

Variabile indexate în limbajul C

BREVIAR TEORETIC

VARIABLE INDEXATE

Putem împărți tipurile de date fundamentale ale limbajului C în tipuri de date numerice și tipul caracter. În mod normal nu există diferențe funcționale între tipul caracter și tipul întreg, dar întâlnim o abordare diferită la modul de inițializare a unui caracter sau șir de caractere. Acest lucru se va observa în secțiunea dedicată tipului caracter, în cadrul laboratorului de față.

VARIABLE INDEXATE NUMERICE

Variabilele indexate se mai numesc și tablouri de date (unidimensionale¹, bidimensionale², etc.). Acestea permit stocarea mai multor valori sub același nume.

DECLARAREA ȘI INIȚIALIZAREA

Sintaxa generală:

```
<tip> <idVar>[ <dim1> ][ <dim2> ]... = { <secv_init.> };
```

Exemplu – declarare șir și matrice fără inițializare:

```
#define N 10 // definim constnata N având val. 10
int sir[ 10 ]; // un șir de 10 int
float mat[ N/2 ][ N+1 ]; // o matrice cu 5 linii și 11 col
```

Exemplu – declarare șir cu inițializare:

```
int sir[ ] = { 1, 2, 3 }; //dimensiunea sirului = 3
int sir[ 10 ] = { 1, 2, 3 }; //dimensiunea sirului = 10
```

Exemplu - declarare matrice cu inițializare::

```
int A[ 4 ][ 3 ] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12} };
int A[4][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12} };
int A[][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12} };
int M[2][3] = { {1}, {2} };
int M[2][3] = { {1, 0, 0}, {2, 0, 0} };
```

¹ Numite vectori, șiruri.

² Numite matrici.

UTILIZARE

Accesul la o variabilă indexată (șir/matrice/...) se face prin intermediul numelui urmat de un set de indecși (în funcție de tipul variabilei, dacă este uni/bi/multi dimensională).

Sintaxa generală:

```
<i dVar>[ <i dx1> ][ <i dx2> ]...;
```

Exemplu:

```
sir[ 9 ] = sir[ 0 ] + 1;  
mat[ 0 ][ 0 ] = mat[ I ][ j+1 ];
```

VARIABLE INDEXATE CARACTER

În cazul tipului caracter inițializarea datelor se poate face ca și în cazul tipurilor numerice, dar mai avem o posibilitate specifică acestui tip, astfel:

- un caracter poate fi inițializat specificând simbolul caracterului 'A', '#', etc.;
- un șir de caractere poate fi inițializat specificând înșiruirea simbolurilor între ghilimele: "Acesta este un șir".

DECLARAREA ȘI INIȚIALIZAREA

Sintaxa generală:

```
<tip> <i dVar>[ <di m1> ][ <di m2> ]... = [ { <secv_init.> } ];
```

Exemplu – declarare șir cu inițializare – IDENTICE:

```
int sir[ ] = { 'T', 'E', 'S', 'T', '\0' };  
int sir[ ] = { "TEST" };
```

OPERAȚII PE BAZĂ DE VARIABLE INDEXATE

În cadrul acestui paragraf se va urmări modul în care citim/afișăm de la consolă elementele unui șir/matrice de întregi/reali/caractere.

CITIREA ȘI AFIȘAREA UNUI ȘIR DE ÎNTREGI

Pentru citirea/afișarea unui șir de întregi se va folosi, cel mai adesea, o structură iterativă (for), prin care putem indica elementul curent, în care se va face citirea sau care va fi folosit la afișare, prin intermediul unui index. Acest lucru se poate observa în exemplul ce va urma.

CITIREA ȘI AFIȘAREA UNUI ȘIR DE CARACTERE

Pentru citirea/afișarea unui șir de caractere se vor folosi familii de funcții, specializate în aceste operații, după cum urmează:

- pentru operarea cu consola (scanf(), citind doar un cuvânt³);
scanf("%s", sir);
printf("%s", sir);
- pentru operarea cu consola (gets(), citind o linie⁴);
gets(sir);
puts(sir);

CITIREA ȘI AFIȘAREA UNEI MATRICI DE REALI

Pentru citirea/afișarea unei matrici de reali se va folosi, cel mai adesea, doua structuri iterative (for), prin care putem indica elementul curent, în care se va face citirea sau care va fi folosit la afișare, prin intermediul unui index. Prima structură for va itera liniile din matrice, pe când a doua structură (îmbricată) va itera elementele de pe linia curenta (coloanele). Acest lucru se poate observa în exemplul următor.

