



# ELEMENTOS DE PROGRAMACION

# VECTORES

FUNCIONES  
POR VALOR O COPIA

(INTRODUCCION)

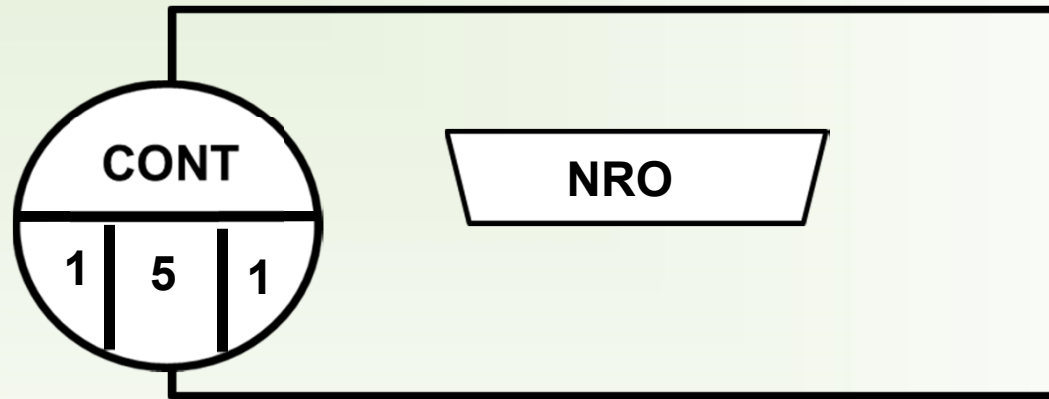
## PROBLEMA A RESOLVER

Se dispone de 5 números enteros.

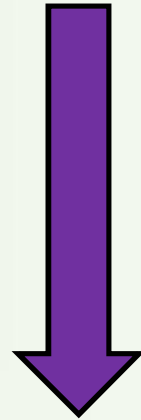
Informar cuales números son  
mayores al último número  
ingresado.

