**Ministerul Educației și Cercetării**

|  |
| --- |
| **Universitatea Tehnica a Moldovei** |

**Facultatea Calculatoare, Informatica si Microelectronica**

**Structuri de date și Algoritmi**

**Raport**

|  |
| --- |
| Lucrarea de Laborator nr. 1  Tema: Algoritmi de prelucrare a structurilor, a tablourilor și pointerilor de la structuri |

|  |  |
| --- | --- |
| A efectuat stundent(ul/a) grupei: | SI-212 |
| Numele si prenumele elev(ului/ei): | Vozian Vladimir |
|  |  |
| Profesorul: | Mititelu Vitalie |

|  |
| --- |
| Chișinău 2021 |

**Scopul lucrării:**

Programarea algoritmilor de prelucrare a structurilor și a tablourilor de la structuri prin utilizarea funcțiilor, pointerilor, alocării dinamice a memoriei în limbajul C.

**Sarcina:**

Pentru tabloul unidimensional dat cu elemente de tip structură (conform variantelor) să se afișeze la ecran următorul meniu de opțiuni:

1. Alocarea dinamică a memoriei pentru tabloul de structuri.

2. Introducerea elementelor tabloului de la tastatură.

3. Afișarea elementelor tabloului la ecran.

4. Adăugarea unui element nou la sfârșit.

5. Modificarea elementului tabloului.

6. Căutarea elementului tabloului.

7. Sortarea tabloului.

8. Eliminarea elementului indicat din tablou.

9. Eliberarea memoriei alocate pentru tablou.

0. Ieșire din program.

Să se elaboreze funcțiile pentru implementarea opțiunilor meniului.

**Varianta:**

16. Structura Canal TV cu câmpurile: denumirea, țara, telefonul, genul, ratingul.

**Indicație:**

Dacă întâmpinați dificultăți la realizarea meniului sau anumitor opțiuni ale acestuia puteți realiza doar anumite opțiune ale acestuia (minim- citirea de la tastatură și afișarea pe cran), dar evident că nota va fi micșorată în acest caz.

**Rezumat**

***Tablouri de structuri***

Cel mai uzual mod de folosire a structurilor este in tablouri de structuri. Pentru declararea unui tablou de structuri, mai intai se defineste o structura si apoi se declara un tablou de variabile de acel tip. De exemplu, pentru declararea unui tablou cu 100 de structuri *addr* definite anterior, se va scrie:

***struct addr addr\_info[100];***

Pentru a avea acces la o structura oarecare din cele 100 se va indexa numele structurii (in cazul acesta *addr\_info*).

De exemplu:

**printf ('%d', addr\_info[2].zip);**

are ca efect afisarea codului zip din a treia structura.

Se observa ca, la fel orice variabila tablou, tablourile de structuri incep cu indexul 0.

Pentru a ilustra modul de utilizare a structurilor si tablourilor de structuri prezentam un exemplu de program pentru actualizarea unei liste de corespondenta.

Informatiile ce vor fi memorate se refera la *name, street, city, state, zip*. Pentru definirea structurii de baza *addr* care va contine aceste informatii vom scrie:

**struct addr addr\_info[SIZE];**

Tabloul *addr\_info* contine *SIZE* structuri de tip *addr*, unde *SIZE* se defineste dupa necesitati.

Prima functie necesara in program este *main(),* a carei structura este urmatoarea:

**void main() }}**

Functia *init\_list()* pregateste tabloul structura pentru utilizare prin punerea unui caracter *null* in primul *byte* al campului 'nume'. Programul impune ca o variabila structura sa nu fie utilizata daca campul nume este vid. Aceasta initializare are loc in memoria interna a calculatorului (nu in fisierul *maillist*de pe disc)*.*

Structura functiei de initializare *init\_list()* ar putea fi urmatoarea:

**/\* Functia init\_list() \*/**

**void init\_list()**

Functia de selectare a meniului *menu()* va afisa mesajele optiunilor si va returna varianta aleasa. Prin tastarea literei din paranteze, se va lansa in executie o anumita procedura.

