

Aplicații șiruri de caractere

Teste antrenament 2020-2021

Grilă sau răspuns scurt		
1.	În secvența alăturată, variabila a memorează un șir cu cel mult 100 de caractere, iar variabilele i și k sunt de tip întreg. Scrieți ce se afișează pe ecran în urma executării secvenței. (6p.)	<pre> k='a'-'A'; strcpy(a,"ViCToriE"); cout<<strlen(a); printf("%d", strlen(a)); for(i=0;i<strlen(a);i++) if(a[i]>='A' && a[i]<='Z') a[i]=a[i]+k; else a[i]=a[i]-k; cout<<a; printf("%s",a); </pre>
2.	Variabila p este de tip întreg, iar variabila s memorează un șir de cel mult 20 de caractere, numai litere mari ale alfabetului englez. Fără a utiliza alte variabile, scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran toate literele șirului memorat de variabila s , cu excepția vocalei A , dacă în șirul inițial aceasta este alături de vocala I . Literele se afișează în ordinea apariției lor în șir. Exemplu: dacă șirul memorat în variabila s este ALIAN<u>T</u>A sau ALAI<u>N</u>T se va afișa ALINTA . (6p.)	
3.	Variabilele i și j sunt de tip întreg, iar variabila s poate memora un șir de cel mult 20 de caractere. Scrieți șirul memorat de variabila s în urma executării secvenței de mai jos. <pre> strcpy(s,"optsprezece"); i=0; j=strlen(s)-1; while(i<j) { if(strchr("aeiou",s[i])==NULL && strchr("aeiou",s[j])!=NULL) { s[i]=s[i]+1; s[j]=s[j]-1; } i=i+1;j=j-1; } </pre>	(6p.)
4.	Variabila s poate memora un șir de cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre> strcpy(s,"stilou"); cout<<s+4<<endl; printf("%s\n",s+4); s[0]=s[0]-1; s[1]=s[0]-3; s[2]=s[0]+1; s[3]=s[0]+3; s[4]='\0'; cout<<s; printf("%s",s); </pre>
5.	Variabila i este de tip întreg, iar variabila s poate memora un șir de cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre> strcpy(s,"stilou"+4); cout<<s<<endl; printf("%s\n",s); strncpy(s,"stilou",4); s[4]='\0'; for(i=0;i<4;i++) if(i%2==0) s[i]=s[0]+i-1; else s[i]=s[0]+3*(2*i/3-1); cout<<s; printf("%s",s); </pre>
6.	Variabila i este de tip întreg, iar variabila s poate memora un șir de cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre> strcpy(s,"stilou"); cout<<s+4<<endl; printf("%s\n",s+4); for(i=0;i<4;i++) s[i]=s[0]+(i-1)*(1-i%2)+3*(2*i/3-1)*(i%2); s[4]='\0'; cout<<s; printf("%s",s); </pre>
7.	Variabilele s1 și s2 pot memora câte un șir cu cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre> strcpy(s1,"bacalaureat2020"); cout<<strlen(s1); printf("%d",strlen(s1)); strcpy(s2,s1+11); strcpy(s1+3,s2); cout<<s1; printf("%s",s1); </pre>

