

**FUNDAMENTOS DE LA PROGRAMACIÓN (PUI)****TRABAJO PRÁCTICO 2: ARREGLOS UNIDIMENSIONALES (VECTORES)**

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

- Realizar la búsqueda creativa de solución/es algorítmica/s 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.**

❖ **Recorrido**

5. A partir del vector APELLIDOS de N alumnos de una comisión, mostrar el mensaje correspondiente:

- "Creciente"**, si el listado esta ordenado estrictamente creciente  $A(1) < A(2) < A(3) < \dots < A(N)$ .
- "Decreciente"**, si el listado esta ordenado estrictamente decreciente  $A(1) > A(2) > A(3) > \dots > A(N)$ .
- "Ninguna"**, si no cumple con ninguna de los anteriores.

Pseudocódigo:

Inicio

Definir N,cv,b,b2 Como Entero

Leer N

Dimension EDADES[N]

cv = 0

b = 0

b2 = 0

Mientras (cv < N) Hacer

Leer EDADES[cv]

cv = cv + 1

FinMientras

cv = 1

Mientras (cv < N) Hacer

Si (EDADES[cv - 1] < EDADES[cv]) Entonces

b = 1

SiNo

Si (EDADES[cv - 1] > EDADES[cv]) Entonces

b2 = 1

FinSi

FinSi

cv = cv + 1

FinMientras

Si (b = b2) Entonces

Escribir "Ninguna"

