

TIPURI DE DATE STRUCTURATE

PROIECT REALIZAT DE
MARIUS MALAI ELEV AL
CLASEI X-A B

DEF:

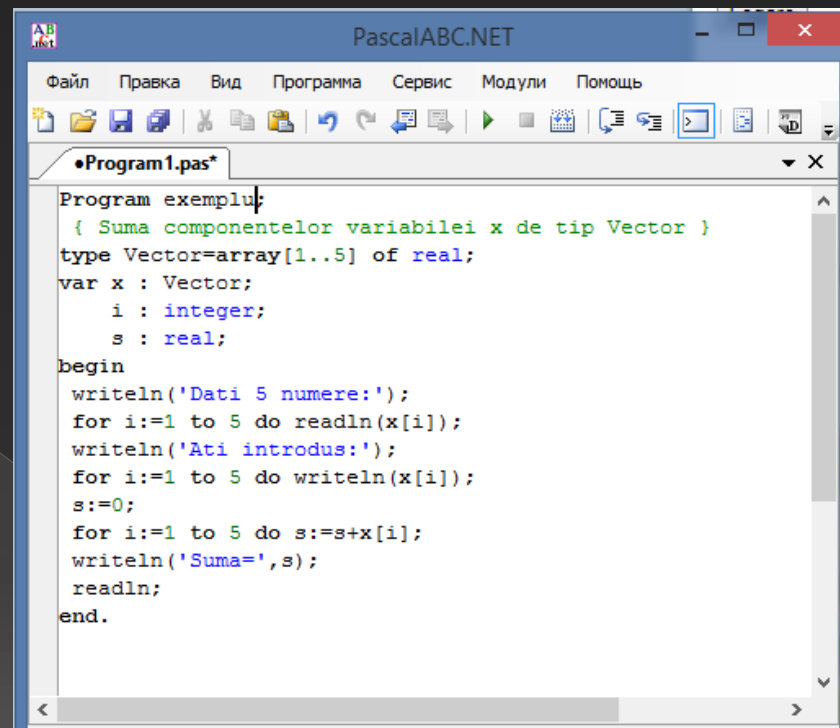
Un sir de elemente de același tip, în care contează ordinea elementelor, se numește vector sau tablou unidimensional. Un tablou(array) este o structura formata dintr-un număr fixat de componente de același tip, numit tip de baza. Numărul de componente este determinat de numărul de valori ale indicilor, care sunt obligatoriu tipuri ordinale. Poziția unui element se mai numește și indicele sau rândul elementului, iar elementele se mai numesc și componente ale vectorului.

SINTAXA:

1. Type vector=array[1..20] of integer
 reprezintă un sir de elemente numere întregi numerotate de la 1 la 20.
2. Prin VAR x:vector;
 x[3] -componenta de ordin 3
 x[i] -componenta de ordin i

Multimea de valori ale unui tip de date array este constituita din tablouri. Tablourile sînt formate dintr-un număr fixat de componente de același tip, denumit tip de baza. Tipul de date tablou se definește printr-o construcție de forma:
type <Nume tip> = array [T1] of [T2]; unde T1 este tipul indicelui care trebuie să fie ordinal, iar T2 este tipul componentelor (tipul de baza) care poate fi un tip oarecare.

TIPUL TABLOU (ARRAY)



```
Program exemplu;  
{ Suma componentelor variabilei x de tip Vector }  
type Vector=array[1..5] of real;  
var x : Vector;  
    i : integer;  
    s : real;  
begin  
    writeln('Dati 5 numere:');  
    for i:=1 to 5 do readln(x[i]);  
    writeln('Ati introdus:');  
    for i:=1 to 5 do writeln(x[i]);  
    s:=0;  
    for i:=1 to 5 do s:=s+x[i];  
    writeln('Suma=',s);  
    readln;  
end.
```

Observații:

Nu este obligatoriu ca elementele unui tablou să fie numere reale. Ele pot fi de orice natură: numere întregi, caractere...

În matematică tablourile se numesc matrici. Termenul a fost preluat și în informatică.

Tipul de date "Șir de caractere"

O constantă de tip șir de caractere se declară între două caractere ". În memoria internă, o constantă de acest tip este reținută sub forma unui vector de caractere.

