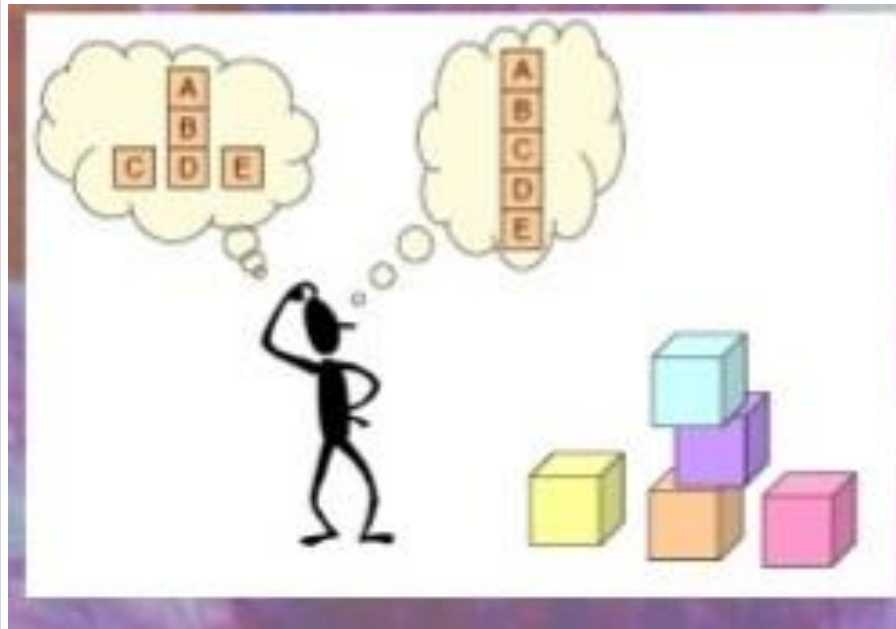


# ORDENAMIENTO



El ordenar un grupo de datos significa mover los datos o sus referencias para que queden en una secuencia tal que represente un orden, el cual puede ser numérico, alfabético o incluso alfanumérico, ascendente o descendente.

- ❖ **EL PROPÓSITO PRINCIPAL DE UN ORDENAMIENTO ES EL DE FACILITAR LAS BÚSQUEDAS DE ELEMENTOS DEL CONJUNTO ORDENADO.**

## EJEMPLO

**ORDENAR UN CONJUNTO DE NÚMEROS EN ORDEN CRECIENTE**

**ORDENAR UNA LISTA DE NOMBRES POR ORDEN ALFABÉTICO**

# ORDENAMIENTO

❖ LOS MÉTODOS DE ORDENAMIENTO SE DIVIDEN EN DOS CATEGORÍAS:

➤ **ORDENAMIENTO DE ARREGLOS (ORDENACIÓN INTERNA):**

✓ SE ALMACENA EN LA MEMORIA INTERNA DE LA COMPUTADORA DE GRAN VELOCIDAD Y ACCESO ALEATORIO.

➤ **ORDENAMIENTO DE ARCHIVOS (ORDENACIÓN EXTERNA):**

✓ SE SUELE HACER CASI SIEMPRE SOBRE SOPORTES DE ALMACENAMIENTO EXTERNO, DISCOS, ETC.

✓ ESTOS DISPOSITIVOS SON MÁS LENTOS EN LAS OPERACIONES DE ENTRADA/SALIDA, PERO, POR EL CONTRARIO, PUEDEN CONTENER MAYOR CANTIDAD DE INFORMACIÓN.

***ORDENAMIENTO INTERNA:*** CLASIFICACIÓN DE LOS VALORES DE UN VECTOR SEGÚN UN ORDEN EN MEMORIA CENTRAL: RÁPIDA.

***ORDENAMIENTO EXTERNA:*** CLASIFICACIÓN DE LOS REGISTROS DE UN ARCHIVO SITUADO EN UN SOPORTE EXTERNO: MENOS RÁPIDO.

## ❖ **ALGORITMO DE ORDENAMIENTO POR MÉTODO DE SELECCIÓN**

**PARA ORDENAR LOS ELEMENTOS DE UN VECTOR DE MENOR A MAYOR/ ASCENDENTE**

**A**

<b>3</b>	<b>5</b>	<b>2</b>	<b>1</b>	<b>4</b>
----------	----------	----------	----------	----------