**/\* Functia menu() \*/**

**char menu() while (!strrchr('edlsq',ch));**

**return tolower(ch); }**

***Observatie***: Functia *strrchr(cs,c)* din *<string.h>* intoarce un pointer la ultima aparitie a lui *c* in *cs* sau NULL daca nu apare.

Functia *enter()* are ca efect introducerea unor noi informatii in urmatoarea structura libera a listei *addr\_info[SIZE]*. Aceasta introducere se efectueaza prin determinarea primei structuri libere din memorie (cu *addr\_info.name* setata la 0) si prin completarea sa cu informatii culese de la tastatura.

**/\* Functia enter() \*/**

**void enter()**

**printf ('Name: ');**

**gets (addr\_info[i].name);**

**printf ('Street: ');**

**gets (addr\_info[i].street);**

**printf ('City: ');**

**gets (addr\_info[i].city);**

**printf ('State: ');**

**gets (addr\_info[i].state);**

**printf ('Zip: ');**

**scanf ('%d',&addr\_info[i].zip);}**

**Codul deplin al programului:**

*#include <stdio.h>*

*#include <stdlib.h>*

*#include <string.h>*

*typedef struct CanalTV{*

*char \*denumire;*

*char \*tara;*

*unsigned long telefon;*

*char \*gen;*

*float rating;*

*}CTV;*

*CTV \*canale;*

*int n;*

*void menu(int \*i){*

*printf("\n1.Alocarea dinamica a memoriei pentru tabloul de structuri");*

*printf("\n2.Introducerea elementelor tabloului de latastatura.");*

*printf("\n3.Afisarea elementelor tabloului la ecran.");*

*printf("\n4.Adaugarea unui nou element la sfarsit.");*

*printf("\n5.Modificarea elementului tabloului.");*

*printf("\n6.Cautarea elementului tabloului.");*

*printf("\n7.Sortarea tabloului.");*

*printf("\n8.Eliminarea elementului indicat din tablou.");*

*printf("\n9.Eliberarea memoriei alocate pentru tablou.");*

*printf("\n0.Iesirea din program");*

*printf("\nAlegeti optiunea - ");*

*fflush(stdin);*

*scanf("%d",i);*

*}*

*CTV \*create(){*

*printf("Dati numarul de canale - ");*

*scanf("%d",&n);*

*while(n<0){*

*printf("Nu putem avea un numar negativ de canale\nDati alt numar de canale:");*

*scanf("%d",&n);*

*}*

*CTV \*canale;*

*canale = (CTV\*)malloc(n\*sizeof(CTV));*

*return canale;*

*}*

*void citireStruct(int i){*

*char str[255];*

*printf("Canalul %d\n",i+1);*

*printf("Denumirea: ");*

*fflush(stdin);*

*gets(str);*

*canale[i].denumire = (char\*)malloc((strlen(str) + 1 ) \* sizeof(char));*

*strcpy(canale[i].denumire,str);*

*printf("Tara: ");*

*fflush(stdin);*

*gets(str);*

*canale[i].tara = (char\*)malloc((strlen(str) + 1) \* sizeof(char));*

*strcpy(canale[i].tara,str);*

*printf("Genul: ");*

*fflush(stdin);*

*gets(str);*

*canale[i].gen = (char\*)malloc((strlen(str) + 1) \* sizeof(char));*

*strcpy(canale[i].gen,str);*

*printf("Telefonul: ");*

*scanf("%lu",&canale[i].telefon);*

*printf("Rating: ");*

*scanf("%f",&canale[i].rating);*

*if (!(0<=canale[i].rating && canale[i].rating<=10))*

*while(!(0<=canale[i].rating && canale[i].rating<=10)){*

*printf("Rating gresit!\nRating:");*

*scanf("%f" ,&canale[i].rating);*

*}*

*}*

*void citire(){*

*printf("\nCitire canale\n");*

*for(int i = 0; i < n; i++)*

*citireStruct(i);*

*}*

*void afisare(){*

*if(n){*

*printf("| Nr. | \tDenumirea | \tTara | \tTelefonul | \tGenul | \tRating |\n");*

*for(int i = 0; i < n; i++)*

*printf("|%3d |%15s |%15s |%5lu |%15s |%12.2f |\n",i+1,canale[i].denumire,canale[i].tara,canale[i].telefon,canale[i].gen,canale[i].rating);*

