

1. Care dintre următoarele instrucțiuni sunt corecte sintactic? Variabilele care intervin sunt de tip *string* / *char* *.

a) `pos('a',s):=2;`
b) `c:=delete(s,2,2);`
c) `writeln(length('ana'));`
d) `insert('red',s,4);`

a) `p=strchr('a',b);`
b) `k = strchr(s,x)-s;`
c) `cout<<strlen("ana");`
d) `k = strcat("ma","ma");`

2. Se consideră variabila *s* de tip *string* / *char* *. Care dintre următoarele secvențe afișează valoarea variabilei *s* din care lipsesc primul și ultimul caracter? Variabila *i* aparține unui tip întreg.

a) `delete(s,1,1);`
`delete(s,length(s),1);`
`writeln(s);`
b) **for** `i:=2 to length(s)-1 do`
`write(s[i]);`
c) `n:=length(s);`
`delete(s,1,1);`
`delete(s,n,1);`
`writeln(s);`
d) `write(copy(s,2,length(s)));`

a) `strcpy(s,s+1);`
`strcpy(s+strlen(s)-1,`
`s+strlen(s));`
`cout<<s<<endl;`
b) **for**(`i=1; i<strlen(s)-1; i++`)
`cout<<s[i];`
c) `strcpy(s,strchr(s,s[strlen(s)-1])+1);`
`strcpy(s,s+1);`
`cout<<s;`
d) **for**(`i=0; i<strlen(s)-2; i++`)
`cout<<s[i];`

3. Ce se va afișa în urma executării următoarei secvențe de instrucțiuni?

`s:='Primavara';`
for `i:=1 to 3 do`
`delete(s,2,1);`
`writeln(s);`

`a = "Primavara";`
for (`int i=1;i<=3;i++`)
`strcpy(a+1,a+2);`
`cout<<a;`

a) Pavara;

b) ara;

c) rim;

d) Para.

4. Ce se va afișa în urma executării următoarei secvențe de instrucțiuni?

```
x:='Mama'; y:='Macara';
if x>y then write(x)
else if x<y then write(y)
      else writeln('Incorect');
```

```
x = "Mama"; y = "Macara";
if (strcmp(x,y)>0) cout<<x;
else
if (strcmp(x,y)==0)
  cout<<"Incorect";
else cout<<y;
```

- a) Macara b) Mama c) MamaIncorect d) Incorect

5. Care dintre următoarele expresii sunt corecte? Toate variabilele care intervin sunt de tip *string / char **.

```
a) a := b+'c';
b) a := pos(a,b);
c) a := a+b;
d) s := insert('red',s,2);
```

```
a) strcat(x,"abi");
b) x = strcmp(x,y);
c) strcat(strcpy(x,x+1),"abi");
d) x = strstr(x,y)-x;
```

6. Ce se va afișa la finalul executării următoarei secvențe de instrucțiuni? Toate variabilele care intervin sunt de tip *string / char **.

```
a := 'albacazapada';
x := upcase(a[1]) + a[1] +
      a[length(a)];
writeln(x);
```

```
x = "albacazapada";
x[0]=x[0]-32;
p=strchr(x,'a');
cout<<x[0]<<p[0]<<x[strlen(x)-1];
```

- a) aaa; b) AaA; c) Aaa; d) AAA.

7. Considerând că variabila *x* este de tip *string* în varianta Pascal, ce valoare va avea variabila șir de caractere *s* după execuția următoarelor instrucțiuni? Variabila *i* aparține unui tip întreg.

```
s:='MacarA';
x:='';
for i:=1 to 6 do
      x:=s[i]+x;
s:=x;
```

```
s = "MacarA";
for (i=0;i<strlen(s)/2;i++){
  char x= s[i];
  s[i]= s[strlen(s)-i-1];
  s[strlen(s)-i-1] = x;}

```

- a) MMMMMM; b) Macara; c) AAAAAA; d) AracaM.

