

Algorítmica y Programación I

ARREGLOS

Características

- › Los arreglos son estructuras de datos **estáticas** que nos proveen la mayoría de los lenguajes de programación. Son espacios contiguos de memoria.
- › Estructura **compuesta** e **indexada** (puedo acceder a un elemento por medio de su posición).
- › Es un conjunto **homogéneo , finito y ordenado** de elementos.

Características

- › El nombre del arreglo (**identificador**) está asociado a un área de **memoria fija y consecutiva**.
- › El **índice**, debe pertenecer a un tipo de dato ordinal, permite acceder a cada elemento del arreglo. Puede verse como un desplazamiento respecto de la posición inicial del arreglo.
- › Homogéneo:
se refiere a que todos los elementos son del mismo tipo de dato.
- › Finito:
se refiere a que tienen una cantidad máxima de elementos (fija).
- › Ordenado:
se refiere a que sus elementos pueden ser identificados en función de su posición por ejemplo el primero, el segundo o el n'ésimo.

Unidimensionales (vector o arreglo lineal)

- › Los arreglos unidimensionales utilizan un **único índice** para indicar la posición en la que se encuentra el elemento.

- › Ej de un arreglo de elementos enteros:

› V=

4	2	8	1	8
---	---	---	---	---

- › Cantidad de elementos = 5

› $V[1] = 4$

N=5

› $V[3] = 8$

Tipo de elemento: Entero

Unidimensionales - Declaración Pascal

> Const

N = 5

> Type

TArreglo = array [1..N] of Integer

> ...

> Var

arreglo: TArreglo;

El tipo del **índice** debe ser **ordinal**
Entero,
Caracter: 'a'..'z'
o Enumerado

Inicialmente, ¿qué valor tiene cada elemento?



Unidimensionales – Uso Pascal

› Const

N = 5

› Type

TArreglo = array [**1..N**] of Integer

› ...

› Var

arreglo: TArreglo;

V[0] := 2 □ Error

V[1] := 10

V[2] := 5

V[3] := -3

V[4] := 6

V[5] := 0

V[6] := 5 □ Error

10	5	-3	6	0
----	---	----	---	---

Unidimensionales – Uso Pascal

```
procedure inicializar(var V:TAreglo;N:integer)
var i:integer;
Begin
    for i := 1 to N do
        v[i] := i*10;
    end;
```

10	20	30	40	50
----	----	----	----	----

EJERCICIO

› Desarrolle en pascal un programa que permita almacenar las temperaturas promedio diarias registradas durante el mes de marzo. Produzca un informe por pantalla que indique cual fue el día que tuvo la mayor temperatura.

. NOTA: considere que solo hay un día con la máxima.

› Ej:
el día 3 registró la máx...

Día	temperatura
1	3
2	2
3	10
...	...
31	7

SOLUCIÓN

```
Program EjerTemp;  
Const  
    N=31;  
Type  
    TTemperaturas = arreglo[1..N] of integer;  
Var  
    temperaturas:TTemperaturas;  
    max:integer;  
Begin  
    cargar(temperaturas,N);  
    maxTemp := obtenerMaxima(temperaturas,N);  
    informar(temperaturas,N,maxTemp);  
End.
```

SOLUCIÓN

```
Procedure cargar(var temp:Ttemperaturas, N:integer);  
Var  
    i:integer;  
Begin  
    for i:= 1 to N do  
        begin  
            writeln('ingrese la temp. promedio del día ',i);  
            readln(temp[i]);  
        end;  
    End;
```

SOLUCIÓN

```
Function obtenerMaxima(temp:TTemperaturas; N:integer):integer
Var
    i:integer;
Begin
    max := temp[1];
    for i:= 2 to N do
        begin
            if(temp[i] > max) then
                max := temp[i];
            end;
        obtenerMaxima := max;
    End;
```

SOLUCIÓN

```
Procedure informar(temp:TTemperaturas; N,max:integer);  
Var  
    i:integer;  
Begin  
    for i:= 1 to N do  
        begin  
            if(temp[i] = max) then  
                writeln('el día ', I,' tiene la máxima temp.);  
End;
```

EJERCICIO

- › Desarrolle en pascal un programa que permita almacenar las temperaturas mínimas y máximas diarias registradas durante el mes de marzo. Produzca un informe por pantalla que indique cual fue el día que tuvo la temperatura mayor y cual tuvo la temperatura menor.
- › NOTA: Puede darse el caso que haya más de un día con la máxima o con la mínima temperatura.

› Ej:

Día	Mínima	Máxima
1	3	10
2	2	8
...		
31	1	10

SOLUCIÓN

```
Program EjerTemp;
Const
    N=31;
Type
    TTempDiaria = record
        max,min:integer;
    end;

    TTemperaturas = arreglo[1..N] of TTempDiaria;
Var
    temperaturas:TTemperaturas;
    max,min:integer;
Begin
    cargar(temperaturas,N);
    obtenerMaxMin(temperaturas,N,max,min);
    informar(temperaturas,N,max,min);
End.
```

SOLUCIÓN

```
Procedure cargar(var temp:TTemperaturas,cont  
N:integer);
```

```
Var
```

```
    i:integer;
```

```
Begin
```

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

```
        begin
```

```
            writeln('ingrese la max. y min. de día ',i);
```

```
            readln(temp[i].max, temp[i].min);
```

```
        end;
```

```
End;
```

SOLUCIÓN

```
Procedure obtenerMaxMin(const temp; const N:integer;  
                        var max,min:integer);
```

```
Var
```

```
  I:integer;
```

```
Begin
```

```
  max := v[1].max;
```

```
  min := v[1].min;
```

```
  for i:= 2 to N do
```

```
    begin
```

```
      if(temp[i].max > max) then
```

```
        max := temp[i].max;
```

```
      if(temp[i].min< min) then
```

```
        min := temp[i].min;
```

```
      end;
```

```
End;
```


SOLUCIÓN

```
Procedure informar (const temp; const N:integer;
                   const max,min:integer);

Var
    i:integer;
Begin
    for i:= 1 to N do
        begin
            if(temp[i].max = max) then
                writeln('el día ', I,' tiene la máxima temp. ');

            if(temp[i].min = min) then
                writeln('el día ', I,' tiene la mínima temp. ');
        end;
    End;
```