

# Sesiunea 6

---

## Agenda

- \* Exerciții pbinfo prelucrări siruri
- \* Exerciții prelucrări siruri din variante de BAC

## Exerciții pbinfo prelucrări siruri

- Exercițiile au fost extrase de pe pbinfo.
  - Link: [https://www.pbinfo.ro/?pagina=probleme-lista&disciplina=0&clasa=10&tag=10&subtag=11&dificultate=1&folosesc\\_consola=-1&eticheta=](https://www.pbinfo.ro/?pagina=probleme-lista&disciplina=0&clasa=10&tag=10&subtag=11&dificultate=1&folosesc_consola=-1&eticheta=)
- 1.
  - Link: <https://www.pbinfo.ro/probleme/4184/bacnume>
  - Solutie:

```
#include <iostream>
#include <cstring>

using namespace std;

void FNuma(char s[], char id[]);

int main() {
    char id[51];
    char s[51] = "David Popovici";
    FNuma(s, id);
    cout << id;
    return 0;
}

void FNuma(char s[], char id[]) {
    char* pch = strtok(s, " ");
    pch = strtok(NULL, " ");
    strcpy(id, pch);
    strcat(id, "2022");
}
```

- 2.
  - Link: <https://wExerciții prelucrări siruri din variante de BACe std;>

```
int main() {
```

```
    char s[256];
    cin.getline(s, 256);
```

```

char vocale[] = {'a', 'e', 'i', 'o', 'u', '\0'};
int contor =0;

for (int i = 1; i< strlen(s)-1;i++) {
    char literaStanga = s[i - 1];
    char literaDreapta = s[i + 1];
    char literaCurenta = s[i];
    if (literaStanga == ' ' || literaDreapta == ' ') {
        continue;
    }
    if (strchr(vocale, literaCurenta) != NULL &&
        (strchr(vocale, literaStanga) == NULL) &&
        (strchr(vocale, literaDreapta) == NULL)) {
        contor++;
    }
}
cout << contor;
return 0;

```

```

}

```

- Solutie 2 (refactorizare):

```

#include <iostream>
#include <cstring>

using namespace std;

int esteVocala(char ch);
int esteConsoana(char ch);

int main() {

    char s[256];
    cin.getline(s, 256);
    int contor =0;

    for (int i = 1; i< strlen(s)-1;i++) {
        char literaStanga = s[i - 1];
        char literaDreapta = s[i + 1];
        char literaCurenta = s[i];
        if (literaStanga == ' ' || literaDreapta == ' ') {
            continue;
        }

        if (esteVocala(literaCurenta) &&
            esteConsoana(literaDreapta) && esteConsoana(literaStanga)) {

```

```

        contor++;
    }
}
cout<< contor;
return 0;
}

int esteVocala(char ch) {
    return strchr("aeiou", ch) != NULL;
}

int esteConsoana(char ch) {
    return ch >= 'a' && ch <= 'z' && !esteVocala(ch);
}

```

3.
  - Link: <https://www.pbinfo.ro/probleme/11/vocale>
  - Solutie:

```

#include <iostream>
#include <cstring>

using namespace std;

int esteVocala(char ch);

int main() {
    char s[21];
    cin >> s;
    for (int i = 0; i < strlen(s); i++) {
        if (esteVocala(s[i])) {
            s[i] = s[i]-32;
        }
    }
    cout << s;
    return 0;
}

int esteVocala(char ch) {
    if(strchr("aeiou", ch)) {
        return 1;
    }
    return 0;
}

```

4.
  - Link: <https://www.pbinfo.ro/probleme/1866/prosir>
  - Solutie:

```

#include <iostream>
#include <fstream>
#include <cstring>

```

```

using namespace std;

int main() {
    ifstream fin("prosir.in");
    ofstream fout("prosir.out");
    char line[201];
    fin.getline(line, sizeof(line));
    int lineLength = strlen(line);
    for(int i = 0; i < lineLength; i++) {
        if(line[i+1] == ' ' && isalnum(line[i])) {
            line[i] = '5';
        }
    }
    line[lineLength-2] = '5';
    // fout.write(line, lineLength); // putem folosi si
    // functia write pentru a scrie in fisier
    fout << line;
    fin.close();
    fout.close();
    return 0;
}

