# PROIECT NR. 5: ȘIRURI DE CARACTERE. FIȘIEE TEXT.

A elaborat: Corina CERNOLEV, gr. IS11Z

A verificat: Vitalie ŢÎCĂU, lector universitar

## 1. Formularea problemei

De trasat tabelele operaţiilor:

1. De determinat numărul de cuvinte dintr-o frază;
2. De determinat cuvintele ce conțin două litere indicate;
3. De determinat cuvintele ce încep cu o literă majusculă;
4. De determinat numărul de cuvinte dintr-un fișier;
5. De determinat cuvintele din fișier ce conțin două litere indicate;
6. De determinat cuvintele ce încep cu o literă majusculă din fișier;
7. De transferat cuvintele ce conțin literele indicate în alt fișier;
8. De eliminat cuvintele care conțin două litere indicate din fișier.

## 2. Studiu preliminar

În lucrul cu fisierele se pot efectua următoarele operații:

1. Deschiderea fișierelor,
2. Închiderea fișierelor,
3. Citirea din fișier,
4. Înscrierea în fișier.

Strtok() – desparte fraza în cuvinte.

Strcpy() – efectuiaza o copie a cuvântului.

## 3. Programul

#include <iostream>

#include <iomanip>

#include <cstdlib>

#include <conio.h>

#include <string.h>

#include <stdio.h>

#include <cstring>

#define LMAX 255

int NumarCuvinte(char sir[]);

int CuvinteLitere(char sir[], char litera1, char litera2);

int Majuscule(char sir[]);

int CuvinteMajuscule(char sir[]);

void TransferCuvinteLitera(char sir[], char litera1, char litera2);

int EliminaCuvinteLitera(char sir[], char litera1, char litera2);

void ViewFile(char nume[30], char sir[]);

using namespace std;

char sir[LMAX];

int main()

{

    int optiune, nr;

    char fraza[LMAX] = "Dumitru este prieten cu Mihai, Mihai prieteneste cu Corina, rezulta...\nDumitru este prieten cu Mihai, Mihai prieteneste cu Corina, rezulta...";

    do

    {

        system("cls");

        cout << "Proiect nr. 5, elaborat de Cernolev Corina IS11Z\n";

        cout << "Tema: Siruri de caractere. Fisiere text";

        cout << "\n 1. Numarul de cuvinte dintr-o fraza";

        cout << "\n 2. Cuvinte ce contin cel putin 2 litere indicate";

        cout << "\n 3. Cuvintele ce incep cu litera majuscula";

        cout << "\n 4. Numarul de cuvinte dintr-un fisier";

        cout << "\n 5. Cuvintele din fisier ce contin cel putin 2 litere indicate";

        cout << "\n 6. Cuvintele ce incep cu litere majuscule din fisier";

        cout << "\n 7. Transfer cuvinte ce contin literele indicate in alt fisier";

        cout << "\n 8. Eliminare cuvinte cu 2 litere indicate din fisier";

        cout << "\n 0. Stop";

        cout << "\n\n\t Alege (de la 0 la 9) -> ";

        cin >> optiune;

        switch (optiune)

        {

        case 1:

            fflush(stdin);

            cout << "Introduceti o fraza: ";

            gets(fraza);

            cout << "Fraza initiala:\n";

            puts(fraza);

            if ((nr = NumarCuvinte(fraza)) > 0)

                cout << "In total " << nr << " cuvinte\n";

            else

                cout << "Nu-s cuvinte!\n";

            break;

        case 2:

            cout << "Fraza initiala:\n";

            puts(fraza);

            char litera1, litera2;

            cout << "Dati literele cautate ";

            cin >> litera1 >> litera2;

            if ((nr = CuvinteLitere(fraza, litera1, litera2)) > 0)

                cout << "\n In total " << nr << " cuvinte care contin literele indicate\n";

            else

                cout << "\n Nu-s cuvinte care contin literele indicate \n";

            break;

        case 3:

            cout << "Fraza initiala:\n";

            puts(fraza);

            if ((nr = CuvinteMajuscule(fraza)) > 0)

                cout << "\nIn total " << nr << " cuvinte cu litere majuscule\n";

            else

                cout << "Nu-s cuvinte cu litere majuscule!\n";

            break;

        case 4:

            ViewFile("fraza.txt", sir);

            cout << "\nIn fisier sunt: " << NumarCuvinte(sir) << " cuvinte.";

            break;

        case 5:

            ViewFile("fraza.txt", sir);

            cout << "\n Dati literele cautate ";

            cin >> litera1 >> litera2;

            if ((nr = CuvinteLitere(sir, litera1, litera2)) > 0)

                cout << "\n In total " << nr << " cuvinte care contin literele indicate\n";

            else

                cout << "\n Nu-s cuvinte care contin literele indicate \n";

            break;

        case 6:

            ViewFile("fraza.txt", sir);

            if ((nr = CuvinteMajuscule(sir)) > 0)

                cout << "\nIn total " << nr << " cuvinte cu litere majuscule\n";

            else

                cout << "\nNu-s cuvinte cu litere majuscule!\n";

            break;

        case 7:

