MINISTERUL EDUCAȚIEI AL REPUBLICII MOLDOVA

UNIVERSITATEA TEHNICĂ A MOLDOVEI

**RAPORT**

LUCRAREA DE LABORATOR NR. 5

LA PROGRAMAREA CALCULATOARELOR

Elaborat: st. gr. TI – 211, Popa Ecaterina

Verificat: Stratulat Ștefan

UTM, Chișinău, 2021

**Scopul lucrării:**

Scopul lucrării este de a familiariza studentul cu procesarea și utilizarea șirurilor de caractere, utilizînd limbajul C.

**Probleme pentru șiruri de caractere:**

1. Scrieți un program C pentru a afla lungimea unui șir.

#include <stdio.h>

#include <string.h>

void main ()

{

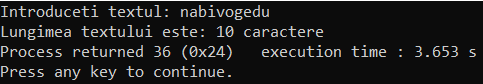
char text [100];

printf ("Introduceti textul: ");

gets(text);

printf ("Lungimea textului este: %d caractere", strlen(text));

}



1. Scrieți un program C pentru a copia un șir în altul.

#include <stdio.h>

#include <string.h>

#include <ctype.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

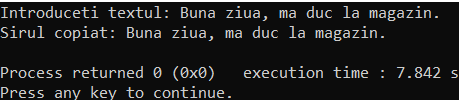
gets(text);

strcpy(text\_1, text);

printf ("Sirul copiat: ");

puts(text\_1);

}



1. Scrieți un program C pentru a concatena două șiruri.

#include <stdio.h>

#include <string.h>

#include <ctype.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti sirul 1: ");

gets(text);

printf ("Introduceti sirul 2: ");

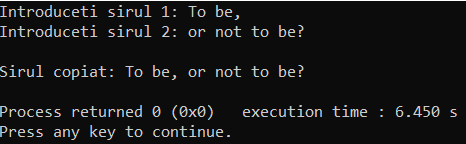
gets(text\_1);

strcat(text, text\_1);

printf ("\nSirul copiat: ");

puts(text);

}



1. Scrieți un program C pentru a compara două șiruri.

#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

int flag;

printf ("Introduceti sirul 1: ");

gets(text);

printf ("Introduceti sirul 2: ");

gets(text\_1);

flag = strcmp(text, text\_1);

printf ("%d\n", flag);

if (flag>0)

printf ("Sirul 1 este mai mic decat sirul 2");

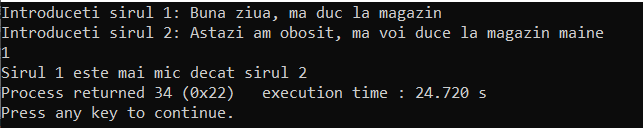
else if (flag==0)

printf ("Sirurile sunt egale");

else

printf ("Sirul 1 este mai mare decat sirul 2");

}



1. Scrieți un program C pentru a converti dintr-un șir, literele minuscule în majuscule.

#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti sirul: ");

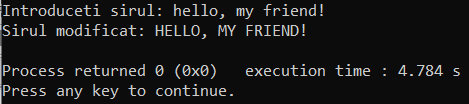
gets(text);

strupr(text);

printf ("Sirul modificat: ");

puts(text);

}



1. Scrieți un program C pentru a converti dintr-un șir, literele majuscule în litere mici.

#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti sirul: ");

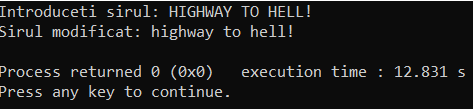
gets(text);

strlwr(text);

printf ("Sirul modificat: ");

puts(text);

}



1. Scrieți un program C pentru a comuta literele minuscule în majuscule și majuscule în minuscule pentru fiecare caracter dintr-un șir.

#include <stdio.h>

#include <string.h>

#include <ctype.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

str(text);

printf ("Text modificat: ");

puts(text);

}

void str(char text[], char text\_1[])

{

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

{

if (text[i]>=65 && text[i]<=90)

{

text[i] = text[i] + 32;

}

else if (text[i]>=97 && text[i]<=122)

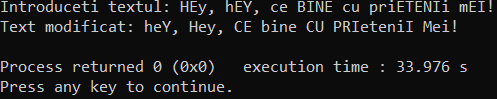
{

text[i] = text[i] - 32;

}

}

}



1. Scrieți un program C pentru a număra numărul total de vocale dintr-un șir.

#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

int count = 0;

for (int j=0; str[j]; j++)

{

if (str[j]=='a'||str[j]=='A'||str[j]=='o'||str[j]=='O'||str[j]=='E'||str[j]=='e'||str[j]=='i'||str[j]=='I'||str[j]=='U'||str[j]=='u')

count++;

}

printf ("In text sunt %d vocale", count);

}

void main ()

{

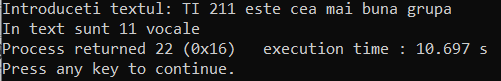
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



1. Scrieți un program C pentru a găsi inversul unui șir.

#include <stdio.h>

#include <string.h>

void main ()

{

char str1[100], i;

printf ("Introduceti textul: ");

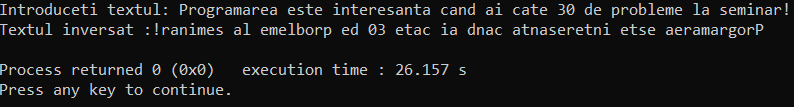
gets(str1);

strrev(str1);

printf ("Textul inversat :");

puts(str1);

}



1. Scrieți un program C pentru a verifica dacă un șir este palindrom sau nu.

#include <stdio.h>

#include <string.h>

void main ()

{

char text[100];

char text\_1[100];

printf ("Introduceti textul: ");

gets(text);

strcpy(text\_1, text);

strrev(text\_1);

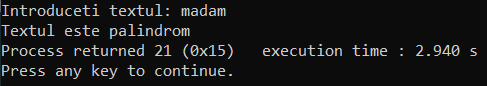
if (!strcmp(text, text\_1))

printf ("Textul este palindrom");

else

printf ("Textul nu este palindrom");

}



**Întrebări de verificare**

* Ce reprezintă un șir de caractere în limbajul C?

În limbajul C, șirurile de caractere sunt ca un tablou unidimensional de elemente de char[].

* Cu ce simbol se finisează un șir de caractere în limbajul C?

Ultimul element din șir este reprezentat prin caracterul '\0' sau valoarea 0. –

* Cum se inițializează un șir de caractere la definirea acestuia ?
* Sintaxa generală pentru definirea unui șir de caractere este:

***char nume\_sir[nr\_el];***

nume\_sir - reprezintă numele variabilei de tip șir de caractere,

nr\_el – reprezintă numărul de elemente/caractere din șir, inclusiv terminatorul de șir.

* Cum citim de la tastatură un șir de caractere cu spații?

La utilizarea funcției gets(), aceasta citește caractere până când este apăsată tasta ENTER, aceasta fiind înlocuită de compilator cu caracterul '\0'.

* Cum aflăm lungimea unui șir de caractere?

Lungimea unui șir poate fi dedusă utilizând funcția bibliotecii <string.h> strlen(str);

* Cum aflăm numărul de cuvinte într-un șir de caractere?

Vom inițializa o variabilă count = 0 și o funcție, care la fiecare spațiu dintre cuvinte va incrementa variabila count.