**MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Departamentul Informatică şi Ingineria Sistemelor**

**RAPORT**

Lucrare de laborator nr. 6

la cursul „***Programarea Calculatoarelor***”

**Tema: Prelucrarea caracterelor şi a şirurilor de caractere**

A efectuat :  **St. gr. CR-221FR Serba Cristina**

**A verificat: Asis.univ. Toma Olga**

**Chișinău 2022**

**Cuprins**

[INTRODUCERE 2](#_Toc113529265)

[REALIZAREA PRACTICĂ A SARCINII DE LUCRU 3](#_Toc113529266)

[CONCLUZII 5](#_Toc113529267)

[BIBLIOGRAFIE 6](#_Toc113529268)

[ANEXE (dacă sunt) 7](#_Toc113529269)

[Anexa 1: *Denumirea anexei…* 7](#_Toc113529270)

# 

# INTRODUCERE

Șirurile de caractere sunt, in principiu, un vector unidimensional de caractere terminate cu un caracter nul „\0”. Fiecare element al șirului (începând cu 0) conține codul ASCII al caracterului pe care îl stochează. Ultimul element al șirului este caracterul nul care este stocat automat. String\_length+1 caractere (+1 pentru caracterul nul) trebuie rezervate.

Deoarece limbajul C nu are un tip de date șir predefinit, acestea sunt create manual, caracter cu caracter.

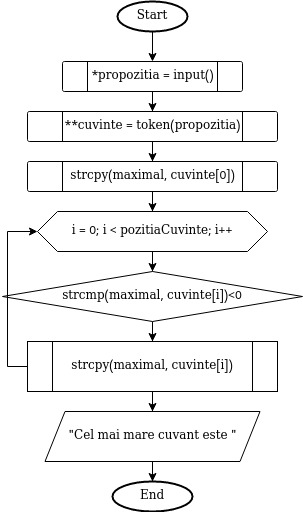
În biblioteca <string.h> există o mulțime de funcții care ajută la manipularea șirurilor, inclusiv strtok(), strcmp(), strcpy(). Prima, în special, va fi folosit pentru a împărți propoziția în cuvinte.

Folosind funcțiile predefinite enumerate, programul va căuta cel mai mare cuvânt lexicografic dintr-o propoziție propusă de utilizator.

# REALIZAREA PRACTICĂ A SARCINII DE LUCRU

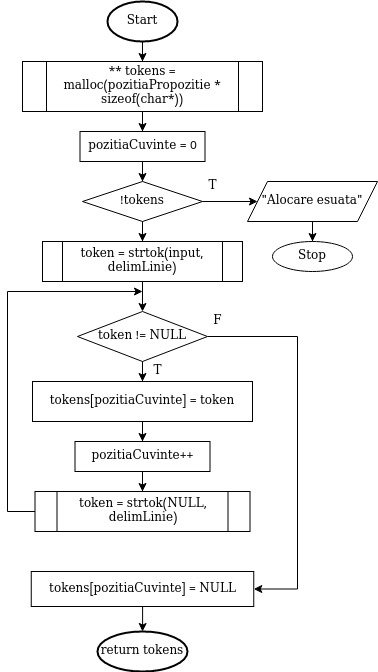
**Sarcina lucrării de laborator (varianta 2):**

Pentru propoziţia dată să se determine cel mai “mare” cuvânt în sens lexicografic.



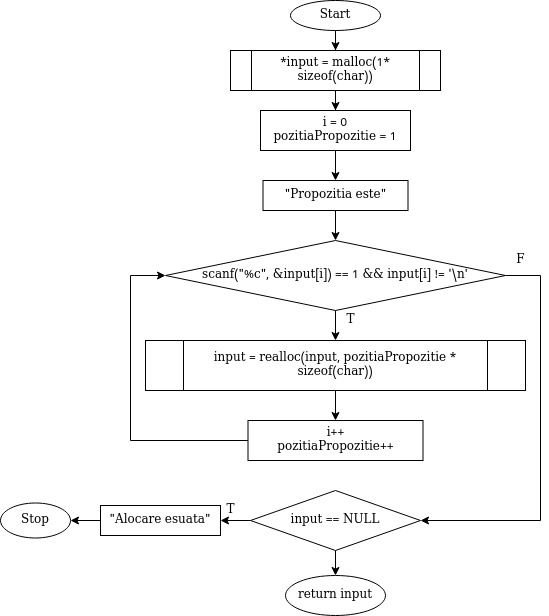
**Fig.1.1 Schema bloc main()**

Figura 1.1 reprezintă schema bloc a functiei principale main()



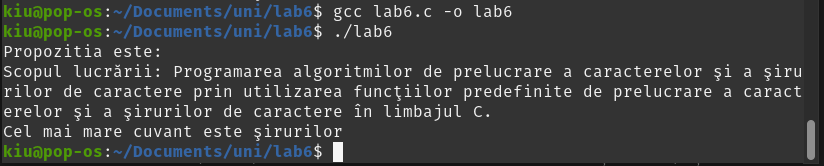
**Fig.1.2 *Schema bloc token()***

Figura 1.2 reprezintă schema bloc a functiei token()



**Fig.1.3 *Schema bloc input()***

Figura 1.3 reprezintă schema bloc a functiei input()



# CONCLUZII

Scopul lucrării date este de a prelucra șiruri de caractere.

Folosind funcții predefinite din biblioteca <string.h>, șirul original este modificat și, la rândul său, modifică variabilele de control. Deoarece șirurile conțin valorile ASCII ale fiecărui caracter, compararea fiecărui cuvânt în sens lexicografic este posibilă datorită funcției strcmp(), care compară fiecare caracter al cuvântului cu codul ASCII.

Împărțirea propoziției în cuvinte separate se face prin crearea unui vector de șiruri de caractere și a funcției strtok(). La crearea acestuia, este necesar să fie introdus caracterul final „\0” pentru a semnala sfârșitul vectorului.

De asemenea, programul folosește alocarea dinamică pentru a crea string-ul, de aceea utilizatorul este liber să introducă o propoziție fără limită prestabilită. Dacă resursele computerului sunt limitate, programul își va opri execuția și va afișa o eroare.

În final, prin lucrarea dată, am învățat să procesez șiruri de caractere începând cu propoziții până la cea mai mică structură - însăşi caracterul.

# BIBLIOGRAFIE

[***https://www.tutorialspoint.com/cprogramming/c\_strings.html***](https://www.tutorialspoint.com/cprogramming/c_strings.htm)