SiNo

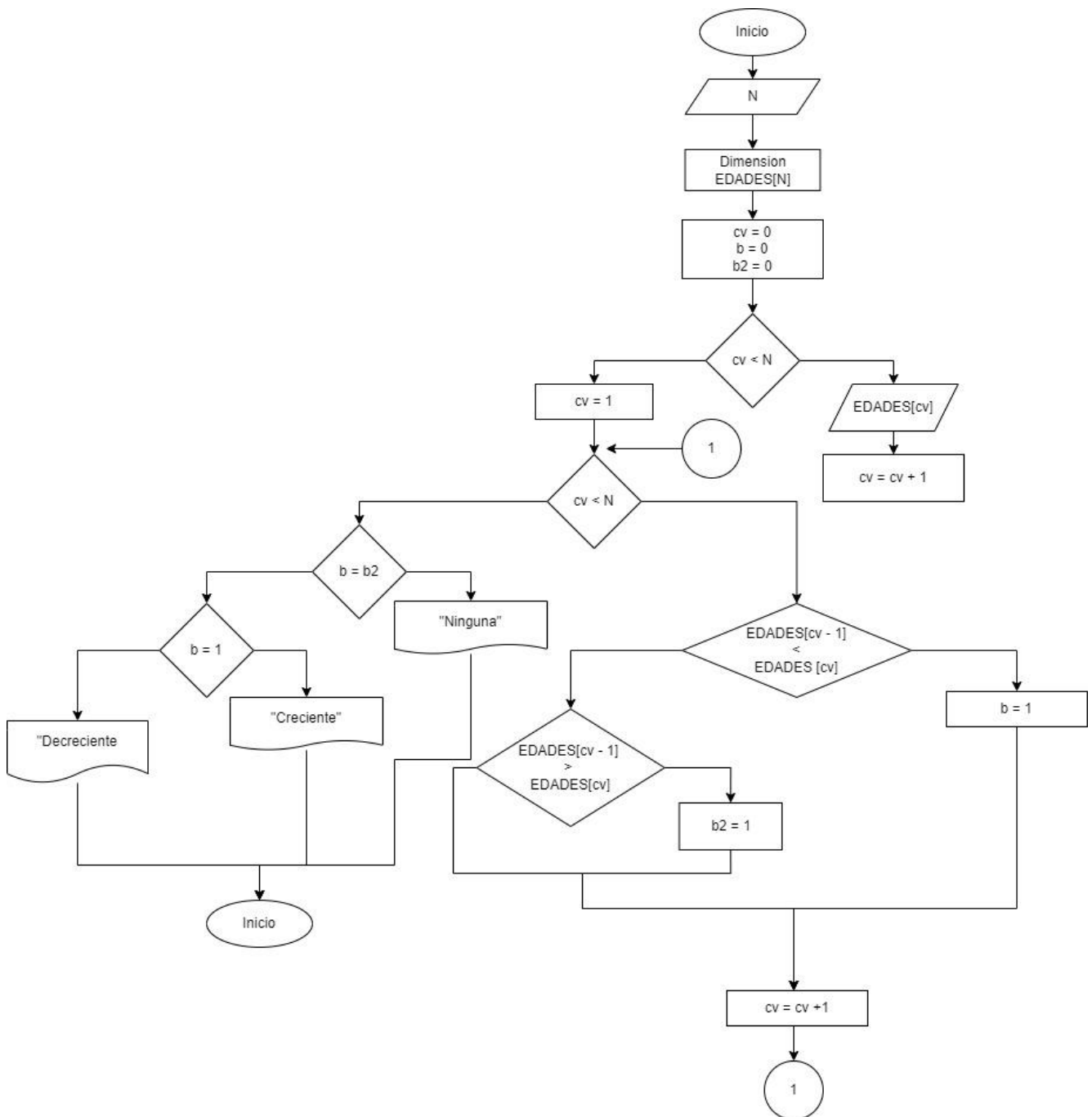
Si (b = 1) Entonces

Escribir "Creciente"

SiNo

FinSi  
FinSi  
Fin

Escribir "Decreciente"



### ❖ **Búsqueda (lineal, binaria)**

2. Dado el vector CANT de 95 elementos, contar y mostrar la cantidad de veces que el vector contiene el número N.

Pseudocódigo:

Inicio

Dimension CANT[95]

CANT[0] = 45; CANT[1] = 40; CANT[2] = 14; CANT[3] = 25; CANT[4] = 64;

CANT[5] = 70; CANT[6] = 76; CANT[7] = 42; CANT[8] = 31;