Aplicații șiruri de caractere

Teste antrenament 2020-2021

8.	Variabilele <i>i</i> și <i>j</i> sunt de tip întreg, iar variabila <i>s</i> poate memora un șir de cel mult 20 de caractere. Scrieți șirul memorat de variabila <i>s</i> în urma executării secvenței de mai jos. <pre>strcpy(s,"informatie"); n=strlen(s)-1; for(i=0;i<n/2;i++) if(strchr("aeiou",s[i])!=NULL && strchr("aeiou",s[n-i])!=NULL) { s[i]=s[i+1]; s[n-i]=s[n-i-1]; }</pre> (6p.)	
9.	Variabila <i>i</i> este de tip întreg, iar variabila <i>s</i> permite memorarea unui șir cu cel mult 50 de caractere, numai litere mari ale alfabetului englez. Scrieți secvența de mai jos înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran toate consoanele din șir, iar în locul vocalelor din mulțimea {O,A,U} să se afișeze simbolul *. Exemplu: dacă șirul este CALCULATOR, se afișează C*LC*L*T**RE (6p.)	<pre>for(i=0;i<strlen(s);i++) if(.....) cout<<...; printf(.....); else cout<<...; printf(.....);</pre>
10.	Variabila <i>i</i> este de tip întreg, iar variabila <i>x</i> permite memorarea unui șir cu cel mult 100 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre>strcpy(x,"bac2021"); cout<<x+3<<endl; printf("%s\n",x+3); for(i=0;i<strlen(x);i++) if(strchr("0123456789",x[i])==0) cout<<x[i]<<'!'; printf("%c! ",x[i]);</pre>
11.	Scrieți ce se afișează în urma executării secvenței de mai jos, în care variabilele <i>s</i> și <i>t</i> permit memorarea câte unui șir de cel mult 50 de caractere. <pre>strcpy(s,"vorbeste"); s[3]=s[0]; s[5]=s[2]; s[0]=s[1]+1; s[2]=s[1]-2; s[6]=s[4]-1; strcpy(t,s); t[3]='\0'; cout<<t<<endl<<s+3; printf("%s\n%s",t,s+3);</pre> (6p.)	
12.	Variabila <i>i</i> este de tip întreg, iar variabilele <i>s</i> și <i>aux</i> permit memorarea câte unui șir cu cel mult 15 caractere. Scrieți ce se afișează pe ecran în urma executării secvenței de program alăturate. (6p.)	<pre>strcpy(s,"voalata"); cout<<strlen(s); printf("%d",strlen(s)); i=0; while (i<strlen(s)) if (strchr("aeiou",s[i])!=NULL) { strcpy(aux,s+i+1);strcpy(s+i,aux); i=i+1; } else i=i+2; cout<<s; printf("%s",s);</pre>
13.	Variabilele <i>i</i> și <i>j</i> sunt de tip întreg, iar variabilele <i>s</i> și <i>t</i> permit memorarea unui șir de cel mult 20 de caractere. Scrieți șirul accesat prin variabila <i>s</i> în urma executării secvenței de mai jos. <pre>strcpy(s,"ABCDUECDA"); i=0; j=strlen(s)-1; while (i<j) if(s[i]==s[j]) {strcpy(t,s+j+1);strcpy(s+j,t);strcpy(t,s+i+1);strcpy(s+i,t);j=j-2;} else { i=i+1; j=j-1; }</pre> (6p.)	
14.	Variabila <i>i</i> este de tip întreg, iar variabilele <i>s</i> și <i>t</i> permit memorarea câte un șir de cel mult 20 de caractere. Scrieți șirul accesat prin variabila <i>s</i> în urma executării secvenței alăturate. (6p.)	<pre>strcpy(s,"PRASLEA*CEL*VOINIC"); i=0; while (i<strlen(s)) if (strchr("ACEI",s[i])!=NULL) { strcpy(t,s+i+1); strcpy(s+i,t); } else i=i+1;</pre>

Aplicații șiruri de caractere

Teste antrenament 2020-2021

15.	Variabila i este de tip întreg, iar variabila s permite memorarea unui șir cu cel mult 10 ² caractere. Scrieți ce se afișează pe ecran în urma executării secvenței alăturate. (6p.)	<pre>strcpy(s, "informatica"); cout<<strlen(s); printf("%d", strlen(s)); for (i=0; i<strlen(s); i++) if (strchr("aeiou", s[i]) != NULL) s[i] = '*'; cout<<s; printf("%s", s);</pre>
16.	În secvența alăturată, variabila i este de tip întreg, iar variabilele s și t pot memora câte un șir cu cel mult 20 de caractere. Scrieți ce se afișează pe ecran în urma executării secvenței. (6p.)	<pre>strcpy(s, "sanataTEA"); cout<<strlen(s); printf("%d", strlen(s)); i=0; while(i<strlen(s)) if(s[i]=='a') { strcpy(t, s+i+1); strcpy(s+i, t); } else i=i+1; cout<<s; printf("%s", s);</pre>
17.	Variabila s permite memorarea unui șir de cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre>strcpy(s, "muzeu"); s[0]=s[0]+1; cout<<s[1]<<s[0]<<endl; printf("%c%c\n", s[1], s[0]); strcpy(s, "muzeu"+2); cout<<s; printf("%s", s);</pre>
18.	Variabila s memorează un șir de cel mult 20 de caractere (litere mari și mici ale alfabetului englez). Declarați eventuale alte variabile necesare și scrieți o secvență de instrucțiuni în urma executării căreia se afișează pe ecran vocalele care NU apar în șirul menționat. Se consideră vocale literele a, e, i, o, u, A, E, I, O, U . Exemplu: dacă se citește textul PREafrUMoASA se afișează pe ecran, nu neapărat în această ordine, vocalele: euOiI (6p.)	
19.	Variabilele s1 și s2 pot memora câte un șir de cel mult 50 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)	<pre>strcpy(s1, "bac2021"); cout<<strlen(s1)<<endl; printf("%d\n", length(s1)); strcpy(s2, s1+3); strcpy(s2+2, "20-"); strcat(s2, s1+3); cout<<s2; printf("%s", s2);</pre>
Parcurgere		
20.	Un text are cel mult 100 de caractere și este format din cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere ale alfabetului englez. Toate numerele sunt reale și sunt formate numai din parte întreagă sau din parte întreagă și parte fracționară, separate prin virgulă (,), numerele negative fiind precedate de semnul minus (-). Scrieți un program C/C++ care citește de la tastatură textul, pe care îl transformă, eliminând din componența sa toate numerele negative. Programul afișează apoi pe ecran textul obținut. Exemplu: pentru textul 2,7 minus 3,5 minus 2 egal 2,7 plus -3,5 plus -2 egal -0,2 rezultat se va afișa pe ecran textul: 2,7 minus 3,5 minus 2 egal 2,7 plus plus egal rezultat (10p.)	