```

5.
  - Link: <https://www.pbinfo.ro/probleme/1456/cuvant>
  - Solutie:

```

#include <iostream>
#include <cstring>

using namespace std;

int esteConsoana(char ch);
int esteVocala(char ch);

int main() {
    char cuvant[101];
    cin.getline(cuvant, 101);
    int areConsoane = 0;
    for(int i = 0; i < strlen(cuvant); i++) {
        if(strchr("aeou", cuvant[i])) {
            break;
        } else if(esteConsoana(cuvant[i])){
            areConsoane = 1;
        }
    }
    if (areConsoane) {
        cout << "DA";
    } else {
        cout << "NU";
    }
}

```

```

    }
    return 0;
}

int esteConsoana(char ch) {
    return ch> 'a' && ch <= 'z' && !esteVocala(ch);
}

int esteVocala(char ch) {
    if(strchr("aeiou", ch)) {
        return 1;
    }
    return 0;
}

```

6.   ◦ Link: <https://www.pbinfo.ro/probleme/2828/acronim>
- Solutie:

```

#include <iostream>
#include <cstring>
#include <cctype>

using namespace std;

int main()
{
    char cuvant[101];
    cin.getline(cuvant, 101);

    char* token = strtok(cuvant, " ");
    while (token != NULL) {
        if (isupper(token[0])) {
            cout << token[0];
        }
        token = strtok(NULL, " ");
    }

    return 0;
}

```

- Solutie2 (folosim propria functie pentru isUpper)

```

#include <iostream>
#include <cstring>

int esteMajuscula(char ch);

using namespace std;

```

```
int main()
{
    char cuvant[101];
    cin.getline(cuvant, 101);

    char* token = strtok(cuvant, " ");
    while (token != NULL) {
        if (esteMajuscula(token[0])) {
            cout << token[0];
        }
        token = strtok(NULL, " ");
    }

    return 0;
}

int esteMajuscula(char ch) {
    return ch >= 65 && ch <= 90;
}
```

7.   ◦ Link: <https://www.pbinfo.ro/probleme/890/nrvocale>  
      ◦ Solutie:

```
#include <iostream>
#include <cstring>
#include <cctype>

using namespace std;

int esteVocala(char);

int main()
{
    char cuvant[256];
    cin.getline(cuvant, 256);
    char vocale[5] = {'a', 'e', 'i', 'o', 'u'};
    int frecventaVocale[5] = {0};

    for (int i = 0; i < strlen(cuvant); i++) {
        if (esteVocala(cuvant[i])) {
            char vocala = tolower(cuvant[i]);
            /*if (vocala == 'a') {
                frecventaVocale[0]++;
            }
            else if (vocala == 'e') {
                frecventaVocale[1]++;
            }
            else if (vocala == 'i') {
                frecventaVocale[2]++;
            }
            else if (vocala == 'o') {
```

```

        frecventaVocale[3]++;
    }
    else {
        frecventaVocale[4]++;
    }*/
    switch (vocala) {
        case 'a':
            frecventaVocale[0]++;
            break;
        case 'e':
            frecventaVocale[1]++;
            break;
        case 'i':
            frecventaVocale[2]++;
            vocalaAparitiiMaximAparitiiMaxim = 'a'; // punem o
            valoare de inceput, si o adaptam pe parcurs
    }
    for (int i = 0; i < 5; i++) {
        if (frecventaVocale[i] > maximAparitii) {
            maximAparitii = frecventaVocale[i];
            vocalaAparitiiMaxim = (char)toupper(vocale[i]);
        }
    }

    cout << vocalaAparitiiMaxim;

    return 0;vocalaAparitiiMaxim

```

8.   ◦ Link: <https://www.pbinfo.ro/probleme/13/prefixe>  
       ◦ Solutie:

```

#include <iostream>
#include <cstring>

using namespace std;
int main() {
    char cuvant[11], auxiliar[11];
    cin >> cuvant;
    for(int i = strlen(cuvant); i >= 0; i--) {
        strncpy(auxiliar, cuvant, i);
        auxiliar[i]='\0';
        cout << auxiliar;
        if (i != 0) {
            cout << endl;
        }
    }

    for(int i=0; i < strlen(cuvant); i++) {
        strcpy(auxiliar, cuvant + i);
        cout << auxiliar << endl;
    }
}

