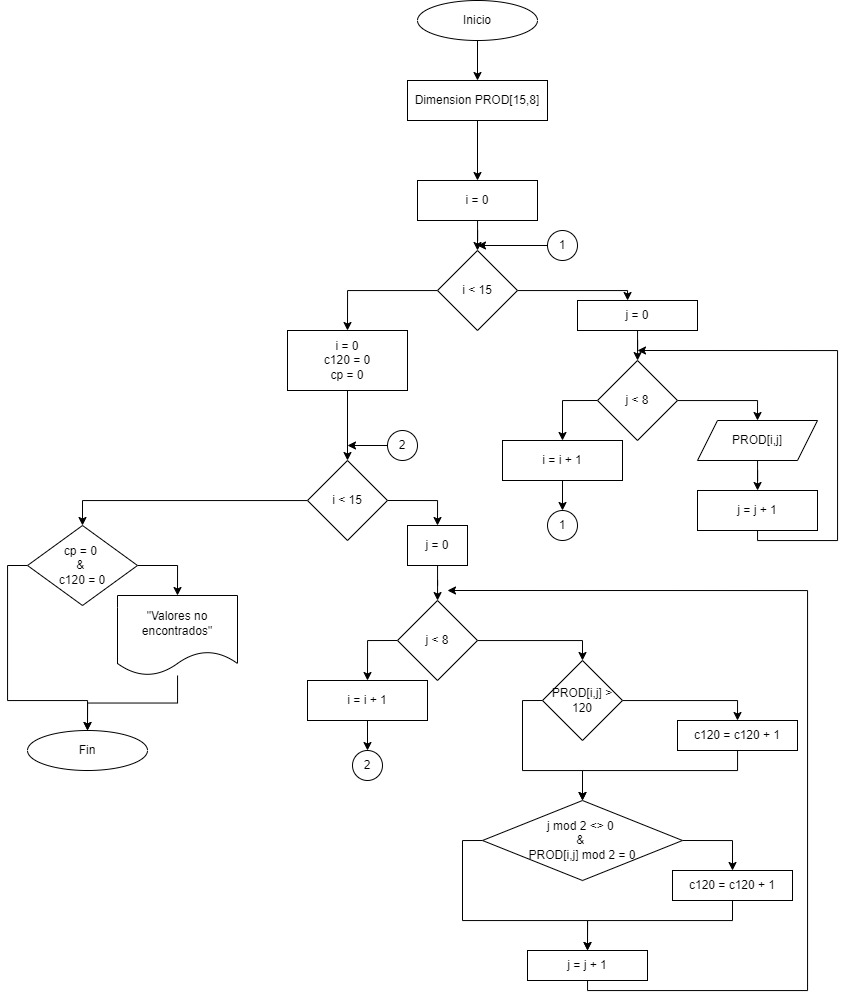
FinMientras

Si (cp = 0 & c120 = 0) Entonces

Escribir "Valores no encontrados"

FinSi

Fin



# Actualización

1. Dada la matriz UNSE (Nx5) con los datos de los libros de la Biblioteca de una Universidad: código de libro, código de autor, número de ejemplares disponibles, año de edición y precio; se pide:
   1. Añadir los datos de 4 libros comprados por la Biblioteca.
   2. Eliminar aquellos ejemplares cuyo año de edición sea menor a 1995.
   3. Generar un vector B con los libros cuyo precio sea menor a $2000

Datos de entrada: Matriz UNSE[N,5]

Datos de salida:

Pseudocodigo:

Inicio

Leer N

Dimension UNSE[N + 4,5]

Dimension B[N+4]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < 5) Hacer

Leer UNSE[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

// a)

Escribir "Ingrese los datos de los libros comprados por la Biblioteca"

i = N

N = N + 4

Mientras (i < N) Hacer

j = 0

Mientras (j < 5) Hacer

Leer UNSE[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

// b)

i = 0

Mientras (i < N) Hacer

Si (UNSE[i,3] = 1995) Entonces

K= i

Mientras (k < N - 1) Hacer

j = 0

Mientras (j < 5) Hacer

UNSE[k,j] = UNSE[k + 1, j]

j = j + 1

FinMientras

k = k + 1

FinMientras

N = N - 1

SiNo

i = i + 1

FinSi

FinMientras

// c)

i = 0

j = 0

Mientras (i < N) Hacer

Si (UNSE[i,4] < 2500) Entonces

B[j] = UNSE[i,0]