**EL ALGORITMO SE BASA EN BUSCAR EL ELEMENTO DE MENOR VALOR DEL VECTOR Y COLOCARLO EN LA PRIMERA POSICIÓN; LUEGO SE BUSCA EL SEGUNDO ELEMENTO DE MENOR VALOR Y SE COLOCA EN LA SEGUNDA POSICIÓN, Y ASÍ SUCESIVAMENTE.**

**EL PROCEDIMIENTO DEL ALGORITMO ES :**

- 1. SELECCIONAR EL ELEMENTO DE MENOR VALOR EN UN VECTOR DE N ELEMENTOS.**
- 2. INTERCAMBIAR DICHO ELEMENTO CON EL PRIMER ELEMENTO DEL VECTOR.**
- 3. REPETIR LAS OPERACIONES CON LOS N-1 ELEMENTOS RESTANTES, SELECCIONANDO EL SEGUNDO ELEMENTO DE MENOR VALOR; CONTINUAR CON LOS N-2 ELEMENTOS RESTANTES HASTA QUE SÓLO QUEDE EL MAYOR.**

**A**

<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
----------	----------	----------	----------	----------

# ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE SELECCIÓN

## A VECTOR DESORDENADO

<b>3</b>	<b>5</b>	<b>2</b>	<b>1</b>	<b>4</b>
----------	----------	----------	----------	----------

1. Encontrar el elemento de menor valor del vector e intercambiar con el primer elemento del vector.

<b>3</b>	<b>5</b>	<b>2</b>	<b>1</b>	<b>4</b>
----------	----------	----------	----------	----------



<b>1</b>	<b>5</b>	<b>2</b>	<b>3</b>	<b>4</b>
----------	----------	----------	----------	----------

INTERCAMBIA A(0) CON A(3)

2. En las posiciones 2 a 5, encontrar el elemento de menor valor e intercambiar con el segundo elemento del vector.

<b>1</b>	<b>5</b>	<b>2</b>	<b>3</b>	<b>4</b>
----------	----------	----------	----------	----------



<b>1</b>	<b>2</b>	<b>5</b>	<b>3</b>	<b>4</b>
----------	----------	----------	----------	----------

INTERCAMBIA A(1) CON A(2)

3. En las posiciones 3 a 5, encontrar el elemento de menor valor e intercambiar con el tercer elemento del vector.

<b>1</b>	<b>2</b>	<b>5</b>	<b>3</b>	<b>4</b>
----------	----------	----------	----------	----------



<b>1</b>	<b>2</b>	<b>3</b>	<b>5</b>	<b>4</b>
----------	----------	----------	----------	----------

INTERCAMBIA A(2) CON A(3)

4. En las posiciones 4 a 5, encontrar el elemento de menor valor e intercambiar con el cuarto elemento del vector.

<b>1</b>	<b>2</b>	<b>3</b>	<b>5</b>	<b>4</b>
----------	----------	----------	----------	----------



<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
----------	----------	----------	----------	----------

INTERCAMBIA A(3) CON A(4)