Aplicații șiruri de caractere

Teste antrenament 2020-2021

21.	Într-un text cu cel mult 10^2 caractere, cuvintele sunt formate din litere mici și mari ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat, pe care îl transformă, astfel încât fiecare cuvânt să aibă prima literă mare, și toate celelalte litere mici. Textul obținut se afișează pe ecran. Exemplu: dacă de la tastatură se introduce textul <code>ABIA aSTept sa Merg lA scoala</code> se obține textul <code>Abia Astept Sa Merg La Scoala</code> (10p.)
22.	Într-un text cu cel mult 10^2 caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat, pe care îl transformă în memorie, eliminând numai ultima vocală care apare în text, ca în exemplu. Programul afișează pe ecran textul obținut sau mesajul <code>nu exista</code>, dacă în text nu există nicio vocală. Se consideră vocale literele <code>a, e, i, o, u</code>. Exemplu: dacă se citește textul: <code>cuvantul ritm poate fi tradus rhythm</code> se obține textul <code>cuvantul ritm poate fi trads rhythm</code> (10p.)
23.	Un text are cel mult 100 de caractere și este format din cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere ale alfabetului englez. Toate numerele sunt reale și sunt formate numai din parte întreagă sau din parte întreagă și parte fracționară, separate prin virgulă (<code>,</code>), numerele negative fiind precedate de semnul minus (<code>-</code>). Cel puțin unul dintre numerele reale este negativ. Scrieți un program C/C++ care citește de la tastatură textul, pe care îl transformă în memorie, înlocuind fiecare număr negativ cu valoarea sa absolută. Programul afișează apoi pe ecran textul obținut. Exemplu: pentru textul <code>Modul de -3,24 este 3,24 si modul de -15 este 15</code> se va afișa pe ecran textul: <code>Modul de 3,24 este 3,24 si modul de 15 este 15</code> (10p.)
24.	Un text cu cel mult 100 de caractere conține cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere mici ale alfabetului englez, iar numerele sunt reale, pozitive, cu partea întreagă și partea zecimală separate prin simbolul virgulă, sau numai cu partea întreagă, ca în exemplu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și îl transformă în memorie, înlocuind fiecare număr real cu partea întreagă a acestuia. Exemplu: pentru textul <code>partea intreaga a lui 5,75 este egala cu a lui 5,25 si cu a lui 5 si este 5</code> se afișează pe ecran <code>partea intreaga a lui 5 este egala cu a lui 5 si cu a lui 5 si este 5</code> (10p.)
Vector de cuvinte	
25.	Un cuvânt este prefix al unui alt cuvânt dacă se obține din acesta, prin eliminarea ultimelor sale litere. Scrieți un program C/C++ care citește de la tastatură un număr natural n ($n \in [2, 20]$) 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, separate prin câte un spațiu, cuvintele care îl au drept prefix pe ultimul cuvânt citit. Dacă nu există astfel de cuvinte, se afișează pe ecran mesajul <code>nu exista</code>. Exemplu: dacă $n=6$ și se citesc cuvintele alăturate, pe ecran se afișează <code>raita raid raion</code> (10p.) <u>raita</u> <u>grai</u> <u>raid</u> <u>raion</u> <u>straie</u> <u>rai</u>

