Tema 1

Ordenamientos, Búsquedas y recursividad

M. Sc. Dunia Soliz T.

Ordenamientos

Permite realizar la organización de un conjunto de datos ya sea en forma ascendente o descendente.

Ordenación interna → **VECTORES**

Ordenación externa → **ARCHIVOS**

El propósito principal de un ordenamiento es el de facilitar las búsquedas de los elementos.

Ordenamientos internos

Se tienen los siguientes tipos de ordenamientos INTERNOS.

Ordenamiento por intercambio
Ordenamiento por inserción
Ordenamiento por selección
Ordenamiento QUICK SORT

Ordenamiento por intercambio

Alternativa 1

```
Para i =1 hasta (n-1) incremento 1
      Para j = 0 hasta (n-2) incremento 1
            Si (A[ j ] > A[ j+1 ] ) entonces
                  temp    = A[ j ]
                  A[ j ]    = A[ j +1]
                  A[ j+1 ] = temp
            fin si
      fin para
fin para
```

Alternativa 2

```
Para i =0 hasta (n-2) incremento 1
      Para j = i+1 hasta (n-1) incremento 1
            Si (A[ i ] > A[ j ] ) entonces
                  temp    = A[ i ]
                  A[ i ]    = A[ j ]
                  A[ j ] = temp
            fin si
      fin para
fin para
```

Ejemplo

Alternativa 1

20	15	10	7	
15	10	7	20	1ra vez
10	7	15	20	2da vez
7	10	15	20	3ra vez

Se comparan pares consecutivos “n-1” veces

Ordenado

Ejemplo

Alternativa 2

-5	7	20	14,8	9	6
-5	6	20	14,8	9	7
-5	6	7	20	14,8	9
-5	6	7	9	20	14,8
-5	6	7	9	14,8	20

Se compara cada elemento de la izquierda con los n-1 elementos de su derecha, avanzado en cada repetición 1 elemento de la izquierda.

EMI
ESCUELA MILITAR DE INGENIERÍA
"Mcal. Antonio José de Sucre"
Prestigio, Disciplina y Mejores Oportunidades

ACTIVIDAD 1

Realizar un programa que realice las 2 alternativas de ordenamiento por intercambio, para ello usar un menú para elegir el tipo de ordenamiento

M. Sc. Dunia Soliz T.

Ordenamiento por inserción

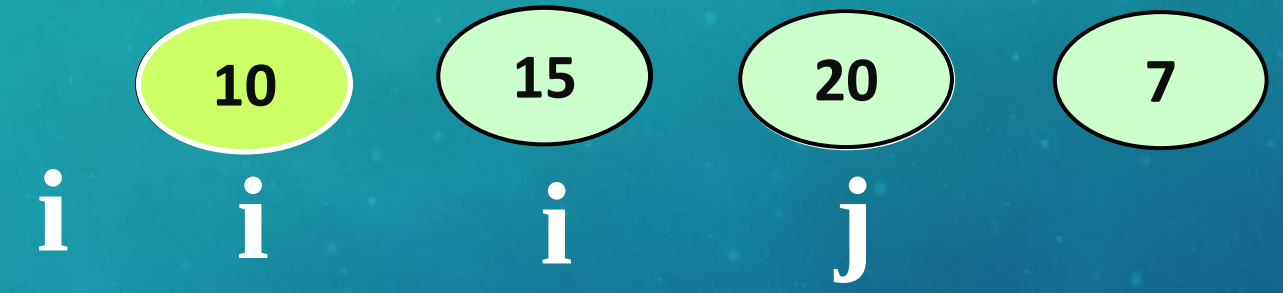
Consiste en insertar un elemento en el vector en una parte que este parcialmente ordenada. Se realiza el desplazamiento de los elementos que sean mayores a una llave y se inserta el elemento delante de los mayores que se encontraron.

```
Para j=1 hasta n-1 incremento 1
    llave= A[ j ]
    i =j -1
    Mientras (i >=0) y (A[i] > llave)
            A[i+1]=A[i]
             i =i -1
    fin mientras
    A[i+1] = llave
fin para
```