```

```
    return 0;
}
```

9.   ◦ Link: <https://www.pbinfo.ro/probleme/84/interschimbarelitere>  
      ◦ Solutie:

```
#include <iostream>
#include <cstring>

int esteConsoana(char ch);
int esteVocala(char ch);

using namespace std;
int main() {
    char cuvant[10];
    cin >> cuvant;
    int pozitieVocala = -1;
    int pozitieConsoana = -1;

    for(int i = 0; i < strlen(cuvant); i++) {
        if(pozitieVocala == -1 && esteVocala(cuvant[i])) {
            pozitieVocala = i;
        } else if (esteConsoana(cuvant[i])) {
            pozitieConsoana = i;
        }
    }

    if (pozitieVocala != -1 && pozitieConsoana != -1) {
        char temp = cuvant[pozitieVocala];
        cuvant[pozitieVocala] = cuvant[pozitieConsoana];
        cuvant[pozitieConsoana] = temp;
        cout << cuvant;
    } else {
        cout << "IMPOSIBIL";
    }

    return 0;
}

int esteVocala(char ch) {
    return strchr("aeiouAEIOU", ch) != NULL;
}

int esteConsoana(char ch) {
    if (isalpha(ch) && !esteVocala(ch)) {
        return 1;
    } else {
        return 0;
    }
}
```



10.   ◦ Link: <https://www.pbinfo.ro/probleme/85/inserareasterisc>  
      ◦ Solutie:

```
#include <iostream>
#include <string.h>

int esteVocala(char ch);

using namespace std;
int main() {
    char cuvant[101];
    char rezultat[201]="";
    int indexRezultat = 0, areVocale = 0;

    cin.getline(cuvant, 101);

    for(int i = 0; i < strlen(cuvant); i++) {
        rezultat[indexRezultat] = cuvant[i];
        if(esteVocala(cuvant[i])) {
            indexRezultat++;
            rezultat[indexRezultat] = '*';
            areVocale =1;
        }
        indexRezultat++;
    }
    if(areVocale)
    {
        cout << rezultat;
    }
    else
    {
        cout<<"FARA VOCALE";
    }

    return 0;
}

int esteVocala(char ch) {
    return strchr("aeiou", ch) != NULL;
}
```

## Exercitii prelucrari siruri din variante de BAC

1.   ◦ Sursa: BAC 2010 August - Subiectul II, exercitiul 2.  
      ◦ Enunt:

Se consideră secvența de instrucțiuni alăturată, în care variabilele *i* și *j* sunt de tip întreg, iar variabila *s* poate

memora un șir de cel mult 20 de caractere. Șirul afișat în urma executării secvenței este:

```
strcpy(s,"bacalaureat");
j=strlen(s);
for(i=0;i<3;i++)
    if(s[i]!=s[j-i-1])
        s[i]=s[j-i-1];
cout<<s; | printf("%s",s);
```

o Variante:

- a. aureatbacal
- b. bacalaureab
- c. taealaureat [CORECT]
- d. taerualacab

2. o Sursa: BAC 2010 Iunie, Subiectul II, exercitiul 5

o Enunt:

- Se consideră un text cu cel mult 100 de caractere, format din cuvinte și numere, separate prin unul sau mai multe spații. Cuvintele sunt formate numai din litere ale alfabetului englez. Toate numerele sunt reale, pozitive și sunt formate numai din parte întreagă sau din parte întreagă și parte fracționară, separate prin punctul zecimal (.). Scrieți un program C/C++ care citește de la tastatură textul, pe care îl transformă, eliminând din componenta sa toate numerele care au parte fracționară. Programul afișează apoi pe ecran textul obținut.

Exemplu: dacă textul citit este  
 Reprezentat cu 4 zecimale PI este 3.1415 si e este  
 2.7182  
 se va afișa pe ecran textul:  
 Reprezentat cu 4 zecimale PI este si e este

