Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Catedra Informatica si Ingineria Sistemelor

**RAPORT**

Lucrarea de laborator nr.6

la Programarea Calculatoarelor

A efectuat:

st. gr. C-171 D. Melniciuc

A verificat:

dr., conf.univ. M. Kulev

Chişinău 2017

**Lucrarea de laborator nr.**

**Tema:** **Prelucrarea datelor de tip String**

**Scopul lucrarii practice:**

Studierea, utilizarea instrucţiunilor de bază si acumularea deprinderilor de algoritmizare si programare .a problemelor ce prelucreaza datele de tip Sir de caractere;

Introducerea si afisarea sirurilor de caractere, prelucrarea lor, folosind functiile si procedurile de prelucrarea a sirurilor ce caractere

**Notiuni de baza:**

C are un sistem de [tipuri de date](https://ro.wikipedia.org/w/index.php?title=Tip_de_date&action=edit&redlink=1) similar cu cel al descendenților [ALGOL](https://ro.wikipedia.org/wiki/ALGOL), cum ar fi [Pascal](https://ro.wikipedia.org/wiki/Pascal), dar totuși cu anumite diferențe. Cuprinde tipuri de date cum ar fi [întregi](https://ro.wikipedia.org/w/index.php?title=Num%C4%83r_%C3%AEntreg_(informatic%C4%83)&action=edit&redlink=1) de diferite dimensiuni, cu sau fără semn, [numere în virgulă mobilă](https://ro.wikipedia.org/w/index.php?title=Num%C4%83r_%C3%AEn_virgul%C4%83_mobil%C4%83&action=edit&redlink=1), [enumerări](https://ro.wikipedia.org/w/index.php?title=Enumerare_(informatic%C4%83)&action=edit&redlink=1) (enum), structuri de date (struct) și [uniuni](https://ro.wikipedia.org/w/index.php?title=Uniune_(informatic%C4%83)&action=edit&redlink=1) (union).

C utilizează foarte mult pointerii, un tip de [referință](https://ro.wikipedia.org/w/index.php?title=Referin%C8%9B%C4%83_(informatic%C4%83)&action=edit&redlink=1) foarte simplu, care păstrează [adresa](https://ro.wikipedia.org/w/index.php?title=Adres%C4%83_de_memorie&action=edit&redlink=1) unui obiect din memorie. Adresa poate fi manipulată cu ajutorul [aritmeticii pointerilor](https://ro.wikipedia.org/w/index.php?title=Aritmetica_pointerilor&action=edit&redlink=1). În momentul compilării, un pointer este un tip de dată complex, ce reprezintă atât adresa de memorie cât și tipul de dată. Acest lucru permite expresiilor ce utilizează pointeri să fie [evaluate](https://ro.wikipedia.org/w/index.php?title=Evaluare_(programare)&action=edit&redlink=1) după tipul de dată. Pointerii au mai multe utilizări în C. De exemplu, [șirurile de caractere](https://ro.wikipedia.org/w/index.php?title=%C8%98ir_de_caractere&action=edit&redlink=1) (engleză *text string*) sunt adesea reprezentate printr-un pointer la un [vector](https://ro.wikipedia.org/w/index.php?title=Vector_(structur%C4%83_de_date)&action=edit&redlink=1) de [caractere](https://ro.wikipedia.org/wiki/Caracter). [Alocarea dinamică a memoriei](https://ro.wikipedia.org/w/index.php?title=Alocare_dinamic%C4%83_a_memoriei&action=edit&redlink=1) este realizată tot cu ajutorul pointerilor.

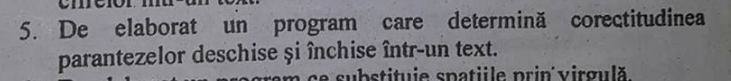
Un pointer *null* are o valoare rezervată, indicând faptul că face referire la o locație nevalidă. Acest lucru este folositor în cazuri speciale cum ar fi pointerul *next* (următorul) în nodul final al unei [liste înlănțuite](https://ro.wikipedia.org/wiki/List%C4%83_(structur%C4%83_de_date)). [Dereferențierea](https://ro.wikipedia.org/w/index.php?title=Dereferen%C8%9Biere&action=edit&redlink=1) unui pointer *null* poate cauza un comportament imprevizibil al aplicației. De asemenea, există și pointeri de tip void, fapt ce indică referirea la un obiect de tip necunoscut.