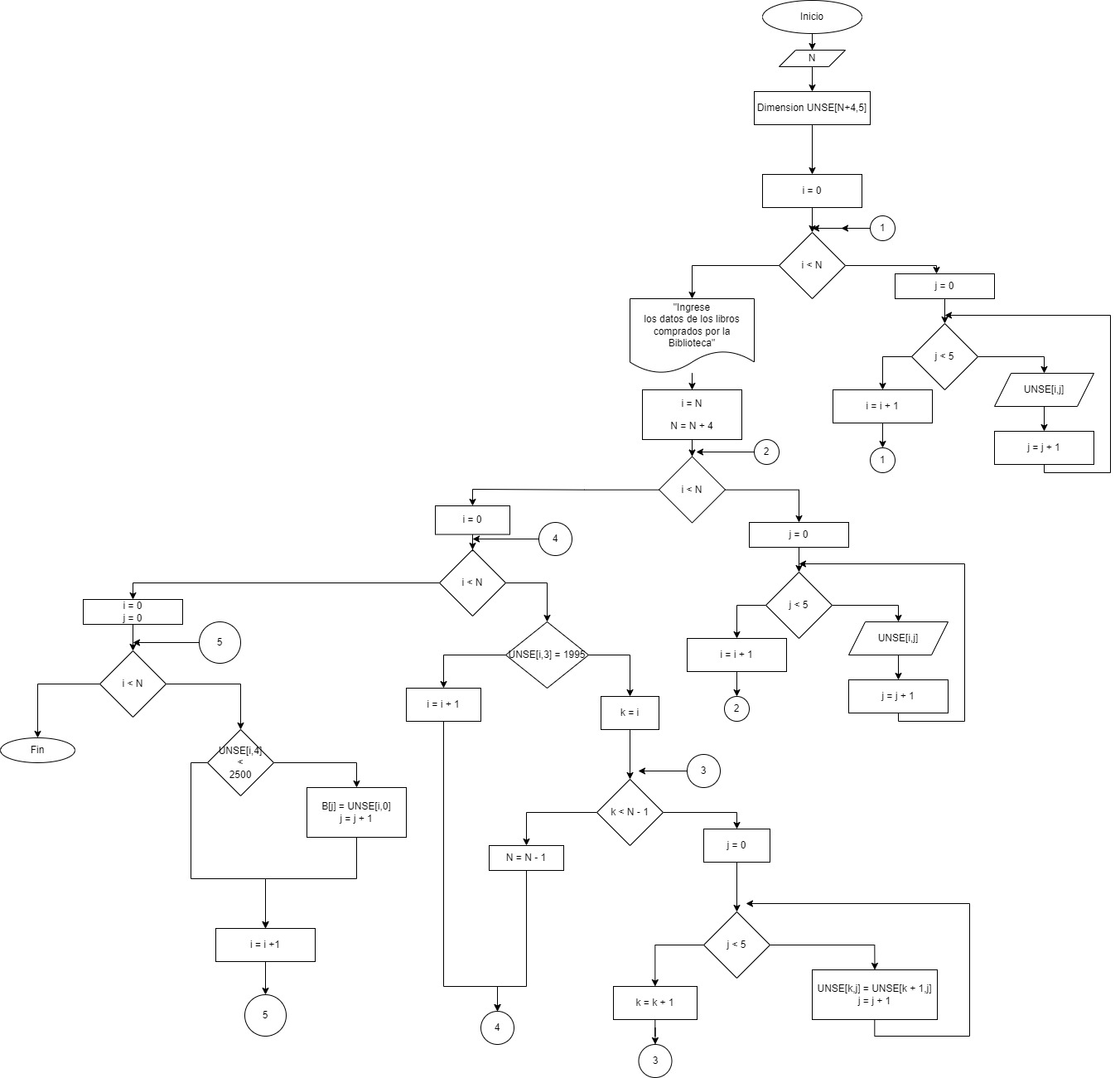
j = j + 1

FinSi

i = i + 1

FinMientras

Fin



1. Se tiene una planilla con los ingresos obtenidos por cada uno de los N vendedores, en la venta de M productos de una zapatería. Los dueños desean:
   1. Registrar en la planilla el promedio de ventas de cada empleado.
   2. Para aquellos empleados que obtuvieron un promedio de ventas inferior a $50000, comunicarles su despido y actualizar la planilla.
   3. Ingresar los datos de tres nuevos empleados.