8. Considerând că variabila *x* este de tip *string / char **, care dintre următoarele variante verifică, în mod corect, dacă primul caracter al său este o minusculă?

```
a)
  if (x[1]>='a')OR(x[1]<='z') then
    write('Corect');
b)
  if (x[1]>=a)AND(x[1]<=z) then
    write('Corect');
```

```
a)
  if (x[0]<'z')
    cout<<"Corect";
b)
  if (x[0]<'z' || x[0]>'a')
    cout<<"Corect";
```

```

c)
  if upcase(x[1])=x[1] then
    write('Corect');
d)
  if (ord(x[1])>=97)AND
    (ord(x[1])<=122) then
    write('Corect');

```

```

c)
  if (!(x[0]>122&& x[0]<97))
    cout<<"Corect";
d)
  if (!(x[0]<'a')&& (x[0]<=122))
    cout<<"Corect";

```

9. Ce se va afișa pe ecran în urma rulării următorului program?

```

var
  a:array[1..3] of string[6];
begin
  a[1]:='vector';
  a[2]:='de';
  a[3]:='siruri';
  write(a[2], ' ');
  write(a[2][1], ' ');
  write(a[3][4], ' ');
  write(a[1][2]);
end.

```

```

#include<iostream.h>
#include<string.h>
char a[3][10];
void main() {
  strcpy(a[0], "vector");
  strcpy(a[1], "de");
  strcpy(a[2], "siruri");
  cout<<a[1]<<' ';
  cout<<a[1][0]<<' ';
  cout<<a[2][3]<<' ';
  cout<<a[0][1]; }

```

a) de d r e b) de e r d c) e e r d d) de d u e

10. Ce se va afișa pe ecran în urma rulării următorului program?

```

var a,b,c:string;
    i:byte;
begin
  a:='mama'; b:='cana';
  c:='casa';
  i:=pos('a',a);
  while i<>0 do begin
    delete(a,i,1);
    delete(c,i,1);
    i:=pos('a',a);
  end;
  insert(a,b,2);
  writeln(a, ' ', b, ' ', c)
end.

```

```

#include<iostream.h>
#include<string.h>
char *a,*b,*c,k[256]; int i;
void main(){
  a="mama"; b="cana"; c="casa";
  int i=strchr(a,'a')-a;
  while (i>0) {
    strcpy(a+i,a+i+1);
    strcpy(c+i,c+i+1);
    i=strchr(a,'a')-a; }
  strncpy(k,b,1);
  strcat(k,a);
  strcat(k,b+1);
  cout<<a<<' '<<k<<' '<<c<<endl;
}

```

a) mcanam cana casa b) mm cmmana cs c) mm cn cs d) mm cana cs

11. Considerăm următoarele declarații:

```

a) var a:array [1..9,1..9]of char;
b) var a:char[9];
c) var a:array [0..9] of word;
d) var a:string[20];

```

```

a) char a[9][9];
b) char a;
c) unsigned int a[10];
d) char* a;

```

Care dintre acestea reprezintă declarația corectă a unei variabile care poate avea valoarea 'dimineata' în Pascal, respectiv "dimineata" pentru C/C++?

12. Considerăm următoarele declarații:

```
var x,y,z:string[200];          | char* z;  
                                | char x[200],y[200];
```

Identificați expresiile corecte sintactic din lista următoare:

a) x := '12' + 'zile';	a) z = z++;
b) z := y + x;	b) strcat(x,y);
c) y := y + str(13, x);	c) z = (x!=y && y!=z);
d) x := dec(z);	d) x = "12Azile";
e) x := copy('12Azile',4,2);	e) z = strchr("exercitii",' ');
f) x := insert('dimineata',y,3);	f) z = strcpy('A',"BBB");

13. Considerăm următoarea secvență de program, în care x și y sunt variabile din tipul șir de caractere. Ce se va afișa în urma execuției lor ?

x:='dimineata';	x="dimineata"; y="min";
y:='min' + x[length(x)];	strcat(y,x+strlen(x)-1);
writeln(pos(y,x));	p=strstr(y,x);
	cout<<(p!=NULL) ? (p-y) : 0;

a) 0 b) 3 c) ta d) dieta

14. Considerăm următoarele declarații:

```
var a:string[200]; i:byte;      | char a[200]; unsigned int i;
```

Se știe că șirul de caractere a conține numai caractere distincte, excepție făcând ultimele două care sunt identice. Identificați care dintre secvențele următoare de instrucțiuni este echivalentă cu funcția *length()* / *strlen()* ?

