

Lucrarea de laborator nr. 12

Tema: Structuri și tablouri de structuri

Obiective

- declararea structurilor;
- citirea și afișarea structurilor;
- accesarea câmpurilor unei structuri;
- declararea tablourilor cu elemente de tip structură;
- ordonarea tablourilor cu elemente de tip structură.

12.1. Teme pentru studiu prealabil

- Tipurile de date în limbajul C.
- Tablouri.
- Structuri.

12.2. Sarcina pentru soluționare

De elaborat un program în care se introduc 7 – 10 rânduri de tabelă. Tabelul este afișat la ecran – imediat după introducerea și sortarea lui (după valorile din prima coloană). Ultimul rând – remarca, poate fi neglijată.

Mănăstiri din Moldova			
Mănăstirea	Hramul	Din secolul	Suprafața (ha)
Căpriană	A	15	18,8
Hâncu	S	17	24,7
Japca	I	17	12,2
Remarcă: A – Adormirea Maicii Domnului, S – Sfânta Cuvioasă Paraschiva, I – Înălțarea Domnului.			

12.3. Exemplu de soluționare a sarcinii

Elaborarea algoritmului de soluționare.

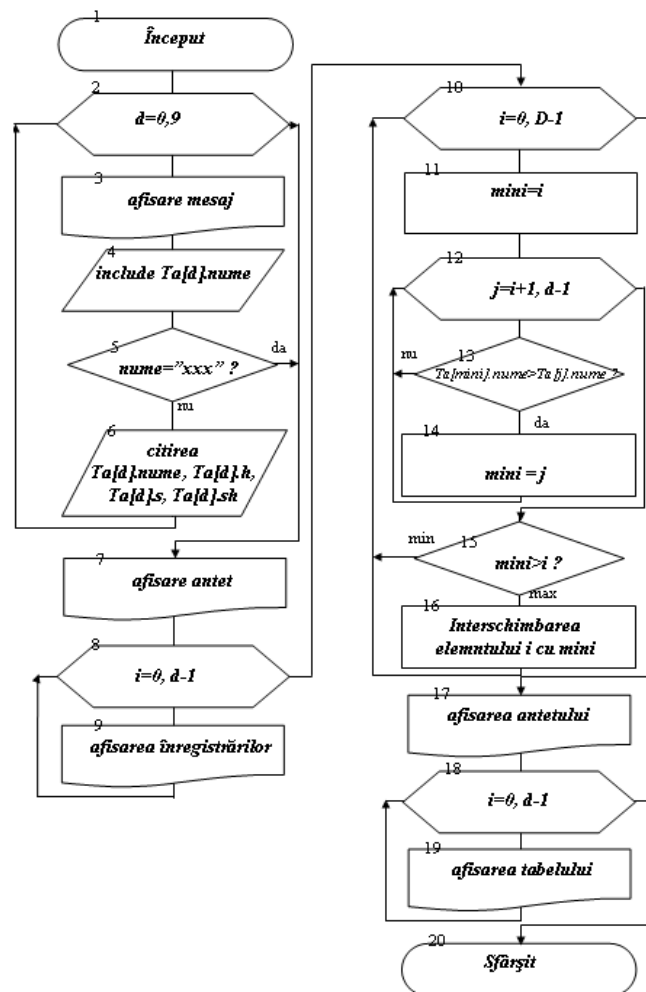
Algoritmul se începe cu introducerea datelor în tablou. Introducerea are loc în bucla cu contorul d , care se schimbă de la 0 până la 9 (blocul 2). La fiecare iterație bucla afișă invitația (blocul 3) de introducere a elementelor structurii (blocul 4,6). După introducerea primei componente – *nume*, se verifică valoarea ei (blocul 5). La introducerea sirul "xxx" – are loc ieșirea din buclă. După ieșirea din buclă variabila d reprezintă numărul de înregistrări introduse în tabel. Astfel, programul poate prelucra un tablou de până la 10 elemente.

Se continuă cu bucla ce tipărește antetul (blocul 7) și rândurile tabelului (blocul 8, 9).

Următoarea buclă compusă (blocul 10) realizează sortarea tabelului cu ajutorul buclei plasate după algoritmul simplelor schimbări de selecție. La prima iterație a buclei externe se execută căutarea elementului tabloului cu valoare minimă din câmpul *nume*. La începutul buclei elementul minim este considerat primul element (blocul 11). Apoi în buclă (blocul 12) se revăd celelalte elemente ale tabloului, și fiecare se compară cu cel minim (blocul 13). Dacă elementul curent *nume* este mai mic decât elementul minim, în continuare acesta va fi considerat minim (blocul 14). Indicele elementului minim se înscrie în variabila *mini*. După ieșirea din buclă

internă, elementul minim găsit (blocul 15), se schimbă cu primul (blocul 16). Astfel, elementul minim al tabloului ajunge la locul său. La următoarea iterație a buclei externe se îndeplinește căutarea minimului între elementele tabloului, începând cu al doilea, al treilea ș.a.m.d.. La ieșirea din bucla externă tabloul se găsește sortat.