*}*

*}*

*void inserare(){*

*CTV \*tempCanal = realloc(canale, ++n \* sizeof(CTV));*

*if(tempCanal){*

*canale = tempCanal;*

*int i = n-1;*

*printf("Noul canal:\n");*

*citireStruct(i);*

*}else*

*printf("Nu e posibil de inserat un element nou");*

*}*

*void editare(){*

*int i;*

*printf("Dari numarul elementului care doriti sal editati : ");*

*scanf("%d",&i);*

*i--;*

*char str[255];*

*printf("\nDati noile valori al canalului dat:\n");*

*citireStruct(i);*

*}*

*void search(){*

*int b = 1;*

*char str[255];*

*printf("Dati denumirea canalului care doriti sal cautati: ");*

*fflush(stdin);*

*gets(str);*

*for(int i = 0; i < n; i++)*

*if(strcmp(str,canale[i].denumire) == 0){*

*b = 0;*

*printf("| Nr. | \tDenumirea | \tTara | \tTelefonul | \tGenul | \tRating |\n");*

*printf("|%3d |%15s |%15s |%5lu |%15s |%12.2f |\n",i+1,canale[i].denumire,canale[i].tara,canale[i].telefon,canale[i].gen,canale[i].rating);*

*}*

*if(b == 1)*

*printf("\nNu a fost gasit canal cu astfel de denumire!");*