a) i:=1; while a[i]<>a[i+1] do inc(i); writeln(i+1);	a) i=0; while(a[i] != a[i+1]) i++; cout<<i+2;
b) i:=0; while a[i]<>a[i+1] do inc(i); writeln(i);	b) i=0; while(a[i] != a[i+1]) i++; cout<<i+1;
c) i:=1; if a[i] = a[i+1] then write(i)	c) i=0; if (a[i] == a[i+1]) cout<<i;
d) i:=1; if a[i]<>a[i+1] then inc(i) else write(i+1);	d) i=0; if (a[i] != a[i+1]) i++ ; else cout<<i ;

15. Considerăm următoarele declarații:

```
var a,b: string[100] ;        | char *a, *b;
```

Ce se va afișa în urma execuției următoarei secvențe de instrucțiuni?

```

a := 'mama'; b := 'Mamaie'
if a>b then write(a)
else
  if a=b then write('identice')
  else write(b);

```

```

a = "mama"; b="Mamaie";
if (strcmp(a,b)>0) cout<<a;
else
  if (strcmp(a,b)==0)
    cout<<"identice";
  else cout<<b;

```

- a) *Mamaie*;
 b) *mama*;
 c) *identice*;
 d) Secvența de instrucțiuni propusă nu execută nici o afișare.

16. Considerăm următoarele declarații:

```

var a:array[1..9] of string[20];
    n, i, j:byte; x:string[20];

```

```

char a[20][20]; unsigned int n, i, j;
char x[20];

```

și următoarea secvență de program:

```

readln(n);
for i:=1 to n do readln(a[i]);
for i:=1 to n-1 do
  for j:=i+1 to n do
    if a[j]<a[i] then begin
      x:=a[j]; a[j]:=a[i]; a[i]:=x;
    end;

```

```

cin>>n;
for(i=0;i<n;i++) cin>>a[i];
for (i=0;i<n-1;i++)
  for (j=i+1;j<n;j++)
    if (strcmp(a[j],a[i])<0) {
      strcpy(x,a[j]); strcpy(a[j],a[i]);
      strcpy(a[i],x); }

```

Ce prelucrare realizează această secvență asupra elementelor vectorului *a*?

- a) Ordonează crescător elementele tabloului *a* după lungimea șirurilor de caractere;
 b) Ordonează lexicografic crescător elementele tabloului *a*;
 c) Înlocuiește elementele tabloului cu șirul de caractere maxim din punct de vedere lexicografic;
 d) Înlocuiește elementele tabloului cu șirul de caractere de lungime maximă.

17. Considerăm programul următor:

```

var a,b: string;
    i,x,y:integer;
begin
  readln(a);readln(b);
  i:=1;
  x:=length(a); y:=length(b);
  while (y-x+1>=i)
    and (copy(b,i,x)<>a) do inc(i);
  if i > y-x+1 then i:=0;
  writeln(i);
end.

```

```

#include <iostream.h>
#include <string.h>
char a[256],b[256]; int i,x,y;
void main() {
  cin>>a; cin>>b;
  x=strlen(a); y=strlen(b);
  for(i=0; y-x>=i &&
    strncmp(b+i,a,x)!=0; i++);
  if (i>y-x) i=-1;
  cout<<i<<endl;
}

```

Identificați care dintre expresiile următoare sunt echivalente cu de programul de mai sus (la evaluarea expresiei se obține aceeași valoare cu cea afișată prin program).