o Solutie:

```
using namespace std;
int main() {
    char text[101], copieText[101];
    cin.getline(text, 101);
    int indexCopie = 0;
    for (int i = 0; i < strlen(text); i++) {
        // Aici ne uitam daca este sau nu cifra
        if (isdigit(text[i])) {
            int j = i+1;
            int arePunct = 0;
```

```

// Aici verificam daca este cifra sau are numar
si
    // mergem cat timp este adevarata conditia
    while(isdigit(text[j]) || text[j] == '.') {
        if (text[j] == '.') {
            arePunct = 1;
        }
        j++;
    }
    // daca am gasit un numar ce are punct atunci
    // efectiv sarim peste tot numarul ce a avut
punct
    //de aceea zicem i = j+1, il mutam pe i, dincolo
de numarul ce avea .
    if (arePunct == 1) {
        i = j+1;
    } else {
        copieText[indexCopie++] = text[i];
    }
} else {
    copieText[indexCopie++] = text[i];
}
}

cout << copieText;
return 0;
}

if (isdigit(text[i])) {
    int j = i+1;
    int arePunct = 0;
    while(isdigit(text[j]) || text[j] == '.') {
        if (text[j] == '.') {
            arePunct = 1;
        }
        j++;
    }
    if (arePunct == 1) {
        i = j+1;
    } else {
        copieText[indexCopie++] = text[i];
    }
} else {
    copieText[indexCopie++] = text[i];
}
}

cout << copieText;
return 0;
}

```

3.
  - o Sursa: BAC 2011 Iunie, Subiectul II, exercitiul 5
  - o Enunt:

- Scrieti un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 20$ ) și apoi  $n$  cuvinte distincte, fiecare fiind format din cel mult 20 de caractere, numai litere mici ale alfabetului englez. La introducerea datelor, după fiecare cuvânt se tastează Enter. Programul afișează pe ecran numărul de cuvinte dintre ultimele  $n-1$  citite, care încep cu primul cuvânt citit.

Exemplu: dacă  $n=5$  și cuvintele citite sunt:  
bun  
buncar  
bunici  
abundent  
bunavoie  
pe ecran se afișează 3 (deoarece numai cuvintele buncar, bunici și bunavoie încep cu bun).

- Solutie:

```
#include <iostream>
#include <string.h>

using namespace std;
int main() {
    int n, contor = 0;
    cin >> n;
    char primulCuvant[21];
    cin >> primulCuvant;
    for (int i = 1; i < n; i++) {
        char cuvant[21];
        cin >> cuvant;
        int continePrimulCuvant = 1;
        for (int j = 0; j < strlen(primulCuvant); j++) {
            if (primulCuvant[j] != cuvant[j]) {
                continePrimulCuvant = 0;
                break;
            }
        }
        if (continePrimulCuvant == 1) {
            contor++;
        }
    }

    cout << contor;
    return 0;
}
```

4. ○ Sursa: BAC 2011 August, Subiectul II, Exercițiul 4

o Enunt:

- În secvența de program alăturată variabila `a` memorează un șir cu cel mult 100 de caractere, iar variabilele `i` și `k` sunt de tip întreg. Scrieți șirul afișat pe ecran în urma executării secvenței.

```
k='a'-'A';
strcpy(a,"bacalaureat");
for(i=0;i<strlen(a);i++)
    if(strchr("aeiou",a[i])!=NULL)
        a[i]=a[i]-k;
cout<<a; | printf("%s",a)
```

o Solutie:

```
programul transforma vocalele in litere mari adica
bAcAlAUrEAt
```

5. o Sursa: BAC 2012 Iunie, Subiectul II, Exercițiul 5

o Enunt:

- Se consideră două șiruri de caractere `a`, de lungime `na` și `b`, de lungime `nb`. Șirul `a` este numit prefix al șirului `b` dacă  $na \leq nb$  și subșirul lui `b` determinat de primele sale `na` caractere coincide cu `a`. Șirul `a` este numit sufix al șirului `b` dacă  $na \leq nb$  și subșirul lui `b` determinat de ultimele sale `na` caractere coincide cu `a`. Scrieți un program C/C++ care citește de la tastatură două cuvinte, fiecare fiind format din cel mult 30 de caractere, numai litere mici ale alfabetului englez. După primul cuvânt se tastează Enter. Programul afișează pe ecran, separate prin câte un spațiu, toate sufixele primului cuvânt care sunt prefixe ale celui de al doilea cuvânt, ca în exemplu. Dacă nu există astfel de subșiruri, se afișează pe ecran mesajul **NU EXISTA**.

```
Exemple:
dacă se citesc, în ordine, cuvintele
"
rebele
elegant
"
se afișează (nu neapărat în această ordine) subșirurile:
"e ele"
iar dacă se citesc, în ordine, cuvintele
"
strai
rai
"
se afișează subșirul "rai"
```

