

# Ordenación vectores

CADP 2023

# CADP – Ordenación de vectores



El proceso por el cual, un grupo de elementos puede ser ordenado se conoce como algoritmo de ordenación.

Ordenar

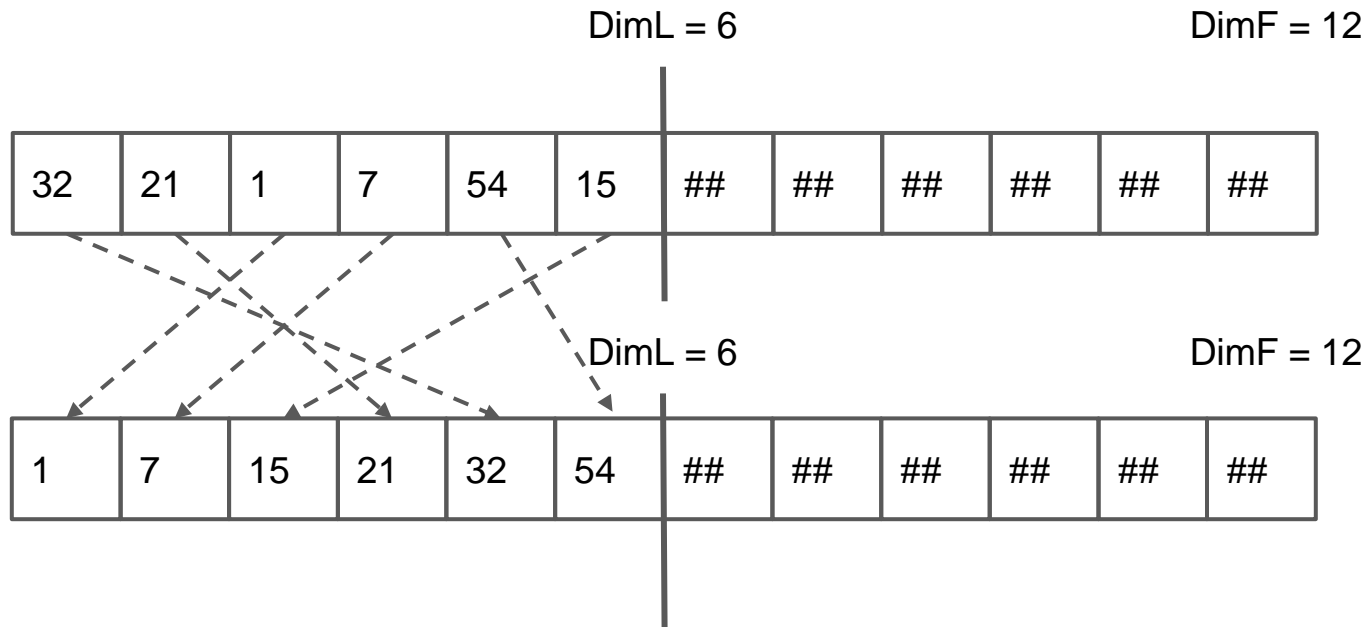
Algoritmos  
de  
ordenación

[ Selección  
Intercambio  
Inserción

Difieren en:  
Dificultad  
Memoria  
Tiempo

Selección

# CADP – Ordenación de vectores - Ejemplo



# CADP – Ordenación de vectores - Algoritmo de selección

## Cómo funciona?

Es un algoritmo de  $\text{dimL}$  pasadas.

Para cada pasada  $i$

Se elige el mínimo en el vector a partir de la posición  $(i+1)$  hasta el final

Si el mínimo de vector es más chico que lo que está almacenado en la posición  $i$  del vector se intercambia.

# CADP – Ordenación de vectores - Algoritmo de selección

dim lógica=6      dim física = 10

9	100	85	2	6	150				
---	-----	----	---	---	-----	--	--	--	--

Vector original

2	100	85	9	6	150				
---	-----	----	---	---	-----	--	--	--	--

Primera pasada

2	6	85	9	100	150				
---	---	----	---	-----	-----	--	--	--	--

Segunda pasada

2	6	9	85	100	150				
---	---	---	----	-----	-----	--	--	--	--

Tercera pasada

2	6	9	85	100	150				
---	---	---	----	-----	-----	--	--	--	--

Cuarta pasada

2	6	9	85	100	150				
---	---	---	----	-----	-----	--	--	--	--

Quinta pasada

# CADP – Ordenación de vectores - Algoritmo de selección

```
Program ordenamos;  
  const  
    tam = 150;  
  type  
    numeros= array [1..tam] of integer;  
  var  
    VN: numeros;  dimL:integer;  
  begin  
    llenarNumeros (VN,dimL);  //ya está hecho  
    ordenar(VN,dimL);  
  end.
```

# CADP – Ordenación de vectores - Selección

```
Procedure Ordenar ( var v: numeros; dimLog: integer);  
    var i, j, p, item: integer;  
    begin  
    for i:=1 to dimLog-1 do  
    begin {busca el mínimo v[p] entre v[i] , ..., v[N] }  
        p := i;  
        for j := i+1 to dimLog do  
            if v[ j ] < v[ p ] then  
                p:=j;  
            {intercambia v[i] y v[p] }  
            item := v[ p ];  
            v[ p ] := v[ i ];  
            v[ i ] := item;  
        end;  
    end;  
end;
```