

Algorítmica y Programación I

BÚSQUEDA

Búsqueda

- › Al proceso de ubicar un elemento en una estructura de datos (una colección), lo denominaremos búsqueda.
- › Este proceso se puede realizar de acuerdo a como se encuentren organizados sus elementos.
- › Se implementa, por lo general, con una función que retorna la posición del elemento o un valor negativo si no se encuentra.

V =	10	20	5	8	-7
-----	----	----	---	---	----

Búsqueda

```
Pos := buscar(V,N,elemento);
```

```
If( pos >= 1) then
```

```
    writeln('el elemento se encuentra en la  
    posición',pos)
```

```
else
```

```
    writeln('el elemento no se encuentra');
```

V =	10	20	5	8	-7
-----	----	----	---	---	----

Búsqueda secuencial

No conocemos la organización de sus elementos, o sea, no puedo saber si el elemento al que hago referencia es mayor o menor a los elementos que tengo hacia adelante.

El proceso consiste en buscar **desde el principio hasta el final** el elemento en cuestión.

Búsqueda secuencial

```
pos := 1;

while (pos <= N) and (V[pos] <> elemento) do
    pos := pos + 1;

if pos < N then
    busqueda := pos
else
    busqueda := -1
```

Búsqueda binaria

Conocemos el criterio con el que se encuentran organizados sus elementos (**ordenado**).

El proceso se basa en la estrategia “divide y vencerás”, consiste en **ubicar el elemento del medio** del arreglo y verificar si es el que se busca.

Si el elemento bajo esa posición coincide con el que se busca se retorna la misma.

Si el elemento no coincide se debe determinar con cual mitad debo continuar.

El proceso se continua hasta encontrar el elemento o no poder reducir más el espacio de búsqueda.

Búsqueda binaria

Arreglo ordenado – elemento buscado = 5

V =	-7	5	8	10	20	22	25	30	37	73
-----	----	---	---	----	----	----	----	----	----	----

↑ Primero = 1



Último = 10 ↑

$$\text{medio} = (\text{primero} + \text{último}) \text{ div } 2$$

```
medio := (primero + ultimo) div 2
```

```
if ( V[medio] = buscado ) then  
  pos = medio  
else if( V[medio] > buscado  
  ultimo := medio - 1  
else  
  primero := medio + 1
```

el proceso continua mientras
los índices no se crucen (el
primero no supere al último)