**COMO LO RESUELVO ??????**

# PUEDO USAR for?????



**5** NUMEROS ENTEROS



**5 variables ????**

**N1**

**20**

**N2**

**84**

**N3**

**15**

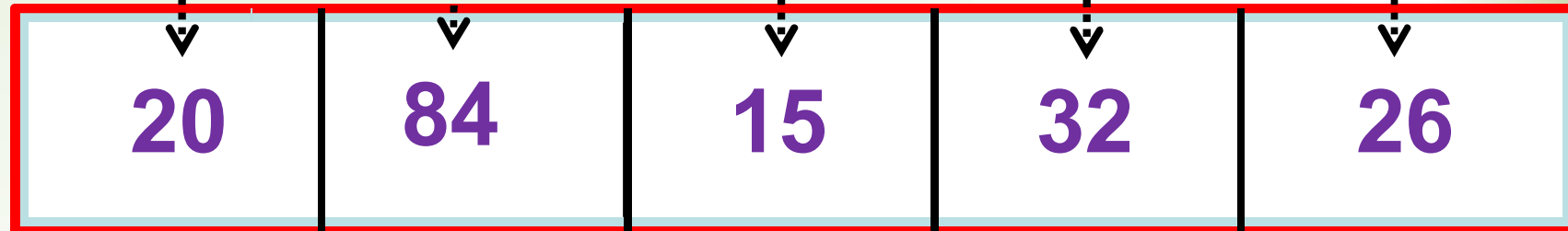
**N4**

**32**

**N5**

**26**

**SE ORGANIZAN LOS NUMEROS ENTEROS  
EN FORMA SECUENCIAL**



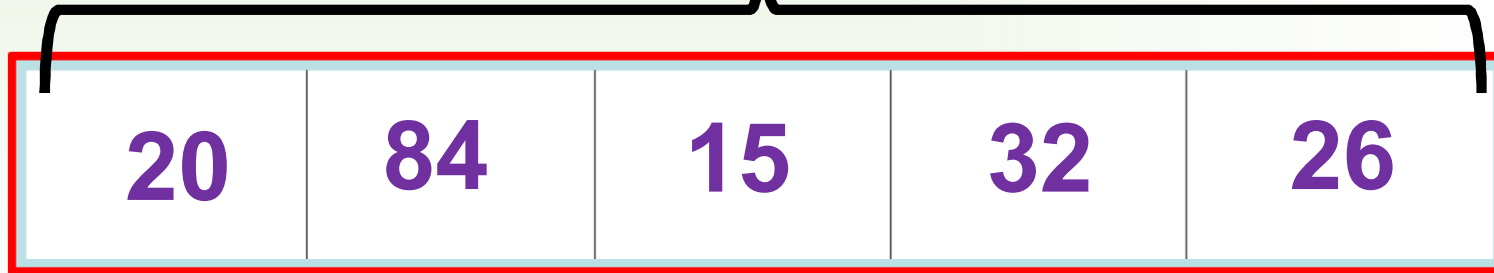
**EN UNA ESPECIE DE CAJA (se guardan)**

**CADA NUMERO EN UNA POSICION O  
UBICACION.**

# Vector (declaración)

Tipo de DATO **IDENTIFICADOR** **[CANTIDAD DE ELEMENTOS]** ;

**int VEC [5]**

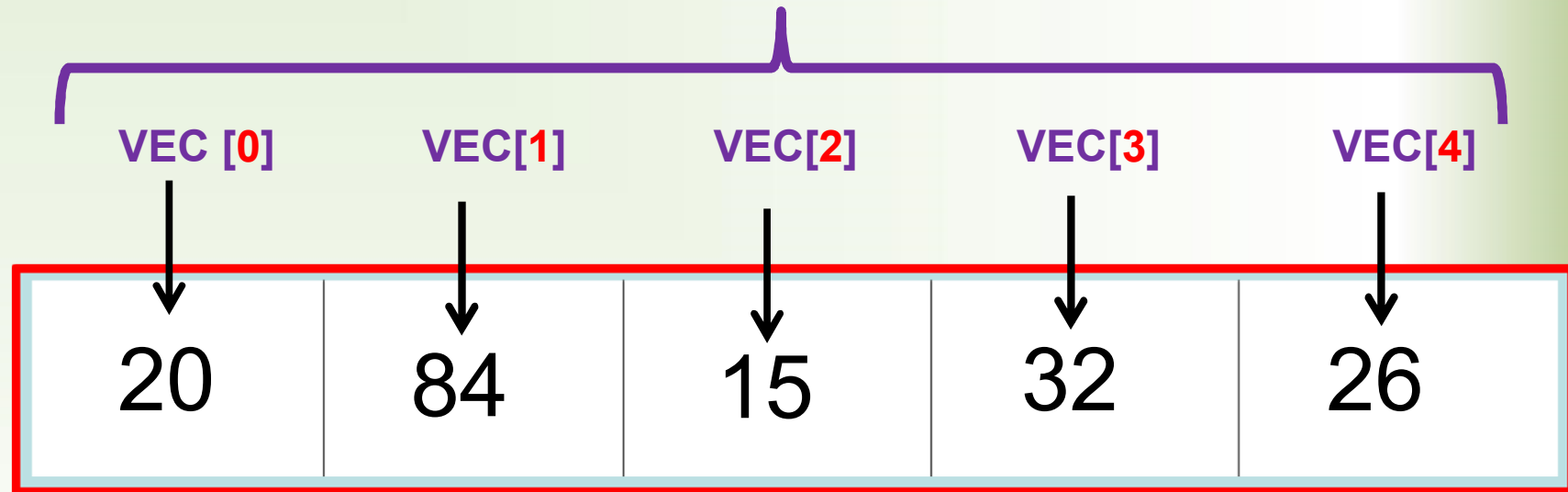


## CARACTERISTICA:

- TODO VECTOR TIENE EL MISMO TIPO DE DATO
- LA INFORMACION SE ORGANIZA EN FORMA CONTIGUA EN MEMORIA.