*}*

*void del(){*

*int x;*

*printf("Dati numarul canalului pe care doriti sal eliminati: ");*

*scanf("%d",&x);*

*x--;*

*for(int i = x; i < n-1; i++)*

*canale[i] = canale[i+1];*

*n--;*

*canale = (CTV\*)realloc(canale,n\*sizeof(CTV));*

*printf("Eliminarea efectuata cu succes");*

*}*

*void eliberare(){*

*for(int i = 0; i < n; i++){*

*free(canale[i].denumire);*

*free(canale[i].tara);*

*free(canale[i].gen);*

*}*

*free(canale);*

*n = 0;*

*}*

*void sort(){*

*int option;*

*printf("Alegeti cum doriti sa fiba sortat tabelul\n");*

*printf("1. Raiting crescator\n2. Raiting descrescator\n");*

*scanf("%d",&option);*

*int sortat;*

*CTV temp;*

*switch(option){*

*case 1 :*

*do*

*{*

*sortat = 1;*

*for (int i = 0; i < n - 1; i++)*

*if (canale[i].rating > canale[i + 1].rating)*

*{*

*sortat = 0;*

*temp = canale[i];*

*canale[i] = canale[i + 1];*

*canale[i + 1] = temp;*

*}*

*} while (sortat == 0);*

*break;*

*case 2 :*

*do*

*{*

*sortat = 1;*

