



Conceptos de Algoritmos Datos y Programas

CADP – TEMAS



- Estructura de datos ARREGLO
- Dimensión física y dimensión lógica

Carga de valores

Lectura / Escritura

Recorridos

Dimensión física y lógica

Agregar elementos al final

Insertar elementos

Borrar elementos

Búsqueda de un elemento

Ordenación de los elementos



Supongamos que se existe un vector cargado de 10 elementos como máximo, pero por alguna circunstancia se cargaron sólo los primeros 4 valores.

20	77	68	2	?	?	?	?	?	?
----	----	----	---	---	---	---	---	---	---

a

Supongamos que sin saber que esto ocurrió se imprime el contenido del vector:

```
for i:= 1 to 10 do  
  write (a[i]);
```

Que se
obtendrá con la
impresión?



DIMENSION FISICA

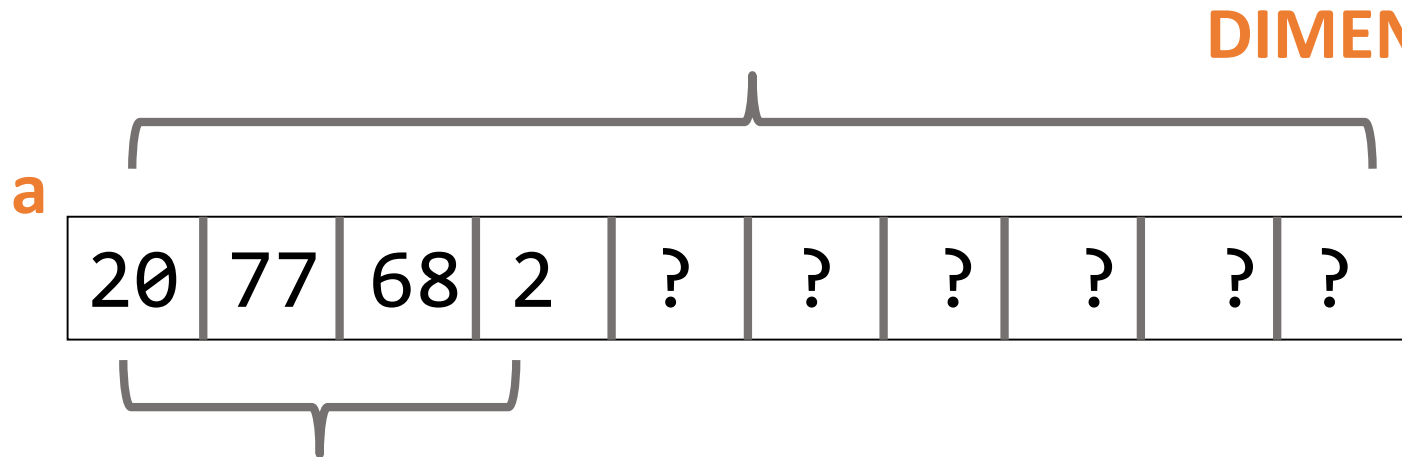
Se especifica en el momento de la declaración y determina su ocupación máxima de memoria.

La cantidad de memoria total reservada no variará durante la ejecución del programa.

DIMENSION LOGICA

Se determina cuando se cargan contenidos a los elementos del arreglo.

Indica la cantidad de posiciones de memoria ocupadas con contenido real. Nunca puede superar la dimensión física.



DIMENSION FISICA

Es la cantidad máxima de elementos que se pueden guardar en el arreglo.
No puede modificarse durante la ejecución del programa

DIMENSION LOGICA

Es la cantidad de elementos reales que se guardan en el arreglo.

Puede modificarse durante la ejecución del programa

Nunca puede ser mayor a la dimensión física (se debe controlar)

Cuándo se determina cada una?
Donde se declaran?



Realizar un programa que cargue un arreglo con números enteros hasta leer el número 50, a lo sumo se cargan 10 números.
Luego de terminar la carga informe cuál es el número mas grande de los leídos.

