

Bacalaureat informatică

FIȘA 4.2. Tipul șir de caractere

Pentru fiecare dintre următorii itemi, alegeți litera corespunzătoare răspunsului corect.

1. În secvența de instrucțiuni alăturată, variabila `s` memorează un șir de caractere, iar variabila `i` este de tip `int`. Știind că în urma executării secvenței s-a afișat succesiunea de caractere `p*r*o*b*a***e*` stabiliți care este șirul de caractere memorat de variabila `s`.
- ```
for(i=0;i<strlen(s);i++)
 printf("%c*",s[i]);
 /cout<<s[i]<<'*';
```
- a. `proba*e`      b. `*p*r*o*b*a***e`      c. `probae`      d. `p*r*o*b*a*e*`
2. Pentru a inițializa variabila `n` cu lungimea efectivă a șirului de caractere memorat de variabila `w` scriem instrucțiunea:
- a. `n=length(w);`      b. `strlen(w)=n;`      c. `n=(int)w[0];`      d. `n=strlen(w);`
3. Variabila `s` a fost declarată astfel: `char s[20];` Ce se afișează după executarea secvenței de mai jos?
- ```
strcpy(s,"bacalaureat");
cout<<s[strlen(s)-4];
```
- a. `r` b. `e` c. `17` d. `8`
4. Se consideră declararea `char s[10];`, iar variabila `s` reține șirul de caractere `bac2007`. Ce valoare are expresia `strlen(s)`?
- a. `8` b. `10` c. `9` d. `7`
5. Variabilele `a` și `b` sunt declarate astfel: `char a[20],b[20];` Ce se afișează în urma executării secvenței de mai jos?
- ```
strcpy(a,"bac20"); strcpy(b,"07");
cout<<strcat(a,"")<<b; | printf("%s%s",strcat(a,""),b);
```
- a. `bac20*07`      b. `bac2007`      c. `bac20bac2007`      d. `bac140`
6. Algoritmul următor testează dacă șirul de caractere `s` este palindrom (citit atât de la stânga la dreapta, cât și de la dreapta la stânga, se obține același șir de caractere). Care este expresia care poate înlocui punctele de suspensie astfel încât algoritmul să fie corect ?
- ```
i=0;j=strlen(s)-1;
while(i<j && s[i]==s[j]) {i=i+1;j=j-1;}
if(.....)
    printf("palindrom");
else
    printf("nu este palindrom");
```
- ```
cout<<"palindrom";
cout<<"nu e palindrom";
```
- a. `i==j`      b. `i>=j`      c. `i<j`      d. `i!=j`
7. Secvența de program alăturată va afișa :
- ```
char c;
for(c='A';c<='Z';c++)
    cout<<c; / printf("%c",c);
```
- a. numerele naturale din intervalul `[1,27]`
b. numerele naturale din intervalul `[65,90]`
c. literele mari ale alfabetului englez
d. codurile ASCII ale literelor mari din alfabetul englez
8. Ce se va afișa după executarea următoarei secvențe de instrucțiuni:
- ```
char s[30]="examen de bacalaureat";
s[1]='X';
cout<<s; | printf("%s",s);
```
- a. `examen de bacalaureat`      b. `Examen de Bacalaureat`  
c. `eXamen de bacalaureat`      d. `Xxamen de bacalaureat`

9. Ce se va afișa după executarea secvenței de program alăturate?
- ```
char a[20]="informatica", b[20]="";
strncat(b,a,strlen(strchr(a,'t')));
cout<<b;          printf("%s",b);
c. rmatica        d. info
```
- a. tica b. form
10. Pentru declararea unei variabile care memorează simultan 20 de cuvinte având maximum 10 caractere fiecare, vom utiliza sintaxa:
- a. `char cuvinte[10][20];` b. `char cuvinte[10][];`
c. `char cuvinte[20,10];` d. `char cuvinte[20][11];`
11. Știind că variabila `s` va fi folosită pentru a memora simultan numele celor 10 profesori ai unei clase și că fiecare dintre aceste nume are cel mult 20 de litere, care este varianta corectă de declarare a lui `s`?
- a. `char s[20][11];` b. `char *s[10][21];` c. `char s[10][21];` d. `string s[10];`
12. Două șiruri de caractere `s` și `t` au lungimile egale dacă și numai dacă:
- a. `strlen(s)==strlen(t)` b. `s==t`
c. `s[0]==t[0]` d. `strcmp(s,t)==0`
13. Dacă `s` este o variabilă de tip șir de caractere (declarată astfel: `char s[100];`), atunci care dintre următoarele expresii reprezintă ultimul caracter memorat în șir?
- a. `s[strlen(s)]` b. `s[strlen(s)-1]`
c. `s[100]` d. `s[strlen[s]]`
14. Se consideră secvența alăturată:
- ```
if (strcmp(a,b)>0)
printf("%s ",a); / cout<<a;
else
if (strcmp(a,b)==0)
printf("egalitate"); / cout<<"egalitate";
else
printf("%s ",b); / cout<<b;
```
- Ce se va afișa dacă șirul `a` este `aabbdeeff` iar șirul `b` este `aabbdeeff`?
- a. `aabbdeeff`                      b. `aabbdeeff`  
c. nici una dintre variantele propuse    d. `egalitate`

Pentru fiecare dintre următoarele enunțuri scrieți programul C++ corespunzător.