Exemple: dacă se citesc, în ordine, cuvintele rebele elegant se afișează (nu neapărat în această ordine) subșirurile: e ele iar dacă se citesc, în ordine, cuvintele strai rai se afișează subșirul rai

o Solutie:

```
#include <iostream>
#include <string.h>

using namespace std;
int main() {
    char a[31], b[31];
    int areSufixePrefixe = 0;
    cin >> a;
    cin >> b;
    int lungimeSirA = strlen(a);
    int lungimeSufix = 1;
    while (lungimeSufix <= lungimeSirA) {
        int indexB = 0;
        int amGasitPrefixNou = 1;
        int i = lungimeSirA - lungimeSufix;
        while(i < lungimeSirA) {
            if (a[i] != b[indexB++]) {
                amGasitPrefixNou = 0;
                break;
            }
            i++;
        }
        if (amGasitPrefixNou) {
            areSufixePrefixe = 1;
            for (int j = lungimeSirA - lungimeSufix; j <
lungimeSirA; j++) {
                cout << a[j];
            }
            cout << " ";
        }

        lungimeSufix++;
    }

    if (areSufixePrefixe == 0) {
        cout << "NU EXISTA";
    }

    return 0;
}
```

### Extra exercitii de antrenament:

- o Sursa: Bac 2012 August, Subiectul 2, exercitiul 4  
o Enunt:

- Se consideră două șiruri de caractere a, de lungime na și b, de lungime nb. Șirul a este numit sufix al șirului b dacă  $na \leq nb$  și subșirul lui b determinat de ultimele sale na caractere coincide cu a. În secvența de instrucțiuni de mai jos variabila i este de tip întreg, iar variabila s memorează un șir cu cel mult 20 de caractere. Fără a utiliza alte variabile, scrieți una sau mai multe instrucțiuni care pot înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran, în ordinea descrescătoare a lungimii, separate prin câte un spațiu, toate sufixele șirului memorat în variabila s, ca în exemplu

- ```
Exemplu: pentru șirul elevi se afișează: elevi levi evi
vi i
for(i=0;i<strlen(s);i++)
.....
```

○ Soluție:

2. ○ Sursa: Bac 2012 Iunie, Subiectul II, Exercițiul 1

○ Enunț:

- Expresia `strlen("bine")` are valoarea:

○ Variante:

- a. 1
- b. 4
- c. 5
- d. 6

3. ○ Sursa: Bac 2013 August, Subiectul II, Exercițiul 5.

○ Enunț:

- Se consideră un text cu cel mult 100 de caractere (litere mici ale alfabetului englez și spații), în care cuvintele sunt separate prin unul sau mai multe spații. Înaintea primului cuvânt și după ultimul cuvânt nu există spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul menționat mai sus și determină transformarea acestuia în memorie prin eliminarea unor spații, astfel încât între oricare două cuvinte alăturate să rămână exact un spațiu. Programul afișează pe ecran textul obținut. Exemplu: pentru textul

```
"in vacanta    plec la          mare"
se obține și se afișează
"in vacanta plec la mare"
```

○ Soluție:

- 4.
- Sursa: Bac 2014 iunie, Subiectul II, exercitiul 2
  - Enunt:
    - Variabila `s` poate memora un sir cu maximum 20 de caractere. In urma executarii secventie de instructiuni alaturate se afiseaza:

```
strcpy(s, "1b2d3")
s[2] = 'a' + 2;
strcpy(s, s+1);
strcpy(s+3, s+4);
cout << s; | printf("%s", s);
```

- Variante:
    - a. 1b438
    - b. 1bcd8
    - c. ba2
    - d. bcd
- 5.
- Sursa: Bac 2014 august, Subiectul II, Exercițiul 4
  - Enunt:

Variabila `s` poate memora un șir cu maximum 20 de caractere, iar variabila `i` este de tip întreg. Scrieți ce se afișează în urma executării secvenței de instrucțiuni de mai jos.

```
strcpy(s,"BACALAUREAT");
cout<<strlen(s); | printf("%d",strlen(s));
i=0;
while (i<strlen(s)-1)
{ if(strchr("EAIUO",s[i])!=NULL) strcpy(s+i+1,s+i+2);
i++;
}
cout<<s; | printf("%s",s);
```