10
70
-1
50

a

10	70	-1	?	?	?	?	?	?	?
----	----	----	---	---	---	---	---	---	---

DF = 10
DL= 3

50
10
70
-1

a

?	?	?	?	?	?	?	?	?	?
---	---	---	---	---	---	---	---	---	---

DF = 10
DL= 0

11
4
2
80
-3
6
...

a

10	70	-1	4	80	-3	11	2	-1	6
----	----	----	---	----	----	----	---	----	---

DF = 10
DL= 10



Realizar un programa que cargue un arreglo con números enteros hasta leer el número 50, a lo sumo se cargan 10 números.

Luego de terminar la carga informe cuál es el número mas grande de los leídos.

Program uno;

Const

DF = 10

Type

valores = array [1..**DF**] of integer;

Var

v: valores;

max:integer;

dL:integer;

Begin

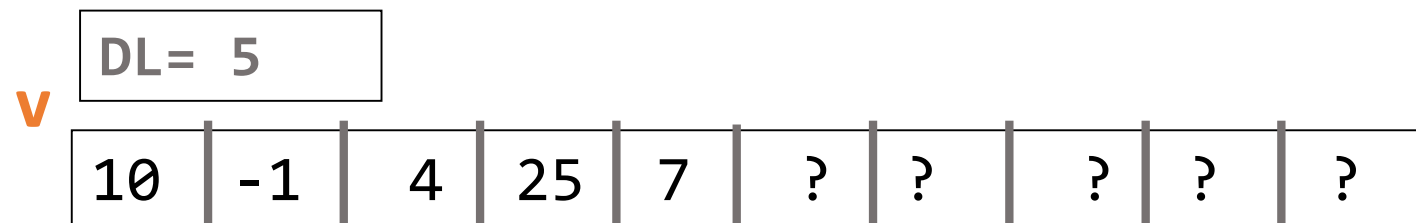
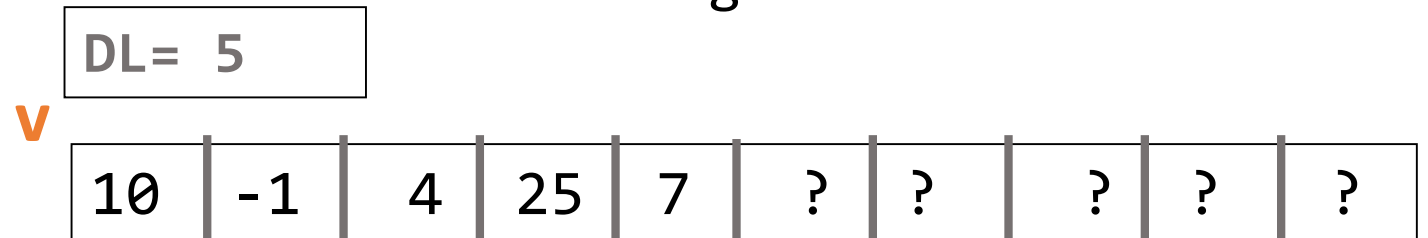
cargarValores (**v** , **dL**);

max:= maximo (**v** , **dL**);

End.

La dimensión física es una constante.

La dimensión lógica es una variable y toma valor cuando se carga el vector.




```
Procedure cargarValores (var a: números; var dimL:integer);
```

```
Var
```

```
  num:integer;
```

```
Begin
```

```
  dimL:=0;
```

```
  read (num);
```

```
  while (num <> 50) do
```

```
    begin
```

```
      a[dimL]:= num;
```

```
      read(num);
```

```
    end;
```

```
End;
```

Es correcto?



Cómo dimL, está inicializado en 0, la primera vez se accede a la posición a[0] y no es válida

```
Procedure cargarValores (var a: números; var dimL:integer);
```

```
Var
```

```
  num:integer;
```

```
Begin
```

```
  dimL:=1;
```

```
  read (num);
```

```
  while (num <> 50) do
```

```
    begin
```

```
      a[dimL]:= num;
```

```
      read(num);
```

```
    end;
```

```
End;
```

Es correcto?