# Vector (declaración)

**ELEMENTOS O COMPONENTES DEL VECTOR**



1 2 3 4 5  
POSICION – INDICE- UBICACIÓN **vista por el usuario**

|   |   |   |   |   |
|---|---|---|---|---|
| 0   | 1 | 2 | 3 | 4 |
| <b>POSICION – INDICE- UBICACIÓN en LENGUAJE C</b> |   |   |   |   |



# Declaración e Inicialización de vectores

```
int V1 [];
```

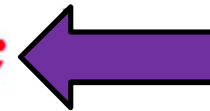


MAL

```
int V2 []={23,45,89,14};
```

```
int V3 [4]={23,45,89,14};
```

```
int V4 [4]={23,67,-1,78,56,100};
```



MAL

```
int V5 [4]={0};
```

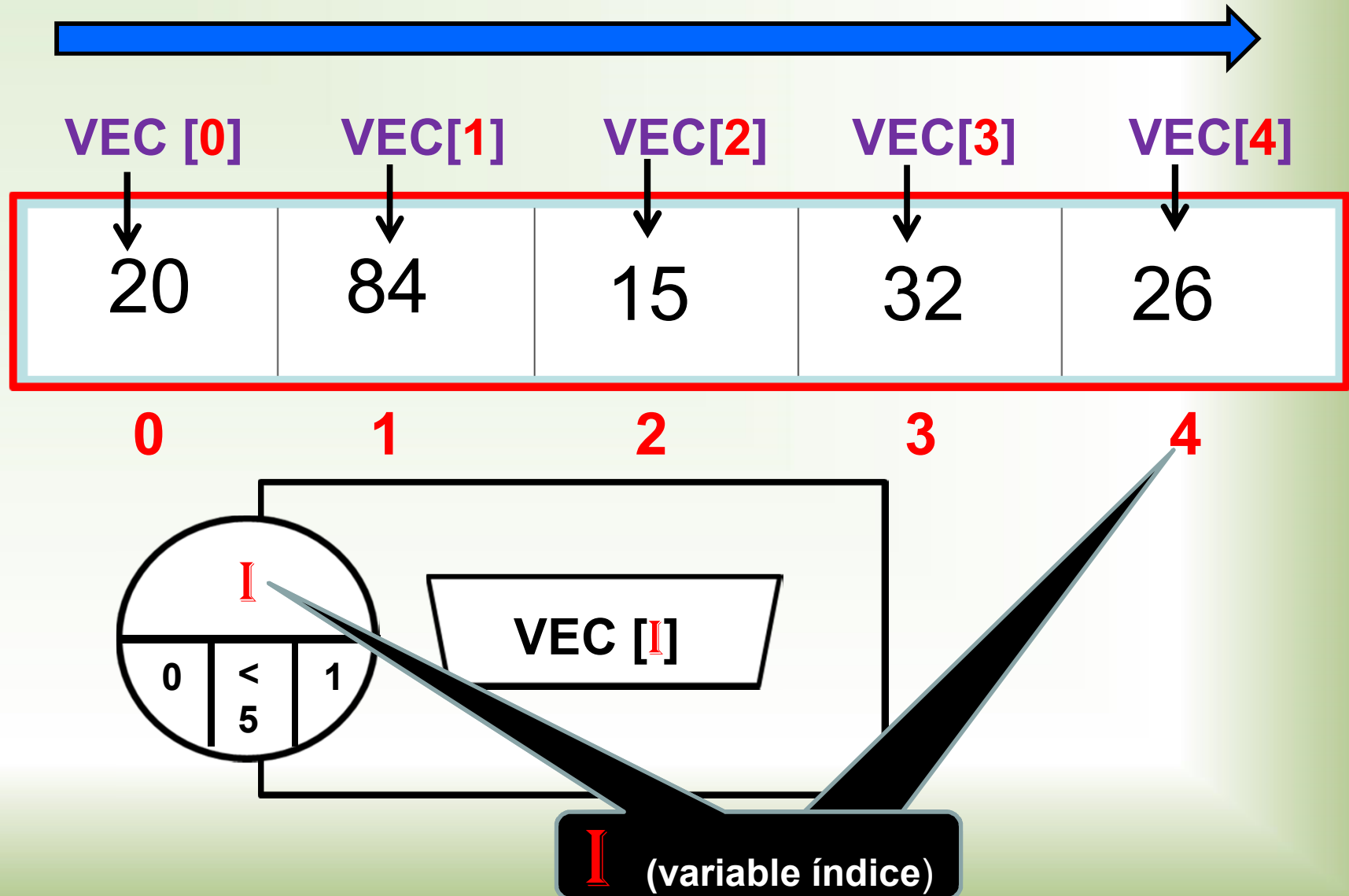
```
int V6 [4]={65};
```

```
int V7 [4]={45,89};
```

```
char V8 [4]='A','B','C','D';
```