Operațiile permise:

>, <, =, <=, >= operatori logici

Declararea unui șir de caractere:

type <Nume tip> ::= packed array [1..n] of char;

Tipul șir caractere (String). Tipul String a fost conceput pentru a prelucra șiruri de caractere. Pentru că date de tip Array Of Char nu permite operații specifice șirurilor de caractere, tipul string reușește astfel de operații. Acest tip se declară astfel :

String [m] sau simplu String. m reprezintă numărul maxim de caractere pe care poate să-l conțină o variabilă de acest tip. Dacă nu se specifică această limită m (varianta a doua), atunci lungimea maximă este 255 (valoarea implicită pentru m).

TABLOU BIDIMENSIONAL (MATRICI)

Def:

MATRICE = tablou bidimensional în care regasirea unui element se face pe baza de 2 indici.

i - indice de linie

j - indice de coloana.

a[i,j] - matricea a cu i linii și j coloane.

Nr de elementelor unei matrice =
 $i*j$;

OPERATII CU MATRICI

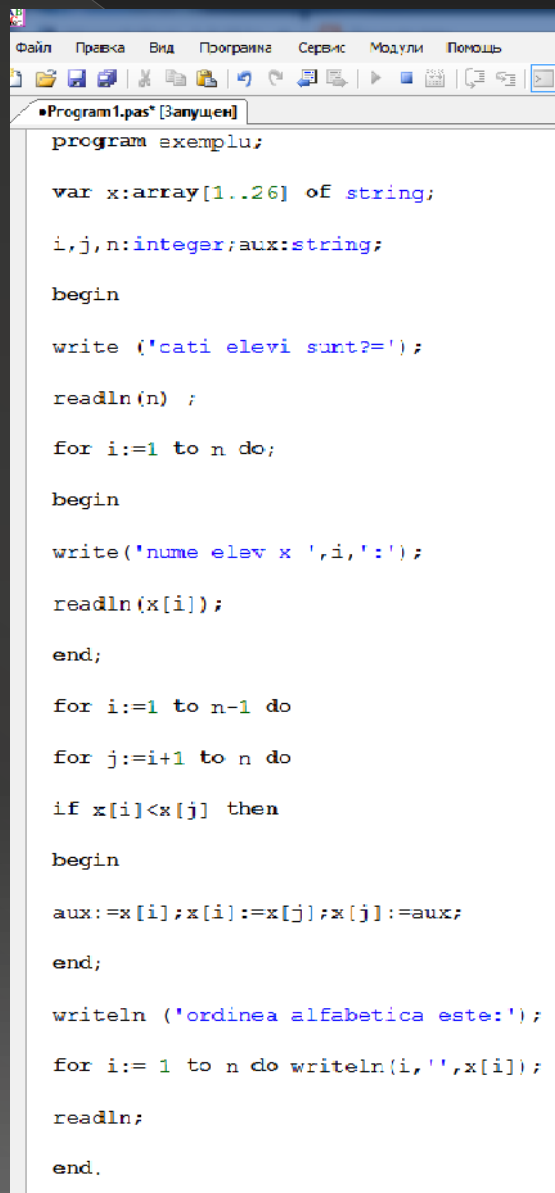
a) citirea

- se face prin intermediul a doua structuri de tip for

```
for i := 1 to m do  
  for j := 1 to n do  
    begin  
      write('a[' , i , ' , ' , j , ' ] = ');  
      readln(a[i,j]);  
    end;
```

b) tiparirea

```
for i := 1 to m do begin  
  for j:=1 to n do write(a[i,j], ' ');  
  writeln;  
end;
```



```
program exemplu;

var x:array[1..26] of string;
i,j,n:integer;aux:string;

begin

write ('cati elevi sunt?');

readln(n) ;

for i:=1 to n do;

begin

write('nume elev x ',i,':');

readln(x[i]);

end;

for i:=1 to n-1 do

for j:=i+1 to n do

if x[i]<x[j] then

begin

aux:=x[i];x[i]:=x[j];x[j]:=aux;

end;

writeln ('ordinea alfabetica este:');

for i:= 1 to n do writeln(i,',',x[i]);

readln;

end.
```

TIPUL STRING

DEF:

- tip special de vector cu elemente de tip caracter (char), care memoreaza in afara caracterelor din sir si lungimea respectivului sir de caractere, adica nr de caractere existente in sir.