Datos de entrada: Planilla de N vendedores

Datos de salida:

Pseudocodigo:

Inicio

Escribir "Numero de vendedores"

Leer N

Escribir "Productos vendidos del vendedor"

Leer M

Dimension P[N + 5 ,M + 5]

i = 0

Mientras (i < N) Hacer

j = 0

b = 0

Escribir "Vendedor " i + 1 ": "

Mientras (j < M - 1 & b = 0) Hacer

Escribir "Precio del producto vendido (0 si no vendio mas)"

Leer P[i,j]

Si (P[i,j] = 0) Entonces

b = 1

SiNo

j = j + 1

FinSi

FinMientras

i <- i + 1

FinMientras

// a)

i <- 0

Mientras (i < N) Hacer

j = 0

prom = 0

sum = 0

Mientras (j < M - 1) Hacer

sum = sum + P[i,j]

j = j + 1

FinMientras

prom = sum / M - 1

P[i,M - 1] = prom

// b)

Si(prom < 50000) Entonces

Escribir "Estas despedido"

j = i

Mientras (j < N) Hacer

k = 0

Mientras (k < M) Hacer

P[j,k] = P[j + 1,k]

k = k + 1

FinMientras

j = j + 1

FinMientras

N = N - 1

SiNo

i = i + 1

FinSi

FinMientras

// c)

i = N

N = N + 3

Mientras (i < N) Hacer

j = 0

b = 0

Escribir "Vendedor " i + 1 ": "

Mientras (j < M - 1 & b = 0) Hacer

Escribir "Precio del producto vendido (0 si no vendio mas)"

Leer P[i,j]

