

Laboratorul 12

1. Exemplu de programare.

Problema

Compararea a două liste. Vi se cere să scrieți un program pentru o aplicație care nu tolerează date eronate la intrare. Pentru aceasta, datele sunt pregătite introducându-le de două ori într-un fișier. Fișierul conține două liste de numere întregi pozitive, separate de un număr negativ. Cele două liste trebuie să fie identice. În cazul în care nu sunt, înseamnă că a aparut o eroare la introducerea datelor. De exemplu, dacă fișierul de intrare conține secvența de numere 17, 14, 8, -5, 17, 14, 8, atunci cele două liste de numere sunt identice. Secvența 17, 14, 8, -5, 17, 12, 8 ilustrează o eroare la introducerea datelor pentru că penultima valoare nu este aceeași în ambele liste. Trebuie să afișați perechile de numere care nu sunt la fel. Nu se cunoaște numărul întregilor din fiecare listă, dar acesta nu este mai mare de 500.

Intrare

Un fișier (`dataFile`) care conține două liste de întregi pozitivi. Listele sunt separate de un întreg negativ și au lungimi egale.

Ieșire

Mesajul că cele două liste sunt identice, sau o lista de perechi de valori care nu se potrivesc.

Discuție

Pentru că cele două liste se găsesc în același fișier, prima listă trebuie citită și memorată până când se întâlnește un număr negativ. Cea de-a doua listă poate fi citită și comparată în același timp. Dacă am face această prelucrare manual, am scrie numerele din prima listă pe o bucată de hârtie, unul sub altul. Numărul liniei corespunde poziției numărului în listă. Aceasta înseamnă că primul număr este pe prima linie, al doilea număr pe a doua linie etc. Primul număr din cea de-a doua listă trebuie comparat cu numărul de pe prima linie, al doilea număr cu numărul de pe a doua linie, etc. Vom folosi un tablou numit `firstList` pentru a reprezenta bucata de hârtie. Declarația lui este următoarea:

```
const int MAX_NUMBER = 500; //maximul din fiecare lista
int firstList[MAX_NUMBER]; //pastreaza prima lista
```

Pentru ca primă componentă a tabloului are indicele 0, trebuie să gândim bucata de hârtie ca având numerotate liniile de la 0, nu de la 1.

Ipoteză

Cele două liste au lungimi egale.

Structuri de date

Un tablou unidimensional de tip `int` (`firstList`) care păstrează numerele din prima listă.

Exemplu de rulare a programului

Conținutul fișierului `liste.dat`:

21
32
76
-3
21
132
67

Programul afișează:

Poziția 1: 32 != 132
Poziția 2: 76 != 67

Programul

```
//*****  
//CheckLists.cpp  
//Acest program compara doua liste de intregi  
//dintr-un fisier de date  
//*****  
#include <iostream>  
#include <iomanip>  
#include <fstream>  
using namespace std;  
  
const int MAX_NUMBER = 500;  
  
void CitestePrimaLista( int[], int&, ifstream& );  
void ComparaListele( bool&, const int[], int, ifstream& );  
  
int main()  
{  
    int firstList[MAX_NUMBER]; //Pastreaza prima lista  
    bool allOK; //True daca listele sunt identice  
    int length; //Lungimea primei liste  
    ifstream dataFile; //Fisierul de intrare  
  
    dataFile.open("liste.dat");  
    if( !dataFile )  
    {  
        cout << "Eroare: Fisierul de date nu poate fi deschis"  
              << endl;  
        return 1;  
    }  
  
    CitestePrimaLista(firstList, length, dataFile);  
    allOK = true;  
    ComparaListele(allOK, firstList, length, dataFile);  
    if(allOK)  
        cout << "Cele doua liste sunt identice" << endl;  
  
    return 0;  
}
```

```
void CitestePrimaLista(int firstList[], //Prima lista
                      int& length, //Numarul de valori
                      ifstream& dataFile)
{
    int counter;
    int number;

    counter = 0; //Variabila index
    dataFile >> number; //Variabila folosita pentru citire
    while(number >= 0)
    {
        firstList[counter] = number;
        counter++;
        dataFile >> number;
    }
    length = counter;
}

void ComparaListele(bool& isOK, //true daca listele coincid
                    const int firstList[],
                    int length,
                    ifstream& dataFile)
{
    int counter; //Variabila de control al buclei
    int number; //Variabila folosita pentru citire

    for(counter = 0; counter < length; counter++)
    {
        dataFile >> number;
        if(number != firstList[counter])
        {
            isOK = false;
            cout << "Pozitia " << counter << ": "
                  << setw(4) << firstList[counter] << " != "
                  << setw(4) << number << endl;
        }
    }
}
```

2. Declarați tablouri unidimensionale conform următoarelor descrieri:

- a) Tablou cu 24 de elemente de tip float;
- b) Tablou cu 500 de elemente de tip int;
- c) Tablou cu 50 de elemente de tip virgulă mobilă în dublă precizie;
- d) Tablou cu 10 elemente de tip char.