Escribir "Ingrese el número a buscar"

Leer W

cv = 0

c = 0

Mientras (cv<N) Hacer

    M = trunc((cv+N)/2)

    Si (W = CANT[M]) Entonces

        c = c + 1

        cv = cv + 1

    SiNo

        Si (W<CANT[M]) Entonces

            N = M - 1

            Escribir N

        SiNo

            cv = M + 1

        FinSi

    FinSi

FinMientras

Si(c = 0) Entonces

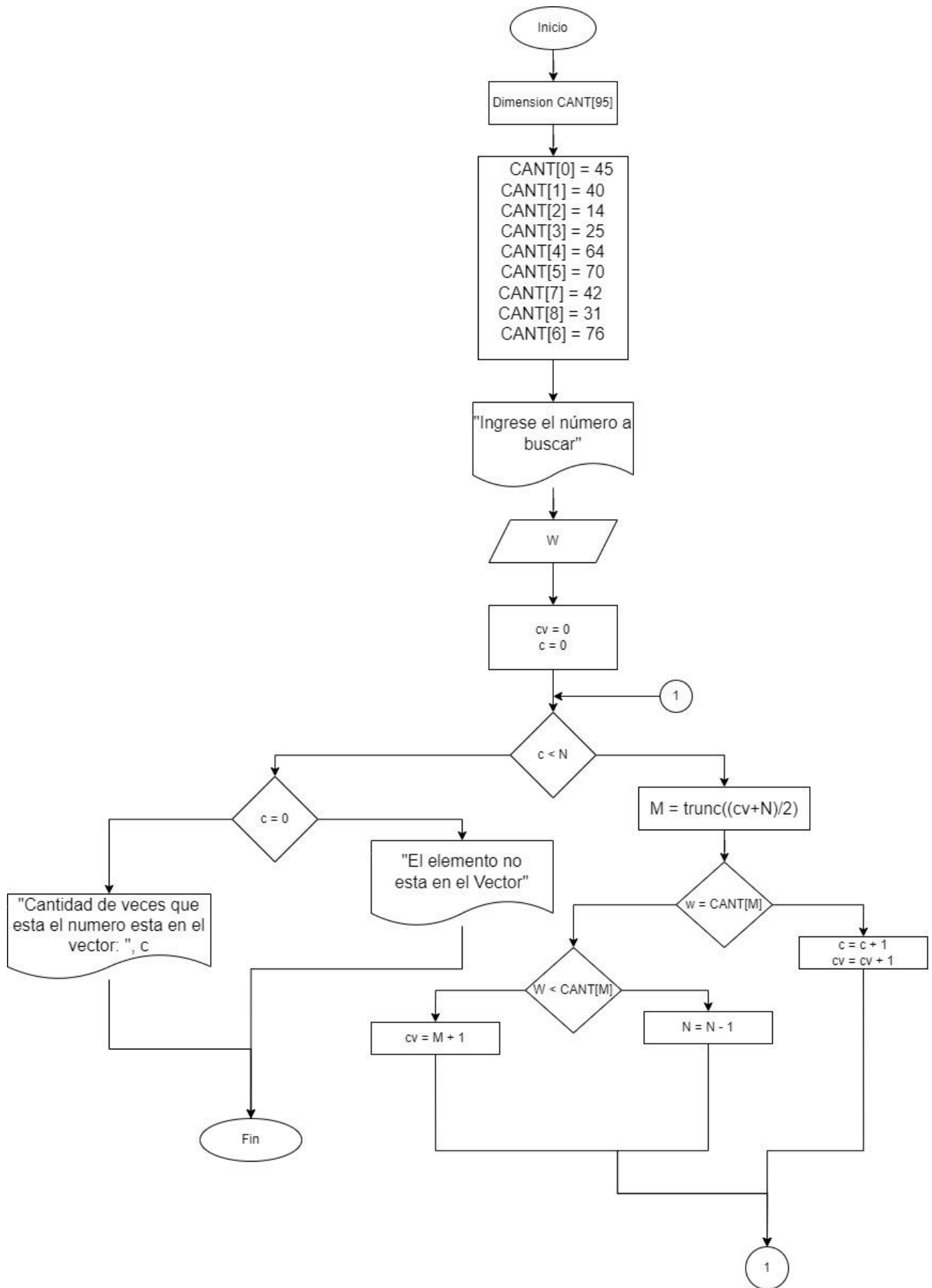
    Escribir "El elemento no esta en el Vector"