De asemenea, C oferă posibilitatea de lucru cu tablouri multidimensionale. Din punct de vedere [semantic](https://ro.wikipedia.org/w/index.php?title=Semantic%C4%83_(limbaje_formale)&action=edit&redlink=1), tablourile multidimensionale sunt tablouri de tablouri.

**Condiţia problemei (sarcina de lucru):**

De executat pe calculator programul de prelucrrare a datelor de tip sir de caractere, in corespundere cu varianta. De afisat Sirul de date introdus si prelucrat:

**Varianta 5:**



**Mersul lucrarii:**

[2]Pentru a programa la calculator e necesara urmarirea unui algoritm strict. Pasii sunt urmatorii:

-cunoasterea problemei;

Citirea si intelegerii corecte a problemei.

-analiza datelor;

Studierea datelor problemei pentru a destinge datele de tip input, output si date intermediare.

-elaborarea algoritmului prin schema logica;

Elaborarea schemei logice a programului.

-codul (textului) programei;

Scrierea codului propriu-ziz.

-testarea;

Rularea problemei pentru a determina erori logice sau de sintaxa.

-verificarea rezultatelor

Introducerea diferitor date pentru a verifica mai multe posibilitati.

-analiza rezultatelor si formarea concluziilor;

Formare concluziilor adecvate dupa efectuarea pasilor anteriori.

Analiza datelor:

1. Cream un subprogram care va contine rezolvarea conditiei;
2. In functia main doar dam textul si scriem un cod ce va arata daca e corecta sau nu textul introdus intrucit acesta trebuie sa aiba numar de paranteze inchise egal cu cele deschise;

**Schema logica:**

calc

“Dati textul”

fflush(stdin)

“Correct”

“Gresit“

gets(text)

**`**

n1++

text[i]=='('

text[i]==')'

n2++

i=0; i<n; i++

n1==n2

return 0

return 1

**Codul (textul) programului in limbajul C:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <conio.h>

int calc(char \*text)

{

int i;

int n1 = 0;

int n2 = 0;

for (i=0; text[i]; i++)

{

if(text[i]=='(')

n1++;

if(text[i]==')')

n2++;

}

if(n1 == n2) return 1;

else return 0;

}

int main()

{

char text[100];

printf("Dati textul\n\r");

fflush(stdin);

gets(text);

if (calc(text))

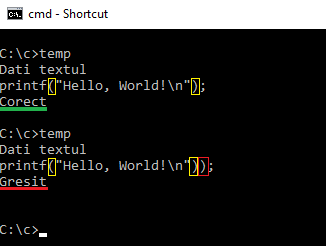
printf("Corect\n\r");

else

puts("Gresit\n\r");

return 0;

}

****

**Rezultatele obtinute:**

Putem observa ca primim mesajul „Corect”

atunci cind avem numar egal de

paranteze inchise si deschise.

Cind numarul de paranteze inchise si deschise

nu este egal primim mesajul „Gresit”

**Bibliografie:**

[1] Derevlenco, Vasile și alț. Programare : Indicaţii de laborator. Partea 1. – Ch.: Tehnica – UTM, 2014. <http://library.utm.md>

[2]Conspectul si notele de la lectiile teoretice

**Concluzie:**

In urma efectuarii programului si a rularii acestuia au fost obtinute anumite abilitati in lucrul cu Sirurile de caractere.

**Avantaje**

Programul da teste interactiv.

Introducem textul care vrem.

**Dezavantje**

Determinarea numarului de paranteze este tot ce face programul dat.

**Dezvoltarea programului**

Micsorarea numarului liniilor din cod.

Restructurarea codului pentru un nivel de citire mai usor.