VECTOR A ORDENADO

# ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE SELECCIÓN

$W = 0$

$N = 5$

$I = 0$

MIENTRAS  $I < N-1$

$P = I$

$J = I + 1$

MIENTRAS  $J \leq N-1$

SI  $A(J) < A(P)$

ENTONCES

$P = J$

FIN SI

$J = J + 1$

FIN MIENTRAS

$W = A(P)$

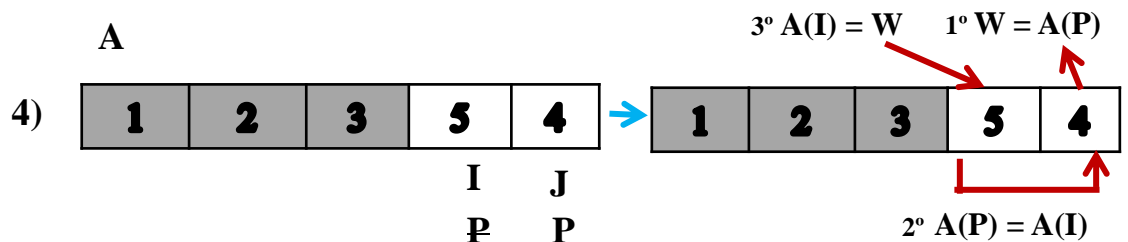
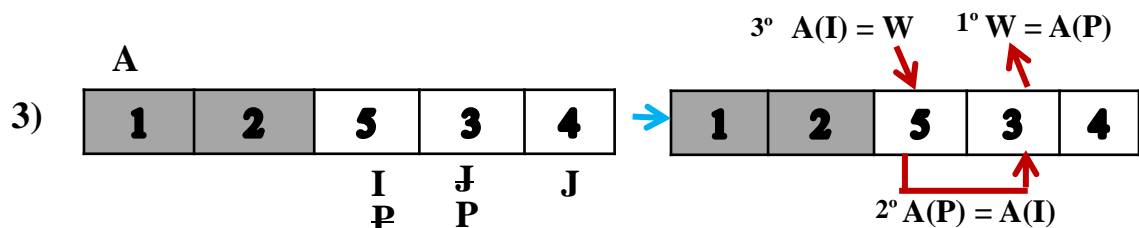
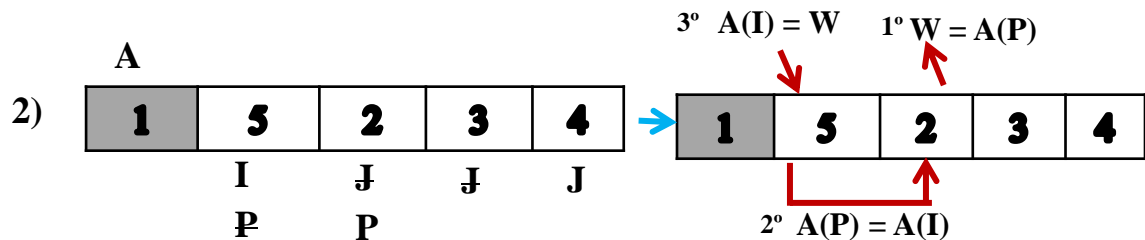
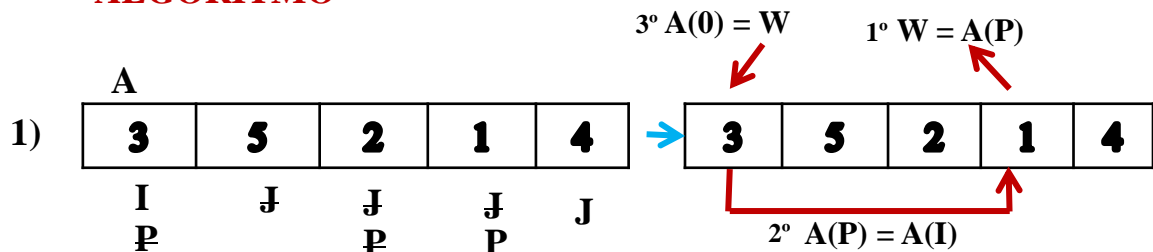
$A(P) = A(I)$