*for (int i = 0; i < n - 1; i++)*

*if (canale[i].rating < canale[i + 1].rating)*

*{*

*sortat = 0;*

*temp = canale[i];*

*canale[i] = canale[i + 1];*

*canale[i + 1] = temp;*

*}*

*} while (sortat == 0);*

*break;*

*default :*

*printf("Ati ales optiune incorecta");*

*break;*

*}*

*}*

*int main(){*

*int i;*

*do{*

*menu(&i);*

*switch(i){*

*case 1 : canale = create(); break;*

*case 2 : citire(); break;*

*case 3 : afisare(); break;*

*case 4 : inserare(); break;*

*case 5 : editare(); break;*

*case 6 : search(); break;*

*case 7 : sort(); break;*

*case 8 : del(); break;*

*case 9 : eliberare(); break;*

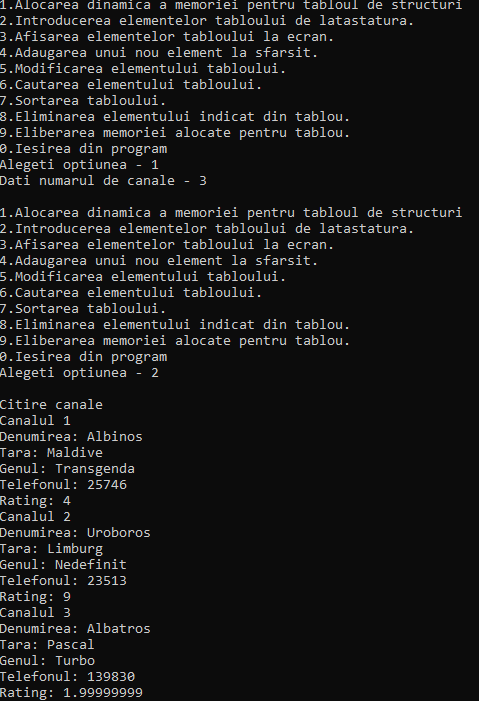
*}*

*}while(i);*

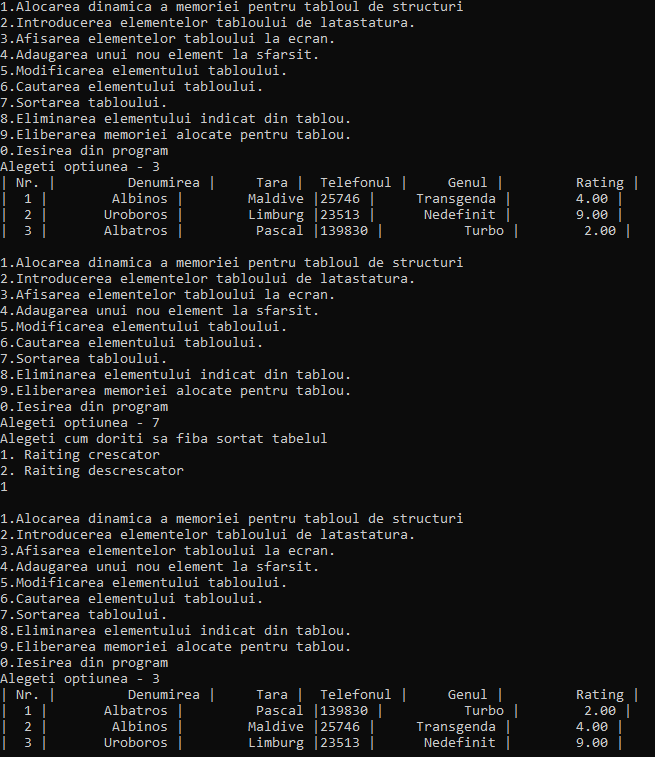
*return 0;*

*}*

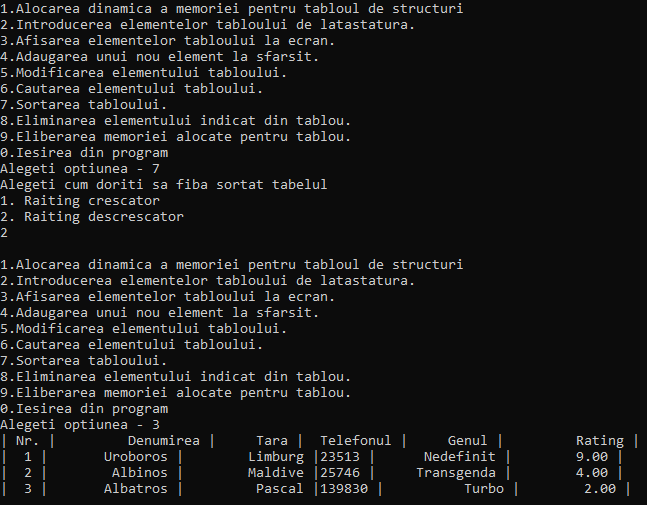
**Exemple de execuție a programului:**



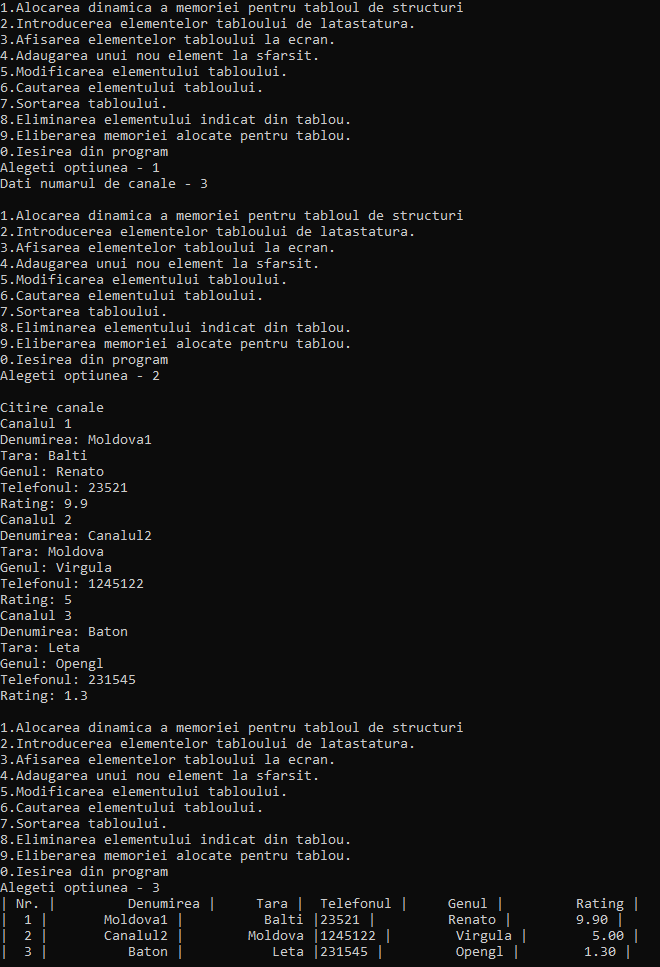
Aici este arătată citirea de la tastatura



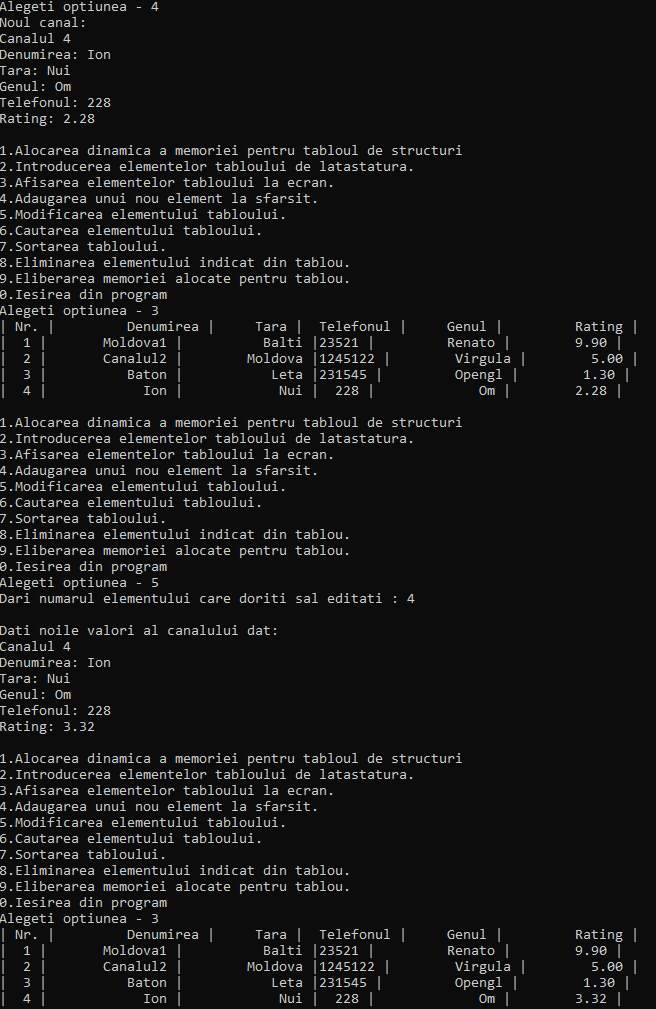
Afișarea elementelor și sortarea lor crescătoare