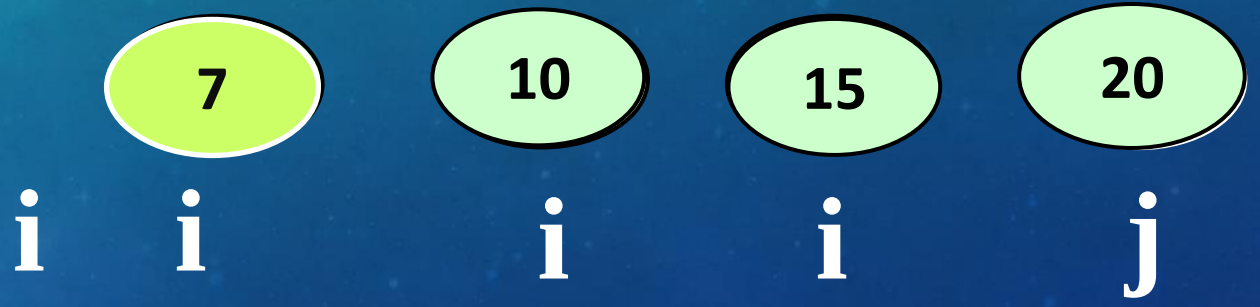
Ejemplo

1ra vez

2da vez



3ra vez



Ordenamiento por selección

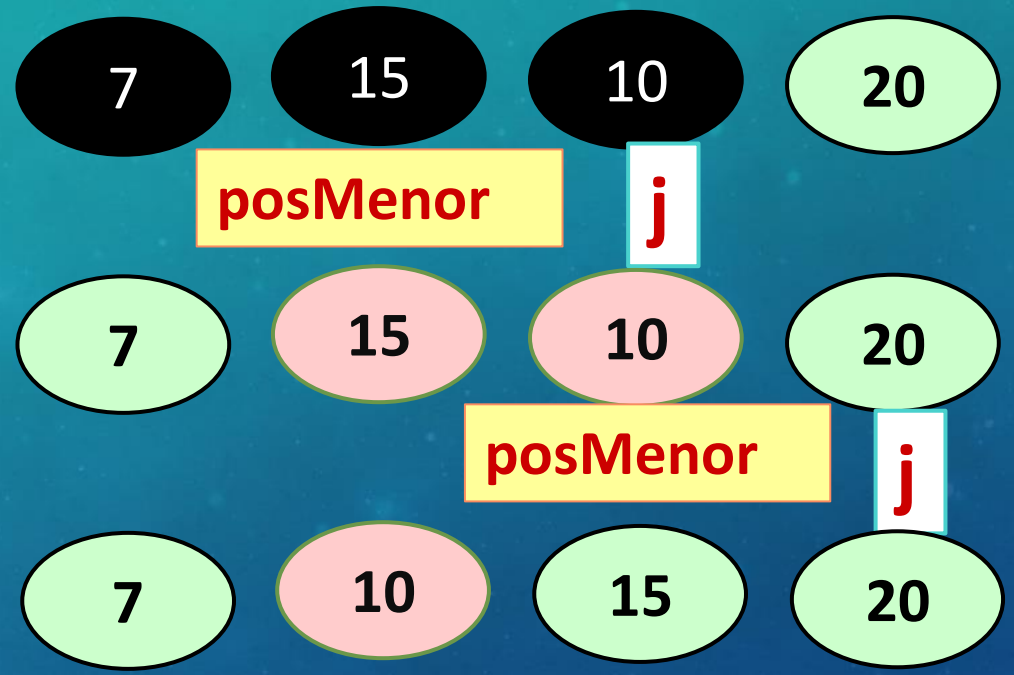
1. Seleccionar el menor elemento
2. Intercambiar dicho elemento con el primero
3. Repetir los pasos 1 y 2 con los n-1 elementos restantes

```
Para i =0 hasta (n-2) incremento 1
    pos_menor = i
    Para j = i+1 hasta n-1
       Si (A[j] < A[pos_menor] )entonces
            pos_menor = j
       fin si
    fin para
    AUX =A[pos_menor]
    A[pos_menor]=A[i]
    A[i] =AUX
fin para
```

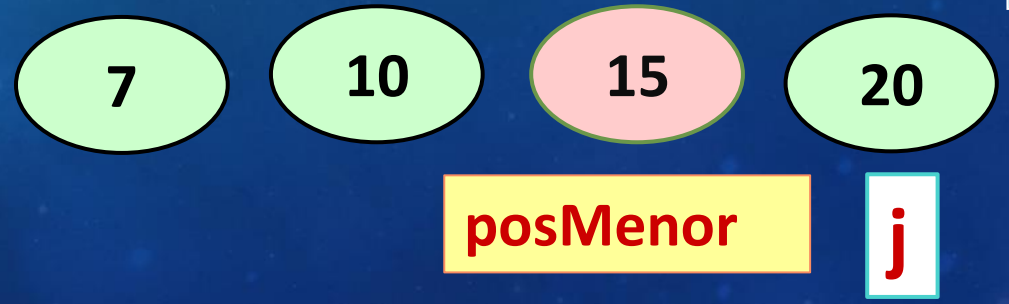
Ejemplo

1ra vez

2da vez



3ra vez



EMI
ESCUELA MILITAR DE INGENIERÍA
"Mcal. Antonio José de Sucre"
Prestigio, Disciplina y Mejores Oportunidades