Aplicații șiruri de caractere

Teste antrenament 2020-2021

26.	Un cuvânt este sufix al unui alt cuvânt dacă se obține din acesta, prin eliminarea primelor sale litere. Scrieți un program C/C++ care citește de la tastatură două numere naturale n și k ($n \in [2, 20]$, $k \in [1, n]$) 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, separate prin câte un spațiu, cuvintele care îl au drept sufix pe al k-lea cuvânt citit, ca în exemplu. Dacă nu există astfel de cuvinte, se afișează pe ecran mesajul nu exista. Exemplu: dacă $n=7$, $k=3$ și se citesc cuvintele alăturate, pe ecran se afișează paratirisi hiritisi	isihast paratirisi isi meremetisire acolisitor hiritisi paraponisit (10p.)
Extragere cuvinte		
27.	Un text cu cel mult 100 de caractere conține cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere mici ale alfabetului englez, iar numerele sunt reale, pozitive, cu partea zecimală și partea întreagă separate prin simbolul virgulă, sau numai cu partea întreagă, ca în exemplu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și afișează pe ecran numărul de valori întregi din text. Exemplu: pentru textul grus leucogeranus are 1,40 m inaltime si traieste intre 30 si 40 de ani se afișează pe ecran 2	(10p.)
28.	Într-un text cu cel mult 100 de caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul menționat și afișează pe ecran, pe linii separate, toate cuvintele sale pentru care numărul de vocale este strict mai mic decât numărul de consoane. Dacă nu există niciun astfel de cuvânt, se afișează pe ecran mesajul nu exista. Se consideră vocale literele din mulțimea a, e, i, o, u. Exemplu: pentru textul ei au plantat tamarix ea a adus iasomie se afișează pe ecran, nu neapărat în această ordine, cuvintele alăturate.	plantat tamarix (10p.)
29.	Într-un text cu cel mult 10^2 caractere cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul menționat, pe care îl modifică în memorie, inserând cuvântul succes între oricare două cuvinte ale sale care se termină cu aceeași literă. Cuvântul inserat este separat prin câte un spațiu de cuvintele vecine. Textul transformat este afișat pe ecran, iar dacă nu există perechi de astfel de cuvinte, se afișează pe ecran mesajul nu exista. Exemplu: dacă textul citit este <u>teste</u><u>z</u> <u>valide</u><u>z</u> <u>utilize</u><u>z</u> <u>date</u><u>z</u> <u>corecte</u><u>z</u> <u>acum</u> se obține textul <u>teste</u><u>z</u> <u>succes</u> <u>valide</u><u>z</u> <u>succes</u> <u>utilize</u><u>z</u> <u>date</u><u>z</u> <u>succes</u> <u>corecte</u><u>z</u> <u>acum</u>	(10p.)
30.	Într-un text cu cel mult 100 de caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul menționat și afișează pe ecran numărul de cuvinte ale sale formate dintr-un număr egal de vocale și consoane. Se consideră vocale literele din mulțimea a, e, i, o, u. Exemplu: pentru textul cuvantul <u>consoane</u> are <u>un</u> <u>numar</u> <u>de</u> <u>patru</u> <u>vocale</u> <u>si</u> <u>patru</u> <u>consoane</u> se afișează pe ecran 6.	(10p.)

Aplicații șiruri de caractere

Teste antrenament 2020-2021

31.	Într-un text cu cel mult 10^2 caractere cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul menționat, pe care îl modifică în memorie, duplicând fiecare cuvânt format numai din vocale. Cuvântul duplicat este separat prin câte un spațiu de cuvintele vecine. Textul transformat este afișat pe ecran, iar dacă nu există astfel de cuvinte, se afișează pe ecran mesajul nu exista. Exemplu: dacă textul citit este <code>oaia aia alba e a ei</code> se obține textul <code>oaia <u>oaia</u> aia <u>aia</u> alba e <u>e</u> a <u>a</u> ei <u>ei</u></code> (10p.)
32.	Într-un text cu cel mult 10^2 caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și afișează pe ecran, pe linii separate, toate cuvintele sale care conțin o singură vocală distinctă, ca în exemplu. Dacă nu există niciun astfel de cuvânt, se afișează pe ecran mesajul nu exista. Se consideră vocale literele din mulțimea <code>a, e, i, o, u</code>. Exemplu: pentru textul <code>a plantat cinci lalele visinii sau rosii</code> se afișează pe ecran, nu neapărat în această ordine, cuvintele alăturate. (10p.) a plantat cinci visinii