Si (P[i,j] = 0) Entonces

b = 1

SiNo

j = j + 1

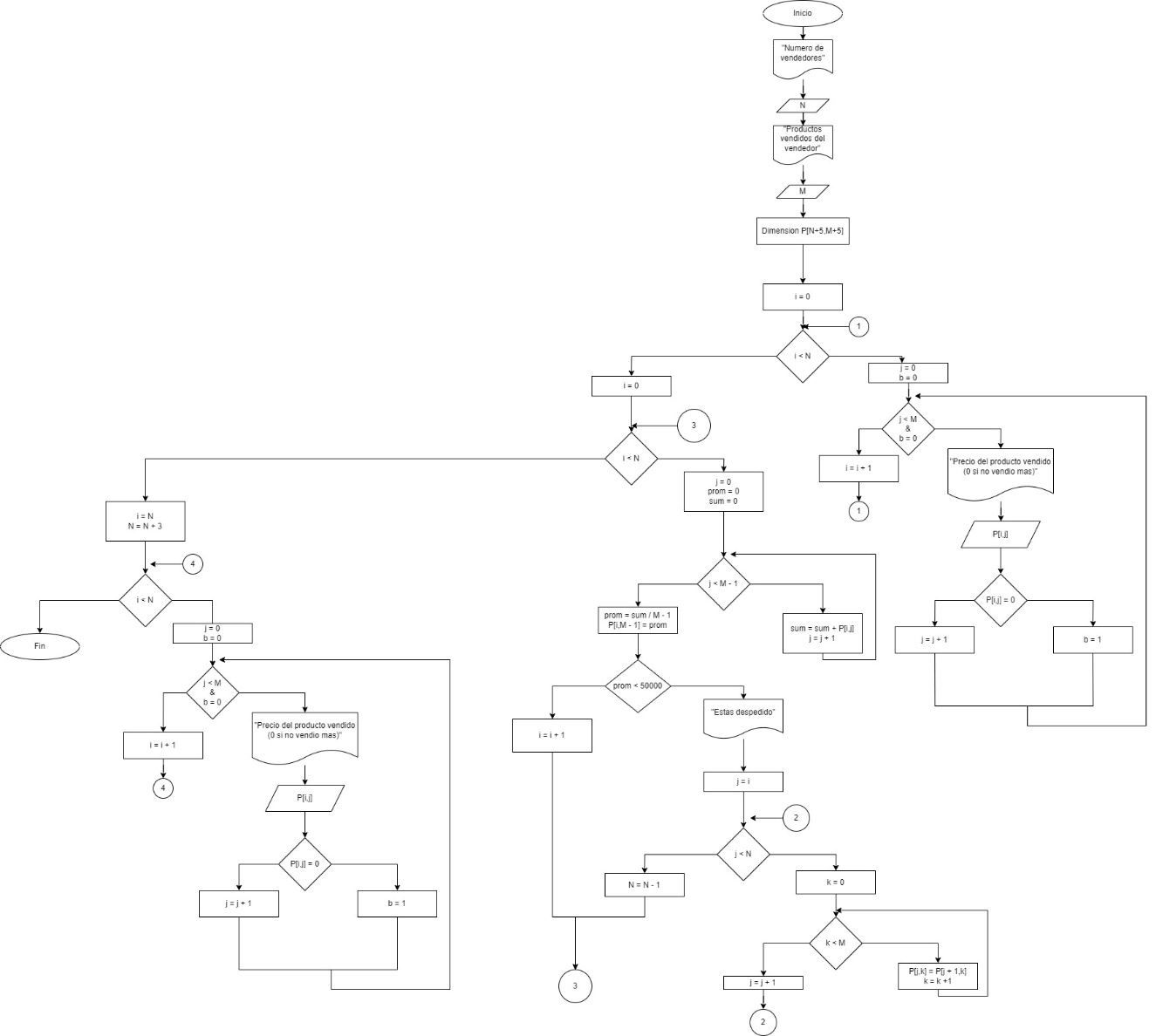
FinSi

FinMientras

i = i + 1

FinMientras

Fin



# Ordenación

1. Dada la matriz LEG (NxM), mostrar:
   1. Ordenar en forma descendente los elementos de la diagonal principal.
   2. Ordenar en forma ascendente los elementos de la triangular superior.

Datos de entrada: LEG[N,M]

Datos de salida:

Pseudocodigo:

Inicio

Leer N

Leer M

Dimension LEG[N,M]

Dimension V[N\*M]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < M) Hacer

Leer LEG[i,j]

j = j + 1

FinMientras

i <- i + 1

FinMientras

// a)

i = 0

Mientras (i < N) Hacer

p = i

j = i + 1

Mientras (j < N) Hacer

Si (LEG[j,j] > LEG[p,p]) Entonces

p = j

FinSi

j = j +1

FinMientras

W = LEG[p,p]

LEG[p,p] = LEG[i,i]

LEG[i,i] = W

i = i + 1

FinMientras

// b)

i = 0

k = 0

Mientras (i < N) Hacer

j = i + 1

Mientras (j < M) Hacer

V[k] = LEG[i,j]

k = k + 1

j = j + 1

FinMientras

i = i + 1

FinMientras

Repetir

b = 0

i = 0

Mientras (i < k - 1) Hacer

Si(V[i] > V[i + 1]) Entonces

W = V[i]

V[i] = V[i + 1]

V[i + 1] = W

b = 1

FinSi

i = i + 1

FinMientras

Hasta Que b = 0

i = 0

k = 0

Mientras (i < N) Hacer

j = i + 1

Mientras (j < M) Hacer

LEG[i,j] = V[k]