-----

# CARGA SECUENCIAL



**Declaración del vector**

**Carga secuencial del vector**

**Informe del vector**

```
#include<stdio.h>
```

```
main()
```

```
{
```

```
    int VEC[5],I;
```

```
    printf ("\n CARGA SECUENCIAL DEL VECTOR");
```

```
    for(I=0;I<5;I++)
```

```
    {
```

```
        printf("\nINGRESAR ELEMENTO EN POSICION %d  ",I+1);
```

```
        scanf("%d",&VEC[I]);
```

```
    }
```

```
    printf("\n DIRECCION POSICION INDICE  ELEMENTO");
```

```
    for(I=0;I<5;I++)
```

```
        printf("\n %x          %d  \t%d  |%4d  ",&VEC[I],I+1, I, VEC[I]);
```

```
}
```

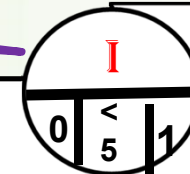
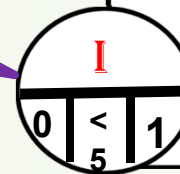
```
////////////////////////////////////
```

PP

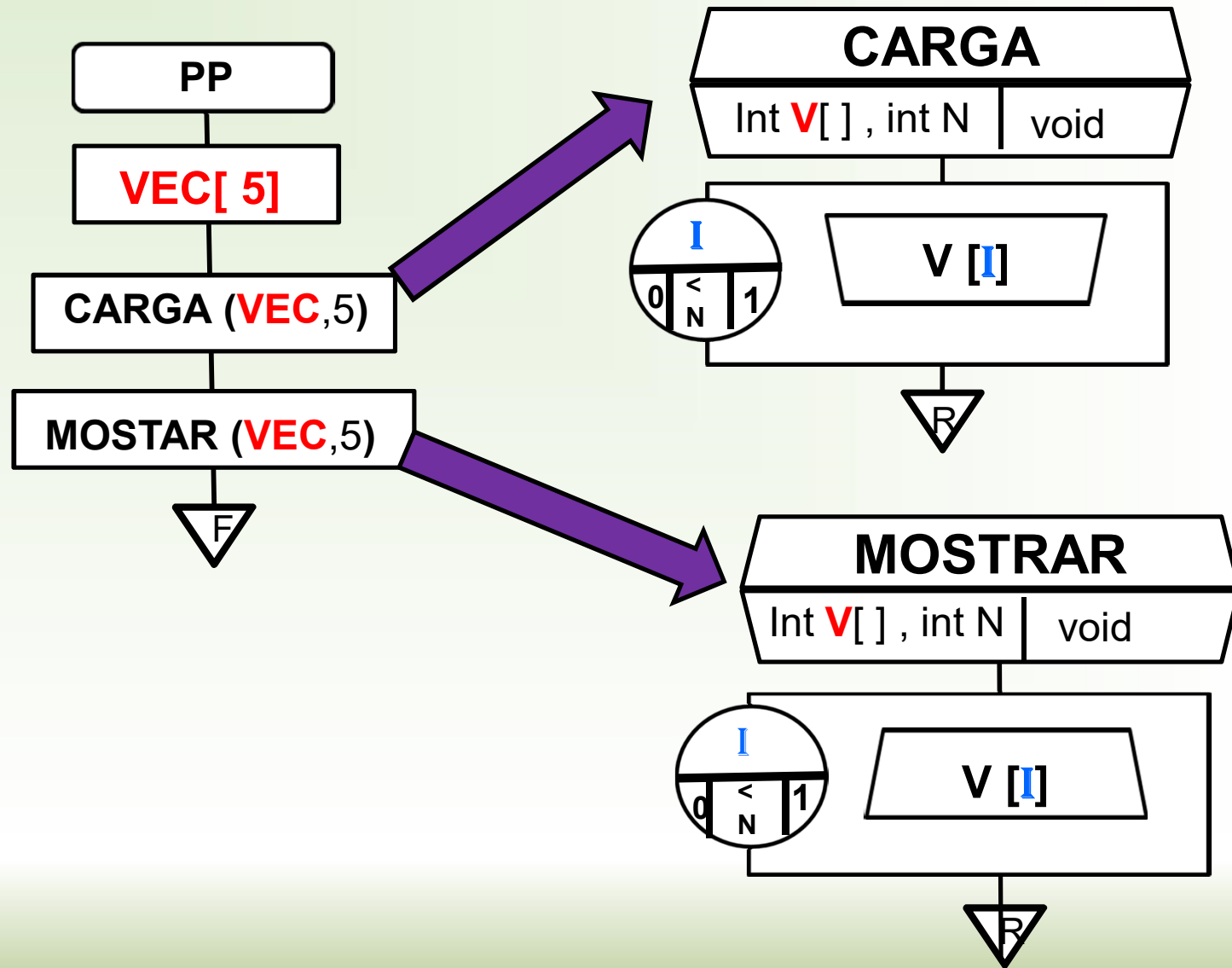
**VEC[ 5]**

**VEC [I]**

**VEC [I]**



# FUNCIONES CON VECTORES



```

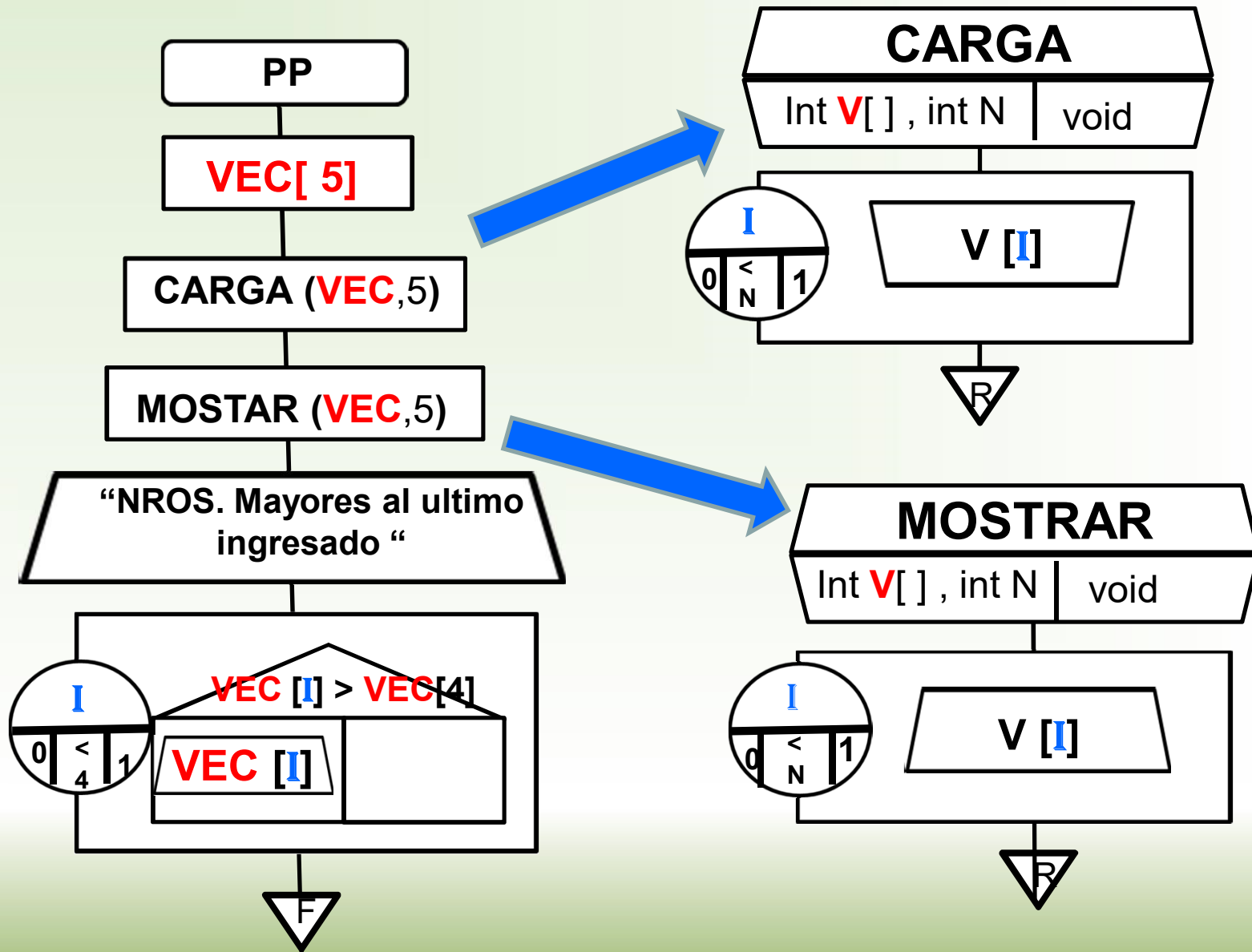
#include<stdio.h>
void CARGA(int [],int);
void MOSTRAR(int [],int);
main()
{ int VEC[5],I;
  CARGA(VEC,5);
  MOSTRAR(VEC,5);
}

////////////////////////////////////
void CARGA(int V[],int N)
{ int I;
  for(I=0;I<N;I++)
  {
    printf("\nINGRESAR ELEMENTO EN POSICION %d ", I+1);
    scanf("%d",&V[I]);
  }
}

////////////////////////////////////
void MOSTRAR(int V[],int N)
{ int I;
  printf("\n POSICION ELEMENTO");
  for(I=0;I<N;I++)
    printf("\n %d \t %4d ",I+1, V[I]);
}