1. Scrieți programul C sau C++ care citește de la tastatură un șir de cel mult 30 de caractere și creează fișierul text **BAC.TXT** ce conține șirul de caractere dat și toate prefixele acestuia de lungime cel puțin 1, fiecare pe câte o linie, în ordinea descrescătoare a lungimii prefixelor. De exemplu, dacă se citește șirul: `proba`, atunci **BAC.TXT** va conține:
- ```
proba
prob
pro
pr
p
```
2. Scrieți programul C sau C++ care citește de la tastatură un șir de cel mult 30 de caractere și creează fișierul text **BAC.TXT** ce conține șirul de caractere dat și toate sufixele acestuia de lungime cel puțin 1, fiecare pe câte o linie, în ordinea crescătoare a lungimii sufixelor, aliniate la stânga. De exemplu, dacă se citește șirul `teste`, atunci **BAC.TXT** va conține:
- ```
e
te
ste
este
teste
```

3. Scrieți programul C sau C++ care citește de la tastatură un șir **s** de cel mult **30** de caractere și un caracter **c**; programul determină dublarea fiecărei apariții a caracterului **c** în **s** și scrie noul șir obținut în fișierul text **BAC.TXT**.

De exemplu, dacă se citește șirul: **alfabetar** și caracterul **a** atunci fișierul **BAC.TXT** va conține șirul: **aalfaabaar**.

4. Scrieți un program care citește de la tastatură două șiruri de caractere, fiecare șir fiind format din cel mult **100** de litere mici din alfabetul englez, și care afișează mesajul **"DA"** dacă toate literele din primul șir se găsesc în cel de-al doilea șir, nu neapărat în aceeași ordine și de același număr de ori, sau mesajul **"NU"** în caz contrar. De exemplu, dacă primul șir este **"baraca"**, iar cel de-al doilea șir este **"abracadabra"**, programul trebuie să afișeze mesajul **"DA"** deoarece literele primului șir apar în cel de-al doilea șir.
5. Scrieți un program care citește de la tastatură un șir de caractere format din cel mult **20** de litere mici din alfabetul englez și care afișează mesajul **"DA"** dacă șirul este format dintr-un număr egal de consoane și vocale, respectiv mesajul **"NU"** în caz contrar.

6. Scrieți un subprogram **vocal** care are ca parametru un șir de caractere **s** format din cel mult **50** de litere ale alfabetului englez și returnează numărul de vocale din șirul respectiv.

7. Fișierul text **BAC.TXT** conține prenumele elevilor unei clase, câte un prenume (șir de maximum **20** de caractere) pe o linie a fișierului. Scrieți un program eficient care afișează pe ecran numărul de caractere ale celui mai lung prenume precum și numărul prenumelor cu această lungime.

De exemplu pentru fișierul alăturat valorile afișate sunt **8 2** (**Gheorghe** și **Vasilica** au **8** caractere).

```
BAC.TXT
Elena
Gheorghe
Andreea
Maria
Vasilica
```

8. Subprogramul **cifre** are un singur parametru prin care primește un șir cu maximum **255** de caractere și tipărește caracterele numerice din șir, în ordinea în care se găsesc în șir. Scrieți definiția completă a subprogramului **cifre**. De exemplu, dacă șirul transmis ca parametru este **"ac56ghr12t9"**, se va afișa **56129**.

Scrieți definiția completă a subprogramului **cifre**.

9. Se citesc de la tastatură două șiruri de caractere formate din cel mult **50** de litere fiecare. Să se afișeze pe ecran șirul format prin preluarea alternativă, din fiecare șir, a câte unei litere (prima literă a primului șir, apoi prima literă a celui de-al doilea, apoi a doua literă a primului șir, apoi a doua literă a celui de-al doilea șir etc). Când se epuizează literele din unul dintre șiruri, se vor prelua toate literele rămase din celălalt șir.

Dacă se citesc șirurile **ABC** și **MNPRTXB** se va afișa șirul **AMBNCPRTXB**.