ACTIVIDAD 2

Realizar un programa que ejecute los algoritmos de intercambio, inserción y selección, para ello el usuario debe llenar el arreglo y elegir el algoritmo que se desee utilizar

Ordenamiento QUICK SORT

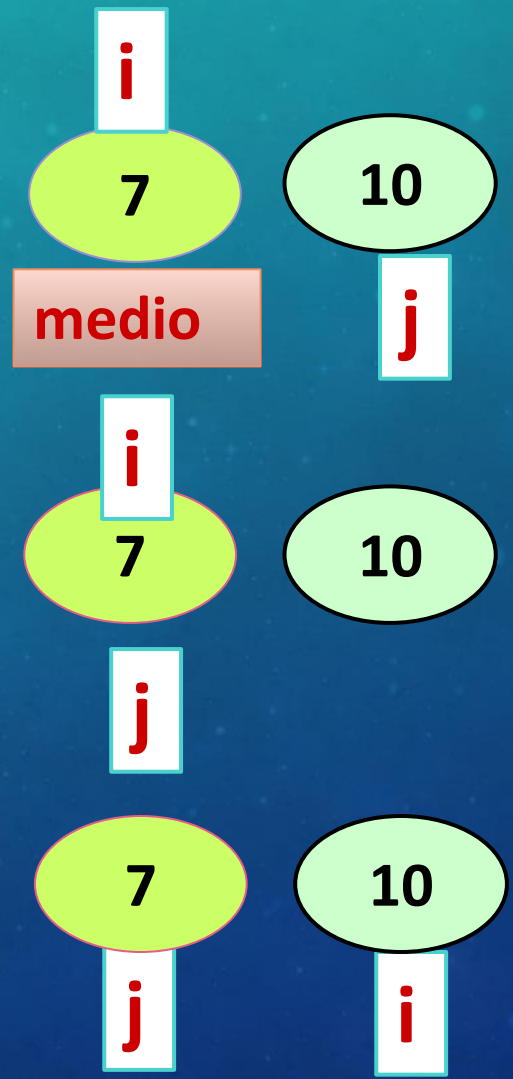
1. Se basa en la idea de divide y vencerás.
2. Se determina el pivote (que es el valor del centro) y se construyen dos sub-listas. En la 1ra lista se almacenan los valores menores al PIVOTE y en la segunda se almacenan los elementos mayores.
3. Este método se aplica hasta llegar a tener un vector de uno o dos elementos por ordenar.

Ordenamiento QUICK SORT

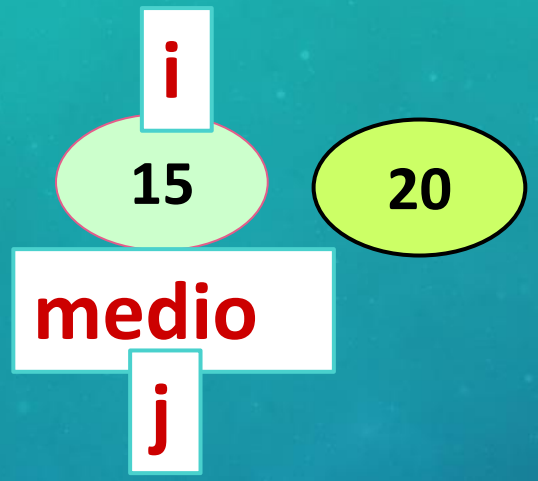
```
Procedimiento QuickSort(V[20] de enteros, Izq es entero, Der es entero)
    i = Izq
    j = Der
    medio = V[ ( Izq + Der) div 2 ]
    Repetir
        Mientras ( V[ i  ] < medio) hacer    i = i + 1  fin mientras
        Mientras ( V[ j  ] > medio) hacer    j = j – 1  fin mientras
        Si ( i <= j) entonces
            AUX =V[ i ]
            V[ i ]=V[ j ]
            V[ j ] =AUX
            i = i +1
            j = j – 1
        fin si
     Hasta ( i > j)
    Si (Izq < j) entonces QuickSort( V, Izq, j)
    Si ( i < Der) entonces QuickSort( V, i , Der)
Fin procedimiento
```

Ejemplo

1ra llamada



Ejemplo



2da llamada

EMI
ESCUELA MILITAR DE INGENIERÍA
"Mcal. Antonio José de Sucre"
Prestigio, Disciplina y Mejores Oportunidades

ACTIVIDAD 3

Realizar un programa que realice el ordenamiento de un vector usando el ordenamiento quickSort

EMI
ESCUELA MILITAR DE INGENIERÍA
"Mcal. Antonio José de Sucre"
Prestigio, Disciplina y Mejores Oportunidades

M. Sc. Dunia Soliz T.