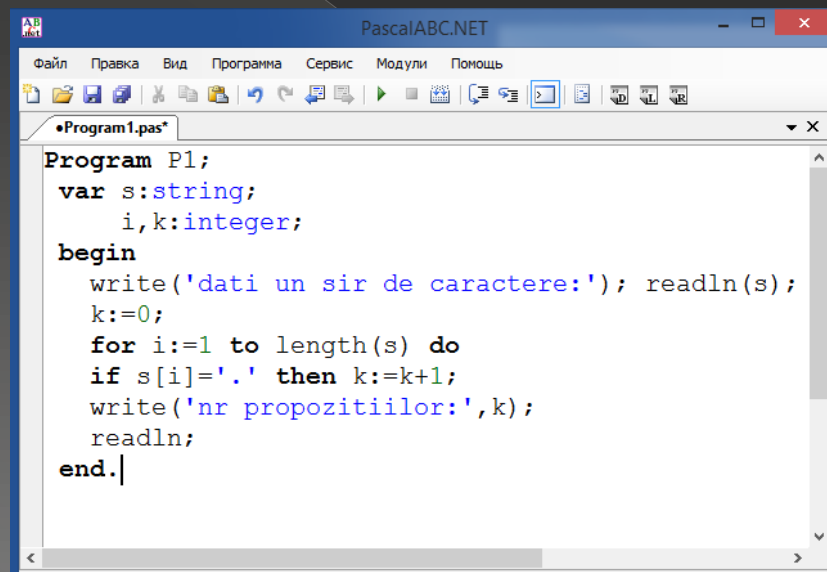
-- variabilele de tip string ocupa in memorie un spatiu egal cu n+1 octeti, unde n reprezinta nr de caractere din sir.

-- o varibila de tip string este o succesiune de caractere cuprinsa intre 2 caractere apostrof

-- tipul string este predefinit, adica nu e nevoie sa fie declarat cu "type" el fiind cunoscut.

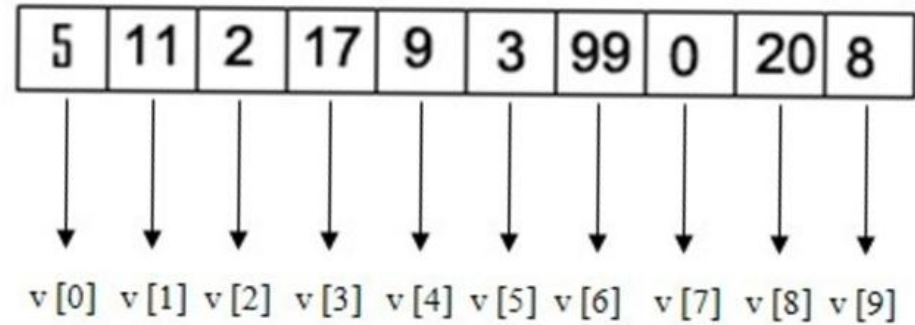


```
program min_matrice;
var x:array[1..20,1..21] of integer;
i,j,n,min:integer;
begin
readln(n);
for i:=1 to n do
for j:=1 to n do begin
write('x[' ,i, ',' ,j, ']=');readln(x[i,j]);end;
min:=x[1,1];
for i:=2 to n do
for j:=2 to n do begin
if (<(j:=n+1-i) and (<min(x[i,j])) then
max:=x[i,j]; writeln('min este:',min);end;
for i:=1 to n do begin
for j:=1 to n do
writeln('x[' ,i, ',' ,j, ']=',x[i,j]);
end;
readln;
end.
```



```
Program P1;
var s:string;
i,k:integer;
begin
write('dati un sir de caractere:'); readln(s);
k:=0;
for i:=1 to length(s) do
if s[i]='.' then k:=k+1;
write('nr propozitiilor:',k);
readln;
end.
```


VECTOR



1

Un tablou pentru care la declarare este specificată o singură dimensiune, iar poziția unui element este specificată utilizând un singur indice, se numește tablou unidimensional sau vector. (reprezentat în imagine)

Operațiile permise într-un vector:

$+$, $-$, $*$, $/$

Declararea unui vector:

```
type vector = array [1..n] of integer;
```

```
var V: vector;
```

```
  i: integer;
```

2

Operațiile permise într-un vector:

$+$, $-$, $*$, $/$

Declararea unui vector:

```
type vector = array [1..n] of integer;
```

```
var V: vector;
```

```
  i: integer;
```

Citirea unui vector de la

tastatură:

```
for i:= 1 to n do write (V[i]);
```

Minim:

```
for i:=1 to n do  
if V[i]<V[i+1] then  
  Min:=V[i] else  
  Min:=V[i+1];
```

Maxim:

```
for i:=1 to n do  
if V[i]>V[i+1] then  
  Max:=V[i] else  
  Max:=V[i+1];
```