$A(I) = W$

$I = I + 1$

FIN MIENTRAS

## PRUEBA DEL ALGORITMO



VECTOR  
ORDENADO

A				
1	2	3	4	5

## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE SELECCIÓN

**W = 0**

**N = 5**

**I = 0**

**MIENTRAS I < N-1**

**P = I**

**J = I + 1**

**MIENTRAS J <= N**

**SI A(J) < A(P)**

**ENTONCES**

**P = J**

**FIN SI**

**J = J + 1**

**FIN MIENTRAS**

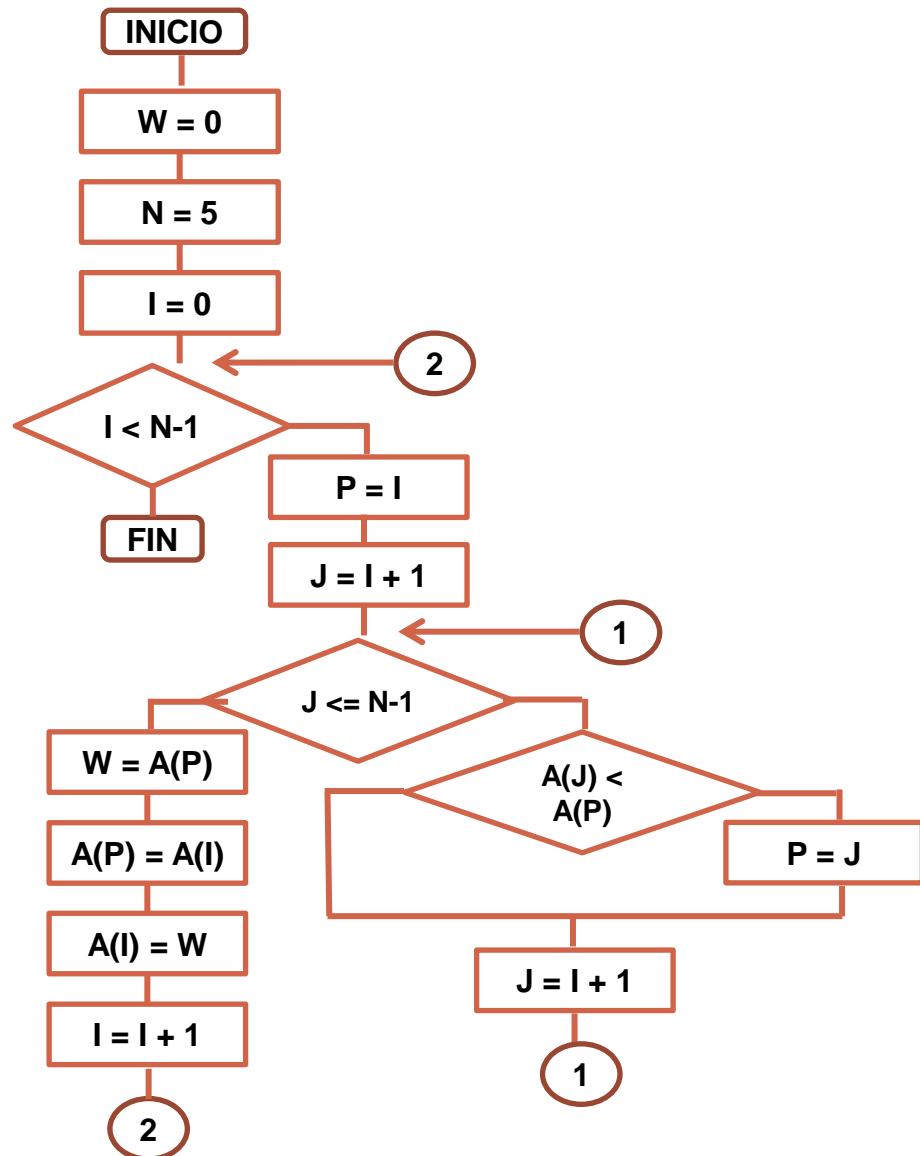
**W = A(P)**

**A(P) = A(I)**

**A(I) = W**

**I = I + 1**

**FIN MIENTRAS**



## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE SELECCIÓN

**PARA ORDENAR LOS ELEMENTOS DE UN VECTOR DE MAYOR A MENOR**

**A**

<b>3</b>	<b>5</b>	<b>2</b>	<b>1</b>	<b>4</b>
----------	----------	----------	----------	----------

EL ALGORITMO SE BASA EN BUSCAR EL ELEMENTO DE MAYOR VALOR DEL VECTOR Y COLOCARLO EN LA PRIMERA POSICIÓN; LUEGO SE BUSCA EL SEGUNDO ELEMENTO DE MAYOR VALOR Y SE COLOCA EN LA SEGUNDA POSICIÓN, Y ASÍ SUCESIVAMENTE.

**EL PROCEDIMIENTO DEL ALGORITMO ES :**

**1. SELECCIONAR EL ELEMENTO DE MAYOR VALOR DE UN VECTOR DE N ELEMENTOS.**

**2. INTERCAMBIAR DICHO ELEMENTO CON EL PRIMERO.**

**3. REPETIR LAS OPERACIONES CON LOS N-1 ELEMENTOS RESTANTES, SELECCIONANDO EL SEGUNDO ELEMENTO DE MAYOR VALOR; CONTINUAR CON LOS N-2 ELEMENTOS RESTANTES HASTA QUE SÓLO QUEDE EL MAYOR.**

**A**

<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
----------	----------	----------	----------	----------

## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE SELECCIÓN

$W = 0$

$N = 5$

$I = 0$

**MIENTRAS  $I < N-1$**

$P = I$

$J = I + 1$

**MIENTRAS  $J \leq N-1$**

**SI  $A(J) > A(P)$**

**ENTONCES**

$P = J$

**FIN SI**

$J = J + 1$

**FIN MIENTRAS**

$W = A(P)$

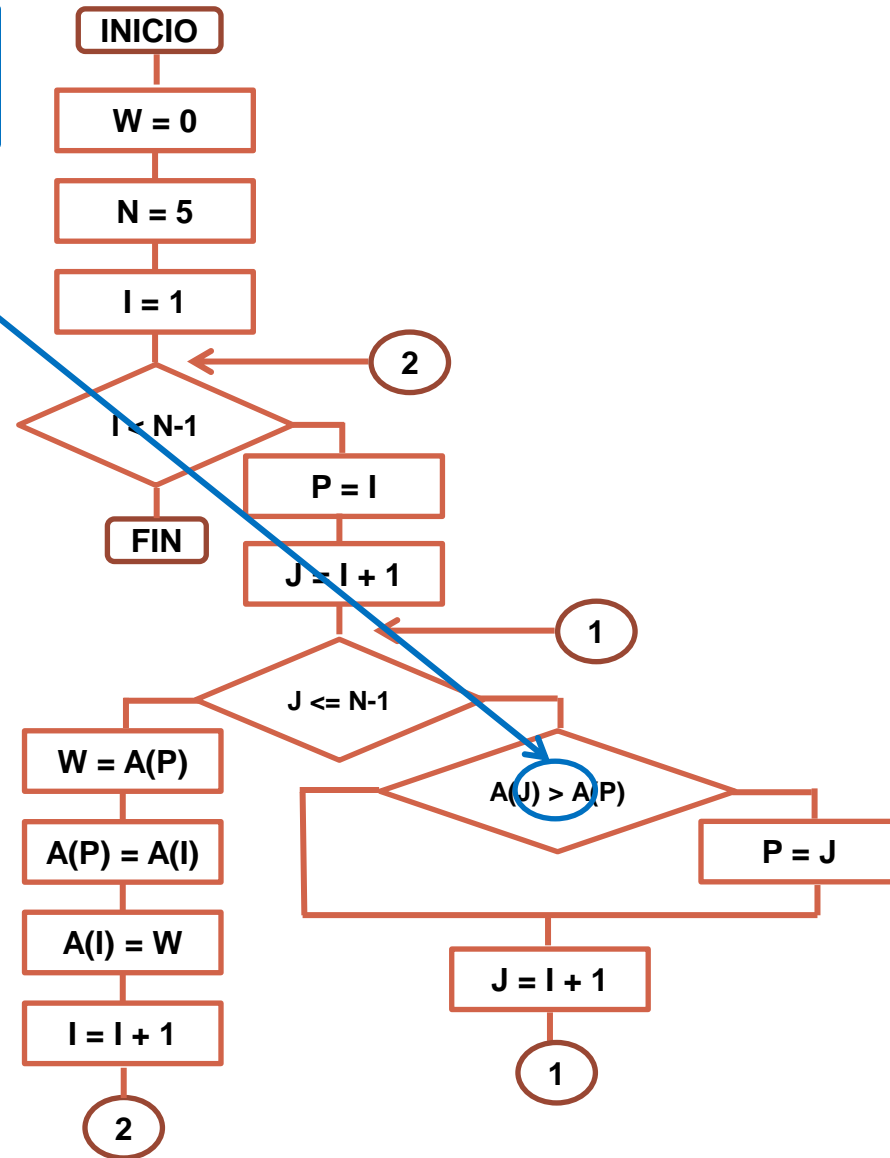
$A(P) = A(I)$

$A(I) = W$

$I = I + 1$

**FIN MIENTRAS**

Para ordenar de MAYOR a MENOR el algoritmo cambia  $A(J) < A(P)$  por  $A(J) > A(P)$





## ❖ **ALGORITMO DE ORDENAMIENTO POR MÉTODO DE BURBUJA**

**EL ORDENAMIENTO DE BURBUJA (BUBBLE SORT EN INGLÉS) ES UN SENCILLO ALGORITMO DE ORDENAMIENTO.**

FUNCIONA REVISANDO CADA ELEMENTO DEL VECTOR CON EL SIGUIENTE O ADYACENTE, INTERCAMBIÁNDOLOS DE POSICIÓN SI ESTÁN EN EL ORDEN EQUIVOCADO. ES NECESARIO RECORRER VARIAS VECES TODO EL VECTOR HASTA QUE NO SE NECESITEN MÁS INTERCAMBIOS, LO CUAL SIGNIFICA QUE EL VECTOR ESTÁ ORDENADO.