k = k + 1

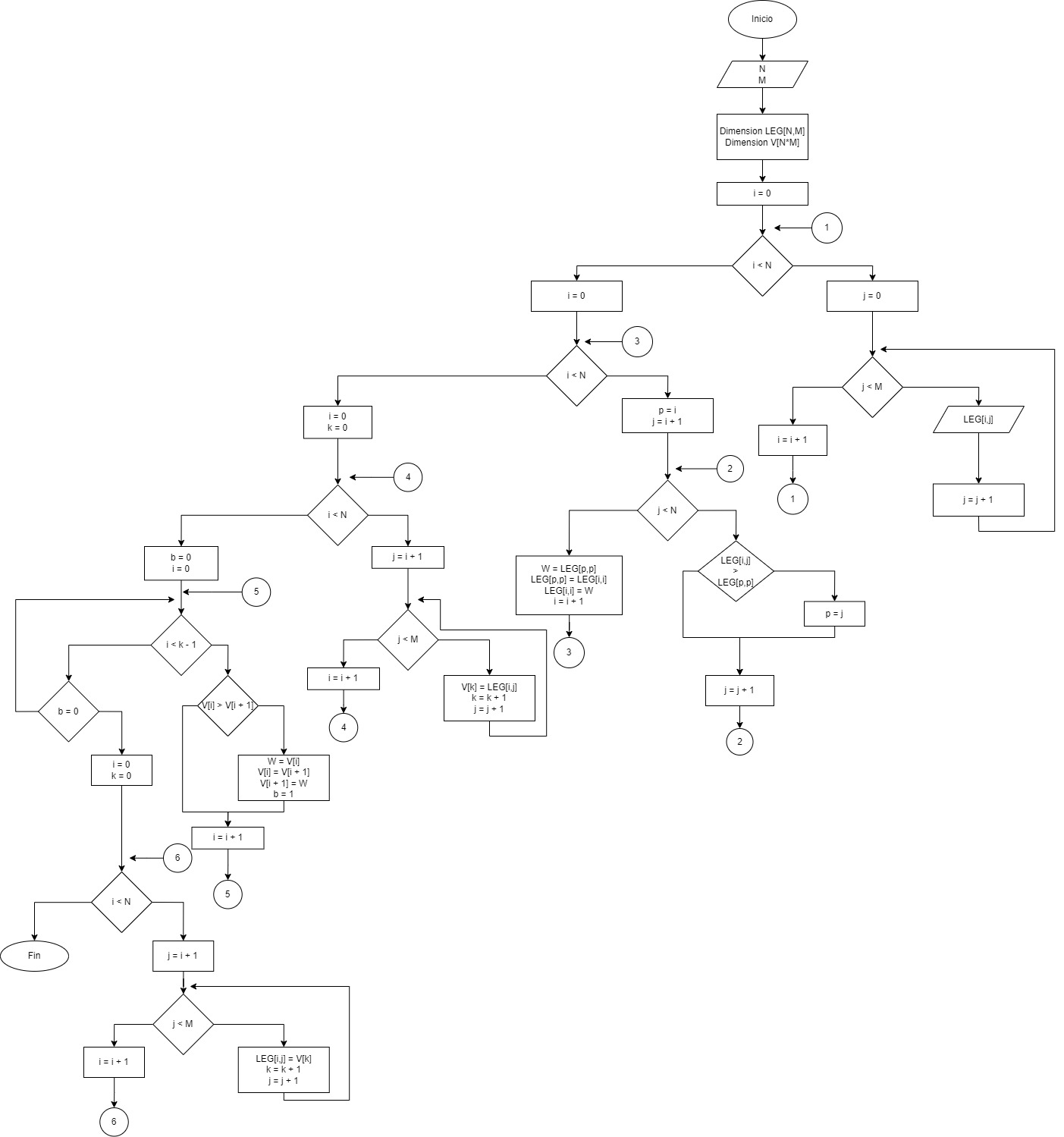
j = j + 1

FinMientras

I = i + 1

FinMientras

Fin



1. Dada la matriz UNSE (Nx4) con los datos de los libros de la Biblioteca de una Universidad: código de libro, código de autor, número de ejemplares disponibles y precio; ordenarla en forma ascendente en función del precio de los libros.

Datos de entrada: Matriz UNSE[N,4]

Datos de salida: Matriz UNSE[N,4] ordenada en forma ascendente en función del precio de los libros

Pseudocodigo:

Inicio

Leer N

Dimension UNSE[N,4]

i = 0

Mientras (i < N) Hacer

j = 0

Mientras (j < 4) Hacer

Leer UNSE[i,j]

j = j + 1

FinMientras

i = i + 1

FinMientras

Repetir

b = 0

i = 0

Mientras (i < N - 1) Hacer

Si (UNSE[i + 1,3] < UNSE[i,3]) Entonces

k = 0

Mientras (k < 4) Hacer

x = UNSE[i,k]

UNSE[i,k] = UNSE[i + 1,k]

UNSE[i + 1,k] = x

k = k + 1

FinMientras

b = 1

FinSi

i = i + 1

FinMientras

Hasta Que b = 0

Fin