3. Care este ieșirea programului următor? Datele necesare programului sunt date mai jos.

```
#include <iostream>
using namespace std;
```

```
int main()
{
    int a[100];
    int b[100];
    int j;
    int m;
    int sumA;
    int sumB;
    int sumDiff;
    sumA = 0;
    sumB = 0;
    sumDiff = 0;

    cin >> m;
    j = 0;
    while(j < m)
    {
        cin >> a[j] >> b[j];
        sumA = sumA + a[j];
        sumB = sumB + b[j];
        sumDiff = sumDiff + (a[j] - b[j]);
        j++;
    }
    j = m - 1;
    while(j >= 0)
    {
        cout << a[j] << ' ' << b[j] << ' '
              << a[j] - b[j] << endl;
        j--;
    }
    cout << endl;
    cout << sumA << ' ' << sumB << ' ' << sumDiff << endl;

    return 0;
}
```

Datele de intrare:

```
5
11 15
19 1
4 2
17 6
1 3
```

4. Cineva a scris următorul fragment de cod cu intenția de a afișa 10 20 30 40.

```
int arr[4] = {10, 20, 30, 40};
int index = 1;
while(index<=4)
{
    cout << ' ' << arr[index];
    index++ ;
}
```

Acest cod afișează, însă, 20 30 40 24835. Explicați unde este greșeala.

5. Fie declarațiile

```
int sample[8];  
int i;  
int k;
```

Arătați care este conținutul tabloului sample după executarea următorului fragment de cod:

```
k = 0;  
while(k < 8)  
{  
    sample[k] = 10 - k;  
    k++;  
}
```

6. Pentru următoarele patru exerciții folosiți declarațiile:

```
const int MAX_STUD = 100; //Numarul maxim de studenti  
bool failing[MAX_STUD];  
bool passing[MAX_STUD];  
int grade;  
int length;  
int score[MAX_STUD];
```

Puteți declara și alte variabile de care considerați că aveți nevoie.

Scrieți o funcție care primește `passing`, `score` și `length` ca parametri. Setează componenta lui `passing` cu `true` peste tot unde valoarea corespunzătoare a lui `score` este mai mare decât 60.

7. Scrieți o funcție numită `PassTally` care întoarce o valoare și care îi ia pe `passing` și `length` ca parametri. Această funcție returnează numărul componentelor din `passing` care au valoarea `true`.

8. Scrieți o funcție numită `Error` care întoarce o valoare și care îi ia pe `passing`, `failing` și `length` ca parametri. `Error` returnează `true` dacă vreuna dintre valorile pereche din `failing` și `passing` sunt la fel.

9. Scrieți o funcție care îi ia pe `score`, `passing`, `grade` și `length` ca parametri. Funcția trebuie să seteze componentele din `passing` cu `true` atunci când valorile corespunzătoare din `score` sunt mai mari decât `grade`.

10. Rescrieți programul de comparare a două liste din exemplul de programare eliminând presupunerea că cele două liste au aceeași lungime. Dacă o listă este mai lungă decât cealaltă, tipăriți un mesaj prin care să indicați care este mai lungă și opriți comparația.

11. Scrieți un program care calculează media aritmetică și deviația standard a întregilor stocați într-un fișier. Ieșirea va fi de tip `float`, trebuie să fie etichetată sugestiv și să fie formatată astfel încât să aibă două poziții zecimale. Deviația standard se calculează după formula

$$S = \sqrt{\frac{\sum_{i=1}^N (X_i - \bar{X})^2}{N-1}}$$

în care $\bar{X}$ reprezintă media aritmetică a numerelor $X_1, X_2, \dots, X_N$.

12. Examenul final al unui curs constă din 30 de întrebări la un test grilă. Profesorul v-a rugat să dacă scrieți un program care verifică automat răspunsurile studenților.

Intrări: Prima linie de date conține răspunsurile corecte la test. Răspunsurile corecte sunt primele 30 de caractere. Ele sunt urmate de un număr întreg care spune câți studenți au dat examenul (n). Următoarele n linii conțin răspunsurile studenților pe câte 30 de poziții, urmate de numele studentului pe următoarele 10 poziții.

Ieșiri: Pentru fiecare student se afișează numele studentului urmat de numărul răspunsurilor corecte, urmat de calificativul ADMIS dacă răspunsurile corecte reprezintă mai mult de 60% din total. În caz contrar, studentul este respins.

13. *Sistem de rezervare a locurilor pentru o companie aeriană.* O companie aeriană de mici dimensiuni a cumpărat un calculator pentru noul său sistem de rezervare automată a locurilor. Vi se cere să scrieți programul pentru acest sistem. Vi se cere să scrieți un program care asignează locurile fiecărui zbor al unui avion al companiei. Avionul are o capacitate de 10 locuri, câte două pe fiecare rând. Locurile sunt numerotate astfel: 1A, 1B, 2A, 2B, 3A, 3B, 4A, 4B, 5A și 5B.

Programul trebuie să tipărească următorul meniu:

Tipariti 1 pentru "Clasa întâi"

Tipariti 2 pentru "Clasa economica"

Dacă persoana tipărește 1, programul trebuie să îi atribuiască un loc în secțiunea clasei întâi – locurile 1A, 1B, 2A, 2B. Dacă persoana tipărește 2, atunci programul trebuie să îi atribuiască un loc în secțiunea economică – locurile 3A, 3B, 4A, 4B, 5A, 5B. Pentru îmbarcare, programul trebuie să tipărească o carte de îmbarcare pentru fiecare pasager. Pe cartea de îmbarcare este trebuit locul și categoria acestuia: clasa întâi sau economic.

Folosiți un tablou bidimensional pentru a reprezenta schema locurilor în avion. Inițializați toate elementele matricii cu 0 pentru a arăta că toate locurile sunt libere. Pe măsură ce locurile se ocupă, asigurați elementelor corespunzătoare ale matricii valoarea 1 pentru a marca locurile care nu mai sunt disponibile.

Programul nu va permite rezervarea unui loc care este deja vândut. Dacă locurile de la una dintre clase s-au epuizat iar la cealaltă clasa mai sunt locuri, trebuie să întrebe persoana dacă este de acord cu schimbarea categoriei. Dacă nu, se afișează mesajul

Următorul zbor este peste 3 ore.