**EL PROCEDIMIENTO DEL ALGORITMO ES (Ordenamiento menor a mayor/ Ascendente):**

- 1. EN EL PRIMER RECORRIDO SE COMPARAN LOS ELEMENTOS ADYACENTES Y LOS INTERCAMBIA SI ESTÁN DESORDENADOS.**
- 2. AL FINAL DEL PRIMER RECORRIDO, SE HABRÁN COMPARADO  $N-1$  PARES DE ELEMENTOS, Y EL ELEMENTO DE MAYOR VALOR HABRÁ SIDO ARRASTRADO, COMO UNA BURBUJA, HACIA EL FINAL DEL VECTOR.**
- 3. AL FINAL DEL SEGUNDO RECORRIDO, EL SEGUNDO ELEMENTO DE MAYOR VALOR HABRÁ SIDO ARRASTRADO, COMO UNA BURBUJA, HACIA LA POSICIÓN  $N-1$ .**

## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE BUBUJA

A

15	67	8	16	44	27	12	35
----	----	---	----	----	----	----	----

Ordenar el vector A en forma ascendente

PRUEBA DEL  
ALGORITMO

Las comparaciones que se realizan en el primer recorrido son:

$A[1] > A[2]$  (15 > 67) no hay intercambio

$A[2] > A[3]$  (67 > 08) sí hay intercambio

$A[3] > A[4]$  (67 > 16) sí hay intercambio

$A[4] > A[5]$  (67 > 44) sí hay intercambio

$A[5] > A[6]$  (67 > 27) sí hay intercambio

$A[6] > A[7]$  (67 > 12) sí hay intercambio

$A[7] > A[8]$  (67 > 35) sí hay intercambio

15	67	8	16	44	27	12	35
----	----	---	----	----	----	----	----

15	67	8	16	44	27	12	35
15	8	67	16	44	27	12	35

15	8	67	16	44	27	12	35
15	8	16	67	44	27	12	35

15	8	16	67	44	27	12	35
15	8	16	44	67	27	12	35

15	8	16	44	67	27	12	35
15	8	16	44	27	67	12	35

15	8	16	44	27	67	12	35
15	8	16	44	27	12	67	35

15	8	16	44	27	12	67	35
15	8	16	44	27	12	35	67

Al terminar las comparaciones en el primer recorrido, el elemento de mayor valor, en este caso 67, fue colocado en la última posición del arreglo.

## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE BUBUJA

Las comparaciones que se realizan en el segundo recorrido son:

$A[1] > A[2]$  (15 > 08)      sí hay intercambio

15	8	16	44	27	12	35	67
8	15	16	44	27	12	35	67

$A[2] > A[3]$  (15 > 16)      no hay intercambio

8	15	16	44	27	12	35	67
---	----	----	----	----	----	----	----

$A[3] > A[4]$  (16 > 44)      no hay intercambio

8	15	16	44	27	12	35	67
---	----	----	----	----	----	----	----

$A[4] > A[5]$  (44 > 27)      sí hay intercambio

8	15	16	44	27	12	35	67
8	15	16	27	44	12	35	67

$A[5] > A[6]$  (44 > 12)      sí hay intercambio

8	15	16	27	44	12	35	67
8	15	16	27	12	44	35	67

$A[6] > A[7]$  (44 > 35)      sí hay intercambio

8	15	16	27	12	44	35	67
8	15	16	27	12	35	44	67

Al terminar las comparaciones en el segundo recorrido del vector, el elemento de mayor valor, en este caso 44, fue colocado en la penúltima posición del arreglo.

## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE BUBUJA

El resultado de los restantes recorridos son:

Tercer recorrido

08	15	16	12	27	35	44	67
----	----	----	----	----	----	----	----

Cuarto recorrido

08	15	12	16	27	35	44	67
----	----	----	----	----	----	----	----

Quinto recorrido

08	12	15	16	27	35	44	67
----	----	----	----	----	----	----	----

Sexto recorrido

08	12	15	16	27	35	44	67
----	----	----	----	----	----	----	----

Séptimo recorrido

08	12	15	16	27	35	44	67
----	----	----	----	----	----	----	----

Se realizan  $N-1$  recorridos transportando en cada recorrido el elemento de mayor valor a la derecha del vector.

Al final de los  $N-1$  recorridos los elementos del arreglo estarán ordenados.