```

Se dispone de 5 números enteros. Informar cuales números son mayores al último número ingresado



```

#include<stdio.h>
void CARGA(int [],int);
void MOSTRAR(int [],int);
main()
{
    int VEC[5],I;
    CARGA(VEC,5);
    MOSTRAR(VEC,5);
    printf("\nNUMEROS MAYORES AL ULTIMO INGRESADO ");
    for (I=0;I<4;I++)
        if (VEC[I] > VEC[4])
            printf("\n\t %d",VEC[I]);
}

////////////////////////////////////
void CARGA(int V[],int N)
{
    int I;
    for(I=0;I<N;I++)
    {
        printf("\nINGRESAR ELEMENTO EN POSICION %d ",I+1);
        scanf("%d",&V[I]);
    }
}

////////////////////////////////////
void MOSTRAR(int V[],int N)
{
    int I;
    printf("\n POSICION      ELEMENTO");
    for(I=0;I<N;I++)
        printf("\n      %d          %d",I+1,V[I]);
}

```

# EJEMPLO

**SE DISPONE DE UNA LISTA DE 10 VALORES ENTEROS**

**A-)** CARGAR LOS DATOS EN UN VECTOR LLAMADO VA

**B-)** COPIAR VA EN VB

**C-)** SUMAR LOS ELEMENTOS DEL VECTOR VA

**D-)** VALOR PROMEDIO DEL VECTOR VA

**E-)** CANTIDAD DE CEROS CONTENIDOS EN EL VECTOR VA

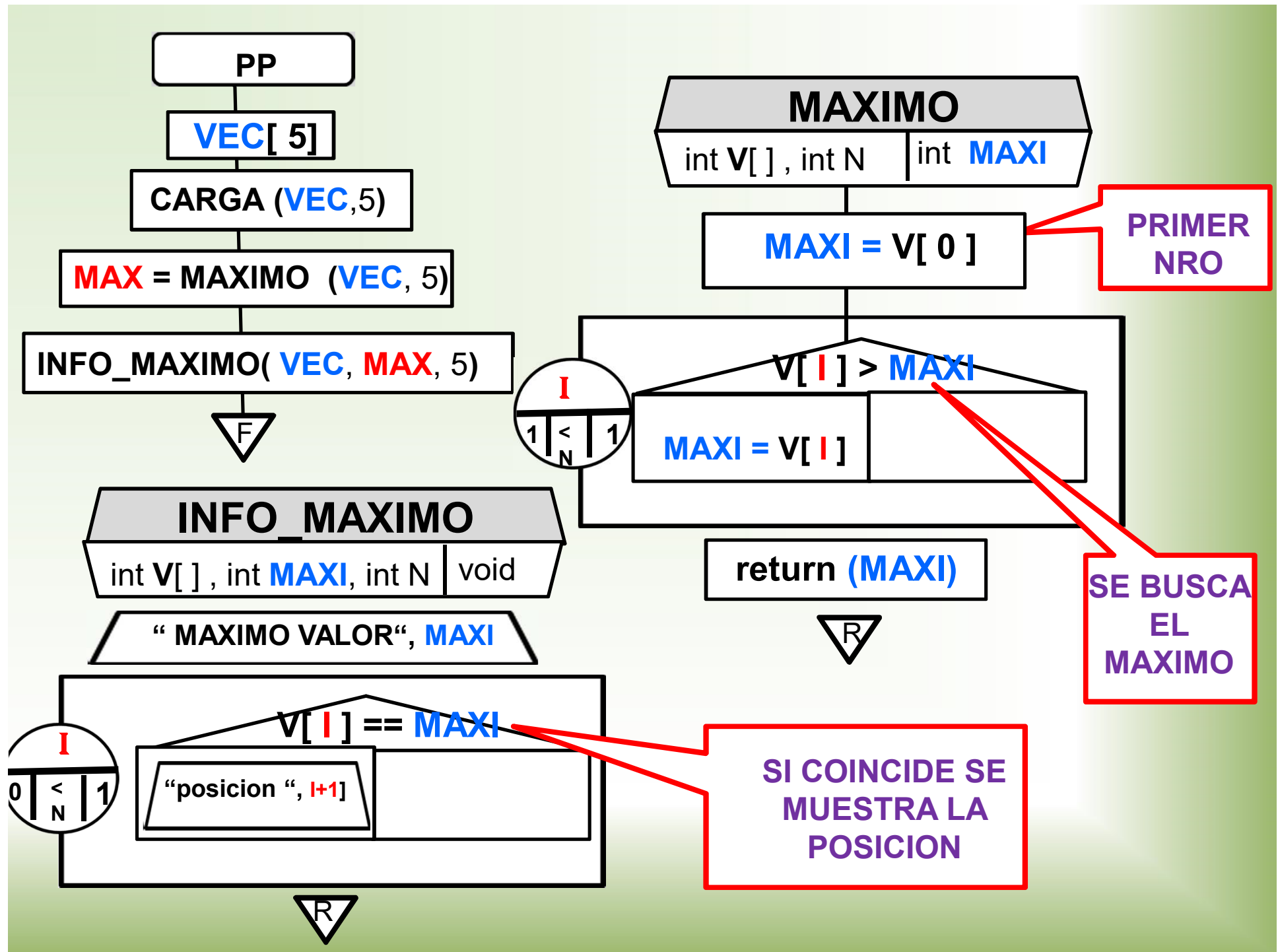


# VECTORES

(MAXIMO O MINIMO)

Se dispone de una lista de 5  
números. (los nros se pueden repetir)

INFORMAR EL MAYOR NRO  
INGRESADO y EN QUE  
POSICION/es SE ENCUENTRA/n.



# EJERCICIO 1

Dada una lista de 10 valores enteros. Se pide:

- a-)** Cargar la lista en memoria en un vector llamado A.