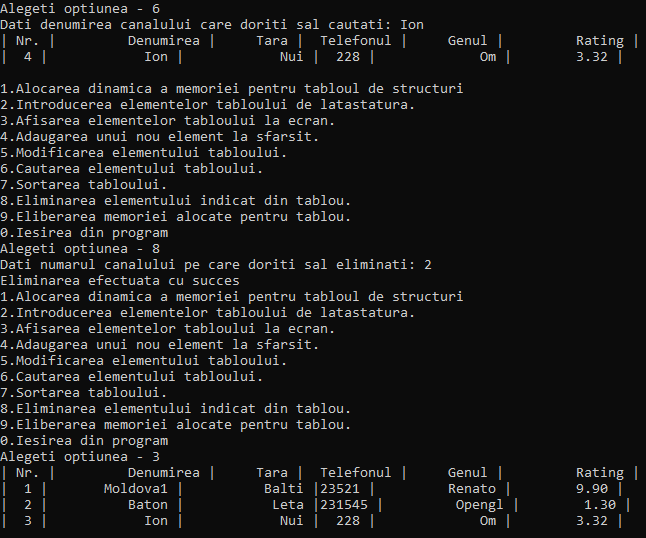
Afișarea și sortarea lor descrescătoare



Deja alt set de canale, citirea și afișarea lor

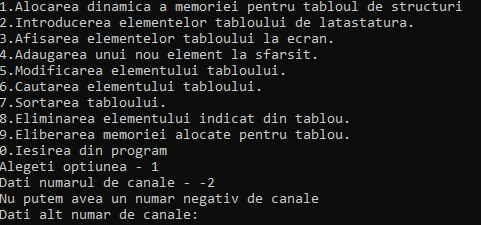


Adăugarea unui canal nou și editarea unui canal

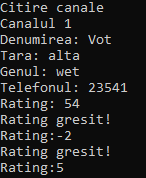


La început cautara unui element al tabloului și la sfârșit eliminarea unui element.

**Prevederea greșelilor ce pot apărea de la introducerea datelor de la tastatura**



Aici este demonstrația că în program sunt prevăzute unele greșeli care pot fi introduse de la tastatura, ca exemplu numarul negativ de canale.



Aici demonstratia ca nu putem da rating care nu se afla între 0-10, ce e logic.

**Concluzii pe baza efectuării lucrării de laborator:**

* Structurile ajuta imens la operarea cu multa informație, variabile, adică ușurarea și mărirea vitezei lucrului.
* Structurile in general reprezintă o categorie de date, cum ar fi Mobilă, în care pot intra mai multe elemente independente, masă, scaun, dulap, ș.a. Este cu mult mai ușor de operat cu toate aceste elemente ele fiind sistematizate și unite într-o categorie mai mare, la fel și cu structurile în programare, operarea mai ușoară cu mai multe elemente.
* Deci studiul structurilor va fi unul foarte util pentru viitor, îndeosebi în lucrul cu foartă multă informație și necesitatea de a o uni și sistematiza.

**Surse**

<https://www.scrigroup.com/calculatoare/c/Tablouri-de-structuri94976.php>