<b>8</b>	<b>12</b>	<b>15</b>	<b>16</b>	<b>27</b>	<b>35</b>	<b>44</b>	<b>67</b>
----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------

## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE BUBUJA

**N = 5**

**REPETIR**

**B = 0**

**I = 0**

**MIENTRAS I < N-1**

**SI A(I) > A(I+1)**

**ENTONCES**

**W = A(I)**

**A(I) = A(I+1)**

**A(I+1) = W**

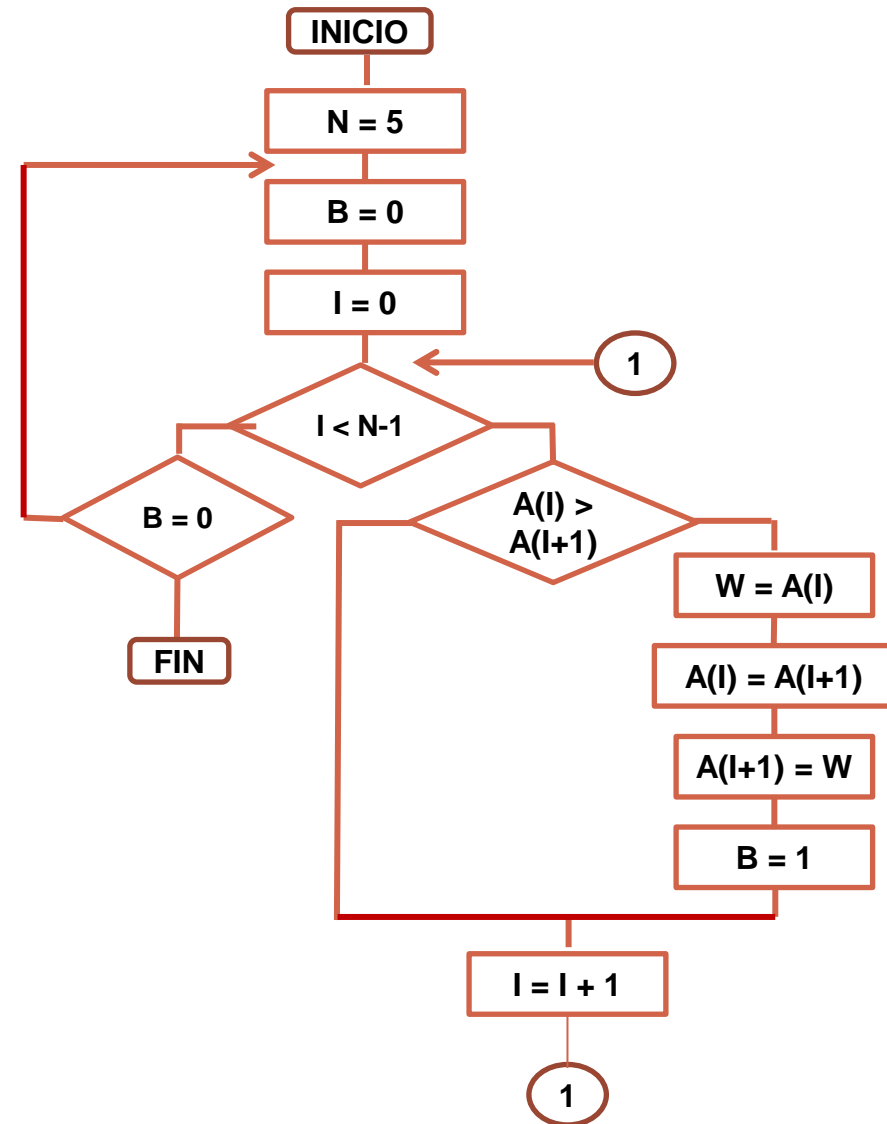
**B = 1**

**FIN SI**

**I = I + 1**

**FIN MIENTRAS**

**HASTA B = 0**



## ❖ ALGORITMO DE ORDENAMIENTO POR MÉTODO DE BUBUJA

Ordenar el vector A en forma Descendente

$N = 5$

REPETIR

$B = 0$

$I = 0$

MIENTRAS  $I < N-1$

SI  $A(I) < A(I+1)$

ENTONCES

$W = A(I)$

$A(I) = A(I+1)$

$A(I+1) = W$

$B = 1$

FIN SI

$I = I + 1$

FIN MIENTRAS

HASTA  $B = 0$

Para ordenar de MAYOR a MENOR el algoritmo cambia  $A(I) > A(I+1)$  por  $A(I) < A(I+1)$