Extragerea tabloului sortat (blocul 17-19).



Des. 12.1. Schema bloc

Determinarea variabilelor programului

Structura definește un șablon, cu ajutorul căruia programul mai târziu poate să definească una sau mai multe variabile. Cu alte cuvinte, programul mai întâi de toate declară structura, apoi anunță variabile de tipul structurii. Declararea acestei structuri are forma:

```

struct manastire {
char nume[15]; /* manastirea */
char h;        /* hramul */
int s;         /* din secolul */
float sh;      /* suprafata */
};
  
```

Declarăm un tabloul de elemente:

```
struct manastire Ta[10];
```

Pentru a efectua interschimbarea elementelor tabloului se declară o variabilă temporală:

```
struct manastire x;
```

Conform schemei algoritmului, este nevoie de următoarele variabile: numărul de elemente introduse - d , indicele extern (i) și intern (j) pentru bucle, indicele elementului minim - $mini$:

```
int i, j, d, mini;
```

Elaborarea textului programului

Textul programului se începe cu includerea fișierelor *stdio.h* și *string.h*. Fișierul *string.h* conține descrierea funcțiilor utilizate în lucrul cu șirurile de caractere.

Programul urmează cu descrierea structurii *manastire* și declararea tabloului *Ta*. Tabloul va fi amplasat în memoria statică.

Funcția principală urmează cu declararea celorlalte variabile.

Se deschide bucla simplă cu contorul d , la fiecare iterație a buclei se afișă invitația și se introduc valorile câmpurilor elementului curent al tabloului. Comparația câmpului *nume* cu constanta "xxx" (semnul de sfârșit a introducerii) se înfăptuiește cu funcția *strcmp()*. Dacă este introdus semnul de sfârșit, are loc ieșirea anticipată din buclă după operatorul *break*.

Urmează, totuși, să ne oprim la introducerea valorilor pentru câmpul *sh*. Aici ne ciocnim cu fenomenul, care nu poate fi explicat alt fel, ca o eroare în sistemul de programare: funcția *scanf()* nu lucrează sigur la introducerea valorilor de tip *float* și *double*, dacă acestea sunt valorile câmpurilor structurii. Din aceste considerente se declară variabila de lucru *shx* de tipul *float* și valoarea câmpului *sh* mai întâi se introduce în această variabilă, dar mai apoi se atribuie câmpului structurii.

Apoi urmează afișarea tabloului.

Următorii operatori ai programului amănunțit realizează blocurile 10 - 15 a schemei algoritmului. O revedere detaliată necesită numai blocul 16 – deplasarea elementelor. În primul rând, la deplasare se utilizează structura de lucru x : Întâi conținutul elementului i se transmite în x , apoi conținutul elementului $mini$ se transmite în elementul i , și conținutul x se transmite în elementul $mini$. În al doilea rând, operația de atribuire, care de obicei se utilizează pentru deplasarea valorilor nu poate fi aplicată către o variabilă, prin urmare atribuirea se înfăptuiește pentru fiecare câmp aparte. Pentru câmpul *nume* care este un șir de caractere se utilizează funcția *strcpy()*.

Afișarea tabelului rezultat este la fel, ca și afișarea tabelului inițial.

Textul integral al programului este adus mai jos.

```

#include <iostream.h>
#include <string.h>
#include <iomanip.h>
//Descrierea structurii
struct manastire{
    char nume[15];
    char h;
    int s;
    float sh;
}
Ta[10]; //Determinarea tabloului manastire
int main(void) {
    struct manastire x; //variabila de lucru
    int d; //numarul de elemente in tablou
    int i, j; //indicii curenti in tablou
    int mini; //indicii elementului minim
    float shx; //variabila de lucru
//citirea datelor
    for(d=0; d<10; d++){
        cout<<"\n Introduceti: denumirea, hramul, secolul, suprafata -"<<d+1;
        cin>>Ta[d].nume;
        if(!strcmp(Ta[d].nume,"xxx")) break;
        cin>>Ta[d].h>>Ta[d].s>>shx;
        Ta[d].sh=shx;
    }
//Afisarea tabelului
    cout<<"-----\n";
    cout<<"|                Manastiri din Moldova                |\n";
    cout<<"-----\n";
    cout<<"| Manastirea | Hramul | Din secolul | Suprafata |\n";
    cout<<"-----\n";
//afisarea sirului cu valori reale
    for(i=0; i<d;i++){
        cout<<setw(13)<<Ta[i].nume<<setw(9)<<Ta[i].h;
        cout<<setw(13)<<Ta[i].s<<setw(11)<<Ta[i].sh<<' \n';
        cout<<"-----\n";
//sortarea tabloului
        for(i=0; i<d-1; i++){
            mini=i; //indicele primului element minim
            for(j=i+1; j<d;j++){
                //daca elementul curent este mai mic decat cel minim, el devine minim
                if(strcmp(Ta[mini].nume, Ta[j].nume)>0) mini=j;
//interschimbarea elementului curent cu cel minim
                if(mini>i){
                    strcpy(x.nume,Ta[i].nume); x.h=Ta[i].h; x.s=Ta[i].s; x.sh=Ta[i].sh;
                    strcpy(Ta[i].nume,Ta[mini].nume); Ta[i].h=Ta[mini].h;
                    Ta[i].s=Ta[mini].s; Ta[i].sh=Ta[mini].sh;
                    strcpy(Ta[mini].nume,x.nume); Ta[mini].h=x.h; Ta[mini].s=x.s; Ta[mini].sh=x.sh;
                }
}
// Afisarea tabloului sortat
        cout<<"-----\n";
        cout<<"|                Manastiri din Moldova                |\n";
        cout<<"-----\n";
        cout<<"| Manastirea | Hramul | Din secolul | Suprafata |\n";
        cout<<"-----\n";
//afisarea sirului cu valori reale
        for(i=0; i<d;i++){
            cout<<setw(13)<<Ta[i].nume<<setw(9)<<Ta[i].h;
            cout<<setw(13)<<Ta[i].s<<setw(11)<<Ta[i].sh<<' \n';
            cout<<"-----\n";
        }
        return 0;
    }
}