PROBLEME REZOLVATE

Exemplificare pentru cazul operării cu șiruri/matrici:

```

/*
 * Elemente de bază – variabile indexate
 * AUTOR:
 * Remus PRODAN
 * DATA:
 * 12. 11. 2009
 */
#include <stdio.h>
#include <conio.h>
#include <math.h>

int main( void )
{
    int i, j, sir[100], N = 3, M = 2;
    char fraza[80];
    float mat[10][10];

    printf( "\n\t[ START]"
           "\n< Operatii de baza cu variabile indexate >\n");
    printf( "\n<----- SIRURI INT ----->");

    //Citirea de la consolă a N elemente întregi
    printf( "\nIntroduceți elementele șirului: \n");
    for( i = 0; i < N; i++ )
    {
        printf( "S(%d) = ", i );
        scanf ( "%d", &sir[ i ] );
    }

    //Afișarea pe consola a N elemente întregi

```

³ Cuvintele sunt delimitate de caractere precum spațiu sau linie nouă.

⁴ În acest caz delimitatorul este caracterul pentru linie nouă

```
printf( "\nSirul: { " );
for( i = 0; i < N; i++ )
    printf( "%3d, ", sir[ i ] );
printf( "\b\b }. " );
printf( "\n<----- /SIRURI INT ----->\n" );
printf( "\n<----- SIRURI CHAR ----->\n" );

//Citirea de la consola a unei fraze (separate cu spatii)
printf( "\nIntroduceti elementele sirului: \n" );
fflush( stdin ); //echivalent cu flushall();
gets( fraza );

//Afisarea pe consola a N elemente intregi
printf( "\nFraza: { %s }", fraza );
printf( "\n<----- /SIRURI CHAR ----->\n" );
printf( "\n<----- MATRICI FLOAT ----->\n" );
printf( "\nIntroduceti elementele matricii:\n" );
for( i = 0; i < N; i++ )
    for( j = 0; j < M; j++ )
        { printf( "M(%d,%d) = ", i, j );
          scanf ( "%f", &mat[ i ][ j ] );
        }
printf( "\nMatricea: " );
for( i = 0; i < N; i++ )
    { printf( "\n" );
      for( j = 0; j < M; j++ )
          { printf ( "%5.2f", mat[ i ][ j ] );
            }
    }
printf( "\n<----- /MATRICI FLOAT ----->\n" );
printf( "\n\t[ END ]\n" );

return 0;
}
```

```
[ START ]
< Operatii de baza cu variabile indexate >
<----- SIRURI INT ----->
Introduceti elementele sirului:
S<0> = 1
S<1> = 2
S<2> = 3

Sirul: < 1, 2, 3 >.
<----- /SIRURI INT ----->
<----- SIRURI CHAR ----->
Introduceti elementele sirului:
Acesta este un test

Fraza: < Acesta este un test >
<----- /SIRURI CHAR ----->
<----- MATRICI FLOAT ----->
Introduceti elementele matricii:
M<0,0> = 1
M<0,1> = 2
M<0,2> = 3
M<1,0> = 3
M<1,1> = 2
M<1,2> = 1
M<2,0> = 2
M<2,1> = 2
M<2,2> = 2

Matricea:
0.33 0.67 1.00
1.00 0.67 0.33
0.67 0.67 0.67
<----- /MATRICI FLOAT ----->

[ END ]
```

PROBLEME PROPUSE SPRE REZOLVARE

1. Determinați cel mai lung subșir, strict crescător, dintr-un șir de elemente întregi, preluat de la consola.
2. Scrieți un program care realizează citirea, afișarea și numărarea cuvintelor dintr-o frază, citită de la consola.
3. Scrieți un program care va determina dacă un cuvânt este sau nu palindrom, cuvântul fiind citit de la consola. Palindromul este acel cuvânt care fiind citit în ambele sensuri (de la un capăt spre celalalt sau invers) are același sens.
4. Calculați transpusa unei matrici, preluată de la consola. Transpusa - matricea la care se inversează liniile cu coloanele.