SiNo

    Escribir "Cantidad de veces que esta el numero esta en el vector: ", c

FinSi

Fin



### ❖ Actualización (añadir, insertar, eliminar, modificar)

1. Un comercio almacena el precio de la mercadería que posee en su stock, en un vector denominado COSTO, se deberá incrementar en un 10% los valores inferiores a \$100.

Pseudocódigo:

Inicio

Leer N

Dimension COSTO[N]

cv = 0

Mientras (cv < N) Hacer

Leer COSTO[cv]

cv = cv + 1

FinMientras

cv = 0

Mientras (cv < N) Hacer

Si (COSTO[cv] < 100) Entonces

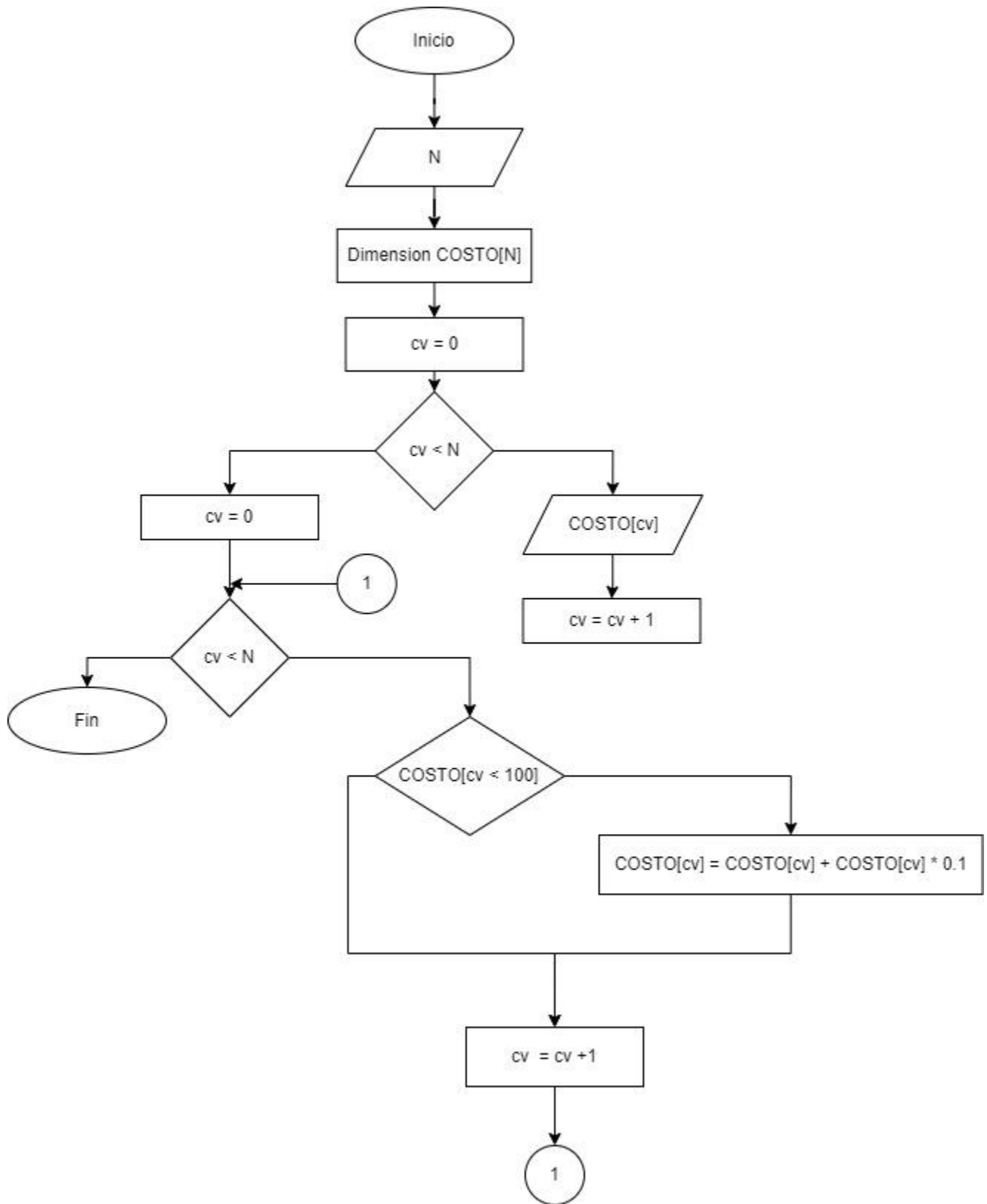
COSTO[cv] = COSTO[cv] + COSTO[cv] \* 0.1

FinSi

cv = cv + 1

FinMientras

Fin



3. El vector A posee 20 elementos, cada uno de 3 dígitos, eliminar aquellos números que en el dígito de las decenas tengan el valor 7.

Ejemplo: **Vector A** original

141	177	693	154	172
-----	-----	-----	-----	-----

Salida esperada: **Vector A**

141	693	154
-----	-----	-----

Pseudocódigo:

Inicio

Dimension A[20]

A[0] = 141; A[1] = 177; A[2] = 693; A[3] = 154; A[4] = 172

cv = 0

Mientras (cv < 20) Hacer

Si (trunc(A[cv] mod 100 / 10) = 7) Entonces

j = cv

Mientras (j < 19) Hacer

A[j] = A[j + 1]