```

Des. 12.2. Textul programului

Ajustarea programului

La ajustarea programului se utilizează regimul pas cu pas pentru a urmări valorile variabilelor ce se introduc. Dacă apar probleme cu introducerea variabilelor, este indicat de introdus fiecare variabilă cu un operator separat.

Mai complicată în ajustare poate fi partea programului, care se execută la sortarea tabloului. De aceea, cel mai bine la apariția erorilor în acest sector al programului de inclus afișări intermediare. De exemplu, a afișa toată tabela după fiecare buclă de sortare. În regimul pas cu pas poate fi urmărită vloarea minimumului găsit la fiecare iterație.

Rezultatul lucrului programului

În rezultatul lucrului programului la ecran au foat afișate următoarele:

```

C:\ Borland C++ for DOS
1. Introduceti: denumirea, hramul, secolul, suprafata - Capriana A 15 18.8
2. Introduceti: denumirea, hramul, secolul, suprafata - Japca I 17 12.2
3. Introduceti: denumirea, hramul, secolul, suprafata - Hancu S 17 24.7
4. Introduceti: denumirea, hramul, secolul, suprafata - xxx

|          Manastiri din Moldova          |
|-----|-----|-----|-----|
| Manastirea | Hramul | Din secolul | Suprafata |
|-----|-----|-----|-----|
| Capriana   | A      | 15          | 18.8      |
| Japca      | I      | 17          | 12.2      |
| Hancu      | S      | 17          | 24.7      |
|-----|-----|-----|-----|
|          Manastiri din Moldova          |
|-----|-----|-----|-----|
| Manastirea | Hramul | Din secolul | Suprafata |
|-----|-----|-----|-----|
| Capriana   | A      | 15          | 18.8      |
| Hancu      | S      | 17          | 24.7      |
| Japca      | I      | 17          | 17.0      |
|-----|-----|-----|-----|

```

12.4. Întrebări, probleme și discuții

1. În ce cazuri sunt utilizate structurile?
2. Cum este interpretată de program denumirea unei structuri?
3. Cum putem să ne adresăm către elementele unei structuri?
4. Se considera ca într-un depozit sunt produse care au un cod, exista într-o anumita cantitate și au un preț unitar. Să se conceapă o structura pe baza căreia sa se scrie un program care să afișeze valoarea totala a produselor existente în depozit și codul produsului cu cea mai mare contribuție la această valoare.
5. Se citesc de la tastatură pentru d ($1 < d < 20$) firme care vând tehnică de calcul, numele, adaosul comercial practicat (în procente) și firma furnizoare. Cunoscându-se prețul unui produs la o firmă, să se afișeze prețul aceluiași produs vândut de altă firmă. Numele celor doua firme se dau prin citire.
6. Se citesc pentru d ($d \leq 50$) țări numele, și culorile drapelelor lor. Se citește apoi o culoare. Se cere listarea țărilor care au aceeași culoare în componența drapelului.
7. Pentru d ($d \leq 10$) concurenți se citesc numele și notele obținute la probele de gimnastica. Să se afișeze concurenții în ordinea clasării lor.
8. Pentru d produse ($d < 10$) se citesc: prețul în lei la data intrării în magazin și data expirării termenului de valabilitate. Știind că prețul scade progresiv cu aceeași suma, astfel încât devine 0 în ziua următoare celei în care termenul de valabilitate a expirat, să se calculeze și afișeze prețul produsului la o anumita dată sau să se afișeze un mesaj din care rezultă ca produsul a expirat.