- b-)** Copiar este vector en otro llamado B.
- c-)** Generar un vector C correspondiente a la suma de A y B.
- d-)** Copiar A en orden inverso en otro vector llamado D.
- e-)** Listar los cuatro vectores simultáneamente informando en la primer columna el número de orden de los elementos.
- f-)** Posiciones de elementos pares del vector A.
- g-)** Elementos de posiciones impares del vector A.
- h-)** La suma de los elementos del vector A.
- i-)** Valor promedio del vector A.
- j-)** Cantidad de elementos iguales a 32 del vector A.

# EJERCICIO 2

Dada una lista de 10 valores enteros. Se pide

- a) Cargar los datos en memoria en un vector llamado "A"
- b) Asignar al quinto elemento del vector el valor 8.
- c) Informar el producto entre el elemento ubicado en la 4ta posición del vector "A" por su subíndice.
- d) Intercambiar el tercer elemento del vector con el octavo.
- e) Informar la cantidad de elementos pares del vector.
- f) Informar la suma de los elementos de orden de ingreso par (ubicados en pos. 1,3, 5....)
- g) Realizar la suma de los últimos 5 elementos del vector
- h) Generar otro vector "B" donde cada elemento sea el contenido del original \* 2
- i) Generar otro vector "C" que sea la suma entre "A" y "B"
- j) Generar el vector "D", que sea "A" permutado.
- k) Generar otro vector con los elementos impares de "A"
- l) Informar la cantidad de veces que existe el número 8 en el vector "A"