            ViewFile("fraza.txt", sir);

            cout << "Dati literele cautate ";

            cin >> litera1 >> litera2;

            TransferCuvinteLitera(sir, litera1, litera2);

            break;

        case 8:

            ViewFile("fraza.txt", sir);

            cout << "Dati literele cautate ";

            cin >> litera1 >> litera2;

            EliminaCuvinteLitera(sir, litera1, litera2);

            break;

        }

        getch();

    } while (optiune);

    return 0;

}

int NumarCuvinte(char sir[])

{

    int nr = 0;

    char \*cuvant, copia[LMAX];

    strcpy(copia, sir);

    if ((cuvant = strtok(copia, ",.!\n\t:? ")) == NULL)

        return 0;

    nr++;

    while ((cuvant = strtok(NULL, ",.!\n\t:? ")) != NULL)

        nr++;

    return nr;

}

int CuvinteLitere(char sir[], char litera1, char litera2)

{

    char \*cuvant, copia[LMAX];

    strcpy(copia, sir);

    int nr = 0;

    if ((cuvant = strtok(copia, ",.!?:\n\t ")) == NULL)

        return 0;

    if ((strchr(cuvant, litera1)) && (strchr(cuvant, litera2)))

    {

        cout << setw(16) << cuvant;

        nr++;

    }

    while ((cuvant = strtok(NULL, ",.!?:\n\t ")) != NULL)

    {

        if ((strchr(cuvant, litera1)) && (strchr(cuvant, litera2)))

        {

            cout << setw(16) << cuvant;

            nr++;

        }

    }

    return nr;

}

int Majuscule(char sir[])

{

    for (int i = 0; i < strlen(sir); i++)

    {

        if (islower(sir[i]))

            return 0;

        return 1;

    }

}

int CuvinteMajuscule(char sir[])

{

    int nr = 0;

    char \*cuvant, copia[LMAX];

    strcpy(copia, sir);

    if ((cuvant = strtok(copia, ",.!?:\n\t ")) == NULL)

        return 0;

    if (Majuscule(cuvant))

    {

        cout << setw(16) << cuvant;

        nr++;

    }

    while ((cuvant = strtok(NULL, ",.!\n\t:? ")) != NULL)

        if (Majuscule(cuvant))

        {

            cout << setw(16) << cuvant;

            nr++;

        }

    return nr;

}

void TransferCuvinteLitera(char sir[], char litera1, char litera2)

{

    FILE \*f;

    int n =0;

    f = fopen("cuvTrans.txt", "w");

    char \*cuvant, copia[LMAX];

    strcpy(copia, sir);

    if ((cuvant = strtok(copia, ",.!?:\n\t ")) == NULL)

        return;

    if ((strchr(cuvant, litera1)) && (strchr(cuvant, litera2)))

    {

        fputs(cuvant, f);

        fputs(" ", f);

        n++;

    }

    while ((cuvant = strtok(NULL, ",.!?:\n\t ")) != NULL)

    {

        if ((strchr(cuvant, litera1)) && (strchr(cuvant, litera2)))

        {

            fputs(cuvant, f);

            fputs(" ", f);

            n++;

            if(n%5 == 0)

                fputs("\n", f);

        }

    }

    fclose(f);

    char cuvTrans[LMAX];

    ViewFile("cuvTrans.txt", cuvTrans);

}

int EliminaCuvinteLitera(char sir[], char litera1, char litera2)

{

    FILE \*f;

    int n =0;

    f = fopen("fraza.txt", "w");

    char \*cuvant, copia[LMAX];

    strcpy(copia, sir);

    if ((cuvant = strtok(copia, ",.!?:\n\t ")) == NULL)

        return 0;

    if (!((strchr(cuvant, litera1)) && (strchr(cuvant, litera2))))

    {

        fputs(cuvant, f);

        fputs(" ", f);

        n++;

    }

    while (((cuvant = strtok(NULL, ",.!?:\t ")) != NULL))

    {

        if (!((strchr(cuvant, litera1)) && (strchr(cuvant, litera2))))

        {

            fputs(cuvant, f);

            fputs(" ", f);

            n++;

        }

    }

    fclose(f);

    char cuvTrans[LMAX];

    ViewFile("fraza.txt", cuvTrans);

}

void ViewFile(char nume[30], char sir[])

{

    char sir2[LMAX];

   FILE \*f;

   f = fopen(nume, "r");

   while(!feof(f))

    {

        fread(sir2,sizeof(sir2),1,f);

        strcat(sir, sir2);

    }

    puts(sir);

   fclose(f);

}

## 4. Concluzii

1. Functia fscanf() citește din fișier până la spațiu;

2. Cu ajutorul funcției fseek(), putem muta cursorul în cadrul fișierului la poziția necesară;

3. În cazul în care avem în fișier informație de tipul string, putem utiliza funcția fgets().