Cómo dimL, nunca se incrementa, entonces carga siempre en la misma posición
a[1]

```
Procedure cargarValores (var a: números; var dimL:integer);
```

```
Var
```

```
  num:integer;
```

```
Begin
```

```
  dimL:=1;
```

```
  read (num);
```

```
  while (num <> 50) do
```

```
    begin
```

```
      a[dimL+1]:= num;
```

```
      read(num);
```

```
    end;
```

```
End;
```

Es correcto?



Cómo dimL, nunca se incrementa, entonces carga siempre en la misma posición
a[1]

```
Procedure cargarValores (var a: números; var dimL:integer);
```

```
Var
```

```
  num:integer;
```

```
Begin
```

```
  dimL:=1;
```

```
  read (num);
```

```
  while (num <> 50) do
```

```
    begin
```

```
      a[dimL]:= num;
```

```
      dimL:= dimL+1;
```

```
      read(num);
```

```
    end;
```

```
End;
```

Es correcto?



Si el primer número leído es 50, no entra al while, y como dimL está inicializado en 1, entonces devuelve que se cargó un elemento

```
Procedure cargarValores (var a: números; var dimL:integer);
```

```
Var
```

```
  num:integer;
```

```
Begin
```

```
  dimL:=0;
```

```
  read (num);
```

```
  while (num <> 50) do
```

```
    begin
```

```
      dimL:= dimL+1;
```

```
      a[dimL]:= num;
```

```
      read(num);
```

```
    end;
```

```
End;
```

Es correcto?



Qué pasa si leo mas de 10
números (el valor 50 no
apareció y ya leí 10 valores)

```
Procedure cargarValores (var a: números; var dimL:integer);
```

```
Var
```

```
  num:integer;
```

```
Begin
```

```
  dimL:=0;
```

```
  read (num);
```

```
  while ((dimL < dF) and (num <> 50)) do
```

```
    begin
```

```
      dimL:= dimL+1;
```

```
      a[dimL]:= num;
```

```
      read(num);
```

```
    end;
```

```
End;
```

Es correcto?



SI!!!!

```
function maximo (a: números; dimL:integer):integer;
```

```
Var
```

```
    max,i:integer;
```

```
Begin
```

```
    max:=-9999;
```

```
    for i:= 1 to dF do
```

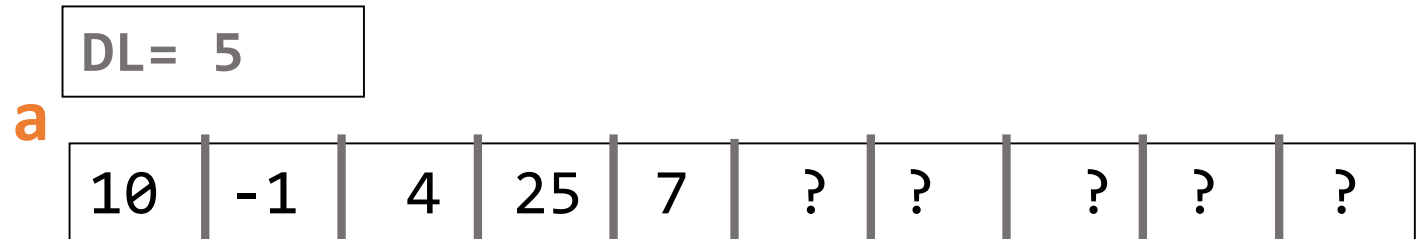
```
        begin
```

```
            if (a[i]>= max) then max:= a[i];
```

```
        end;
```

```
    maximo:= max;
```

```
End;
```



Es correcto?



NO! Sólo hay que recorrer
hasta la cantidad de elementos
cargados realmente

```
function maximo (a: números; dimL:integer):integer;
```

```
Var
```

```
    max,i:integer;
```

```
Begin
```

```
    max:=-9999;
```

```
    for i:= 1 to dimL do
```

```
        begin
```

```
            if (a[i]>= max) then max:= a[i];
```

```
        end;
```

```
    maximo:= max;
```

```
End;
```