- | | |
|-------------------|-------------------|
| a) concat(a,b) | a) strlen(a); |
| b) length(a + b); | b) strlen(b); |
| c) pos(a,b); | c) strstr(b,a)-b; |
| d) copy(a,b,i); | d) strcmp(a,b); |

18. Fie secvența de instrucțiuni următoare :

For i :=1 to length(a) do	for (i=0;i<strlen(a);i++)
if a[i] in ['A'..'Z'] then	if (a[i]>='A' && a[i]<='Z')
a[i] :=chr(ord(a[i])+32) ;	a[i] += 32;

Știind că *a* este un șir de caractere și *i* o variabilă de tip întreg, identificați prelucrarea realizată asupra caracterelor sale.

- transformarea caracterelor de tip minusculă în majuscula corespunzătoare;
- inserarea șirului de caractere 32 după fiecare caracter de tip majusculă.
- transformarea caracterelor de tip majusculă în minuscula corespunzătoare;
- ordonarea alfabetică a majusculilor în cadrul șirului;
- ștergerea caracterelor spațiu.

19. Fie următorul program:

var s:string[10];	#include<iostream.h>
i:integer; x:char;	#include<string.h>
begin	char s[10]; int i; char x;
s:='clasa';	void main()
for i:=1 to length(s)-1 do begin	strcpy(s,"clasa");
if (s[i]>s[i+1]) then begin	for (i=0;i<strlen(s)-1;i++){
x:=s[i];	if (s[i]>s[i+1]){
s[i]:=s[i+1];	x=s[i]; s[i]=s[i+1];
s[i+1]:=x;	s[i+1]=x;
end;	}
write(s,' ');	cout<<s<<" ";
end;	}
end.	}

Ce se va afișa pe ecran în urma execuției acestui program?

- clasa clasa calas calas
- clasa calsa calas
- clasa calsa calsa calas
- clasa aclsa aclas acals

20. Se consideră șirurile de caractere a și b . Identificați care dintre următoarele secvențe de instrucțiuni modifică valoarea șirului b prin ștergerea primei apariții a lui a în b . În situația în care a nu se regăsește în b , valoarea acestuia din urmă trebuie să rămână neschimbată. Variabila x este de tip *integer* pentru Pascal, respectiv de tip *int* pentru C/C++.

a)
`delete(b, pos(a, b), length(a));`

b)
`x:=pos(a, b);`
`b:=copy(b, 1, x-1)+`
`copy(b, x+length(a), x)`

c)
`delete(a, pos(a, b), length(b));`

d)
`x:=pos(a, b);`
`if x<>0 then`
`b:=copy(b, 1, x-1)+`
`copy(b, x+length(a), x);`

a)
`x=strstr(b, a)-b;`
`if (x>=0)`
`strcpy(b+x, b+x+strlen(a));`

b)
`x=strstr(b, a)-b;`
`strcpy(b+x, b+x+strlen(a)-1);`

c)
`x=strstr(a, b)-a;`
`if (x>=0)`
`strcpy(a+x, a+x+strlen(b));`

d)
`x=strstr(a, b)-a;`
`strcpy(a+x, a+x+strlen(b));`

21. Considerăm următorul program. Ce condiție trebuie îndeplinită pentru ca în urma rulării acestuia să se afișeze mesajul *Corect* ?

```
var a, b: string;
begin
  readln(a);
  readln(b);
  while (a[1]=b[1]) and (a<>'') and
    (b<>'') do
    begin
      delete(a, 1, 1);
      delete(b, 1, 1);
    end;
  if (a='') and (b='') then
    write('Corect')
  else
    write('Incorect');
end.
```

```
#include<iostream.h>
#include<string.h>
char a[256], b[256];
void main()
{
  cin>>a;
  cin>>b;
  while (a[0]==b[0] && a[0]!=0
    && b[0]!=0) {
    strcpy(a, a+1);
    strcpy(b, b+1);
  }
  if (a[0]==0 && b[0]==0)
    cout<<"Corect";
  else cout<<"Incorect";
}
```

- a) afișează mesajul *Corect* doar dacă șirurile a și b au lungimi egale;
- b) afișează mesajul *Corect* doar dacă șirurile a și b au valori identice;
- c) afișează mesajul *Corect* doar dacă șirurile a și b au lungimi nule;
- d) afișează mesajul *Corect* doar dacă șirurile a și b au valori identice de lungime 1;