10. Scrieți programul **C/C++** care citește de la tastatură un șir cu maximum **255** de caractere format numai din litere mici și mari ale alfabetului englez și îl scrie în fișierul **bac.out** "codificat" conform următoarei reguli: dacă numărul de vocale (a,e,i,o,u,A,E,I,O,U) din șir este mai mare sau egal cu numărul de consoane din șir, fiecare vocală se va înlocui cu caracterul următor din alfabetul englez, altfel fiecare consoană se va înlocui cu caracterul precedent din alfabetul englez. De exemplu, pentru șirul **bacalaureat**, fișierul va conține șirul **bcb1bvrfbt** deoarece șirul inițial are **6** vocale și **5** consoane; pentru șirul **basca**, fișierul va conține șirul **aarba** deoarece șirul inițial are **2** vocale și **3** consoane.

11. Scrieți programul **C/C++** care citește de la tastatură un text **t** format din cel mult **80** de caractere, litere și spații, și apoi elimină spațiile multiple din textul **t**. Dacă între două cuvinte există două sau mai multe caractere spațiu, trebuie eliminate unele dintre ele astfel încât să rămână exact unul. Se știe că la începutul și la sfârșitul șirului nu sunt spații. Textul obținut după eliminare se va afișa pe ecran.

De exemplu, dacă se citește textul: **Maria**    **a**  **adus**   **cana**, atunci textul afișat va fi: **Maria**  **a**  **adus**  **cana**.

12. Se citește de la tastatură un text format din cuvinte separate între ele prin câte un spațiu. Fiecare cuvânt are cel mult 40 de caractere, doar litere mici ale alfabetului englez. Textul are cel mult 200 de caractere. Scrieți programul C sau C++ care afișează pe ecran, pe linii separate, doar cuvintele din textul citit care conțin cel mult trei vocale. Se consideră vocale: **a, e, i, o, u**.

Exemplu. Dacă textul este:

**pentru examenul de bacalareat se folosesc tablouri**

se afișează pe ecran:

**pen  
tr  
u  
fo  
lo  
sesc**

13. Scrieți un program C/C++ care citește de la tastatură 4 șiruri de caractere formate din maximum 14 litere mici fiecare. Să se creeze fișierul **BAC.TXT** în care să se scrie toate perechile de șiruri dintre cele citite, perechi de forma **x,y** în care șirul **x** este subsecvență a șirului **y** sau șirul **y** este subsecvență a șirului **x**. Fiecare pereche determinată se va scrie în fișierul **BAC.TXT** pe câte un rând separate printr-o virgulă. Dacă nu există nici o astfel de pereche, în fișierul **BAC.TXT** se va scrie textul "**FĂRĂ SOLUȚIE**".

Exemplu:

Se citesc:     **ari**                     Se afișează în **BAC.TXT** :     **ari,mari**  
                  **calculator**                                     **calculator,lat**  
                  **mari**  
                  **lat**

14. Scrieți programul C/C++ care citește de la tastatură un cuvânt de cel mult 15 litere mici ale alfabetului englez și care afișează pe ecran, pe linii distincte, cuvintele obținute prin ștergerea succesivă a vocalelor din cuvântul citit, de la stânga la dreapta, ca în exemplu de mai jos:

Exemplu: Dacă se citește cuvântul **examen** se afișează:

**xamen**  
**xmen**  
**xmn**

15. Scrieți programul C/C++ care citește de la tastatură un cuvânt de cel mult 15 litere mici ale alfabetului englez și care scrie pe ecran, pe linii distincte, cuvintele obținute prin ștergerea succesivă a vocalelor în ordinea alfabetică a lor (**a, e, i, o, u**). La fiecare pas se vor șterge toate aparițiile din cuvânt ale unei vocale (ca în exemplu).

Exemplu: Dacă se citește cuvântul **bacalaureat** se afișează:

**bcluret**     (s-au șters toate cele patru apariții ale vocalei **a**)  
**bclurt**     (s-a șters unica apariție a vocalei **e**)  
**bclrt**     (s-a șters unica apariție a vocalei **u**)

16. Fișierele text **x.TXT** și **y.TXT** conțin fiecare numele a 7 persoane, câte un nume pe fiecare linie, fiecare nume având cel mult 15 litere. Știind că în fiecare fișier numele sunt memorate în ordine alfabetică, scrieți un program C/C++ care să citească din cele două fișiere și să afișeze pe ecran toate numele din cele două fișiere în ordine alfabetică, separate printr-un singur spațiu.

Exemplu:

Dacă fișierul **x.TXT** are conținutul :

**Ana**  
**Dana**  
**Daniel**  
**Ene**  
**Mara**  
**Nae**  
**Paul**

iar fișierul **y.TXT** are conținutul :

**Angi**  
**Cora**  
**Dora**  
**Horia**  
**Oana**  
**Paul**  
**Tibi**

Se va afișa pe ecran :

**Ana Angi Cora Dana Daniel Dora Ene Horia Mara Nae Oana Paul Paul Tibi**