j = j + 1

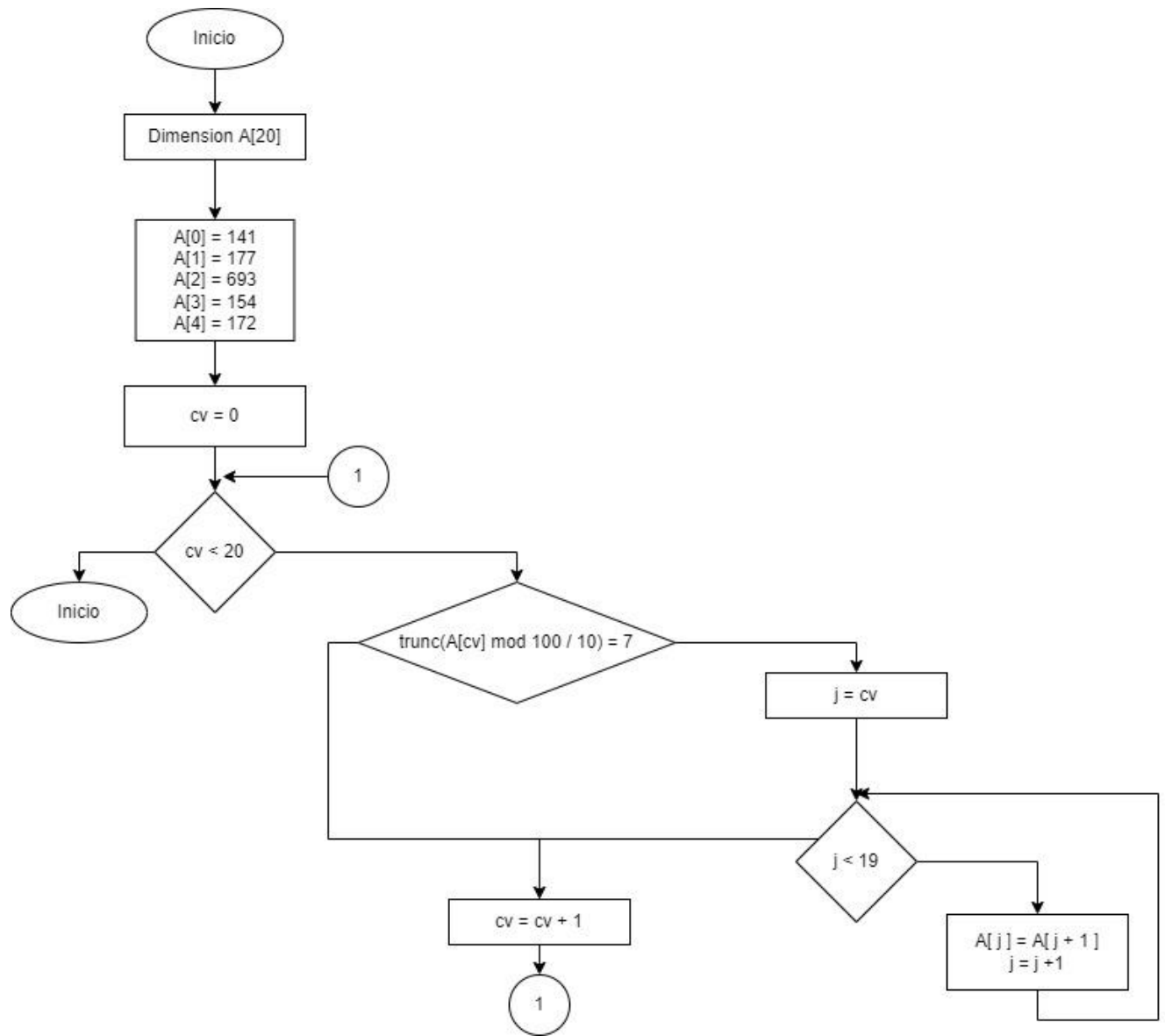
FinMientras

FinSi

cv = cv + 1

FinMientras

Fin





6. Dado el vector A de 20 elementos (ordenado en forma descendente) y el vector B de 15 elementos (desordenado), insertar los elementos del vector B en el vector A; el vector A debe quedar ordenado. Para la solución no debe utilizar ningún método de ordenación.

Psudocódigo:

Inicio

N = 20

Dimension A[N]

Dimension B[15]

cv = 0

A[0] = 98; A[1] = 87; A[2] = 76; A[3] = 72; A[4] = 70; A[5] = 67; A[6] = 61; A[7] = 56;

A[8] = 50; A[9] = 45; A[10] = 41;

B[0] = 44; B[1] = 64; B[2] = 80; B[3] = 43; B[4] = 95; B[5] = 10;

Mientras (cv < 15) Hacer

    j = 0

    band = 0

    Mientras (j < 20 - 1 y band = 0) Hacer

        Si(B[cv] > A[j]) Entonces

            k = N - 1

            Mientras (k >= cv) Hacer

                A[k + 1] = A[k]

                k = k - 1

            FinMientras

            A[j] = B[cv]

            N = N + 1

            band = 1

        SiNo

            j = j + 1

    FinSi

    FinMientras

    cv = cv + 1

FinMientras

Si (band = 0) Entonces

    A[j] = B[cv]

    N = N + 1

FinSi

Fin