22. Se consideră următorul program:

```
var a,b:string;
    i:integer;
begin
    b:='';
    for i:=1 to 3 do begin
        readln(a);
        b:=b+copy(a,i,length(a)-i+1);
    end;
    writeln(b)
end.
```

```
#include<iostream.h>
#include<string.h>
char a[256],b[256]; int i;
void main()
{ strcpy(b,"");
  for(i=0; i<3; i++){
    cin>>a; strcat(b,a+i);
  }
  cout<<b;
}
```

Identificați ce se va afișa dacă de la tastatură se vor introduce, în ordine, șirurile de caractere: "copil", "masina", "bloc".

- a) opilsinac b) cmb c) copilmasinbl d) copilasinaoc

23. Considerăm următorul program:

```
var a,b:string;
begin
    readln(a);
    readln(b);
    while (pos(a[1],b)<>0) do
    begin
        delete(b,pos(a[1],b),1);
        delete(a,1,1);
    end;
    if (a='') and (b='') then
        write('Da')
    else
        write('nu');
end.
```

```
#include<iostream.h>
#include<string.h>
char a[256],b[256],*p; int i;
void main()
{ cin>>a; cin>>b;
  while (strchr(b,a[0])!=NULL &&
        a[0]!=0) {
    p=strchr(b,a[0]);
    strcpy(p,p+1);strcpy(a,a+1);
  }
  if (a[0]==0 && b[0]==0)
    cout<< "da";
  else cout<<"nu";
}
```

La sfârșitul rulării acestuia se va afișa mesajul *Da* dacă și numai dacă:

- a) șirurile *a* și *b* au valori egale;
 b) șirurile *a* și *b* au lungimi identice;
 c) șirurile *a* și *b* sunt formate din exact aceleași caractere, eventual în altă ordine.
 d) fiecare caracter al șirului *a* apare și în șirul *b*;

24. Fie *a* un șir de caractere (*string* / *char **) și *x* o variabilă întreagă. Care dintre următoarele instrucțiuni elimină toată secvența de caractere identice de la începutul șirului *a* (pentru a fi ștearsă, secvența va conține minimum 2 caractere):

```
a) x:=length(a);
   while((a[1])=a[2]) and
   (length(a)>1) do delete(a,1,1);
   if length(a)<>x then
       delete(a,1,1);
```

```
a) x=strlen(a);
   while (*a==*(a+1)) {a=a++;}
   if (strlen(a)!=x)
       strcpy(a,a+1);
```


b) `while((a[1])=a[2])and
(length(a)>1) do delete(a,1,1);`

c) `while((a[1]) <> a[2])and
(length(a)>1) do delete(a,1,1);`

d) `while((a[1]) <> a[2])and
(length(a)>1) do delete(a,1,1);
delete(a,1,1);`

b) `while (*a==*a+1) {
a=a++;
}
strcpy(a,a+1);`

c) `while (*a!=*(a+1)) {a=a++;
strcpy(a,a+1);`

d) `while (*a==*(a+1)) {a=a++;
strcpy(a,a+1);`

25. Fie a un șir de caractere (*string / char **) și i, j variabile întregi. Care dintre următoarele instrucțiuni permit afișarea mesajului "Da" dacă și numai dacă valoarea lui a este palindrom?

a) `i:=1; j:=length(a);
while (a[i]<>a[j])and(i<j) do
begin dec(i); inc(j); end;
if (i>j) then write('DA')
else write('NU');`

b) `i:=1; j:=length(a);
while (a[i]<>a[j])and(i<j) do
begin inc(i); dec(j); end;
if (i>=j) then write('DA')
else write('NU');`

c) `i:=1; j:=length(a);
while (a[i]=a[j])and(i<j) do
begin inc(i); dec(j); end;
if (i>=j) then write('DA')
else write('NU');`

d) `i:=1; j:=length(a);
while (a[i]=a[j]) and (i<j) do
begin dec(i); inc(j); end;
if (i>=j) then write('DA')
else write('NU');`

a) `i:=0; j:=strlen(a)-1;
while ((a[i]==a[j])&&(i<j))
{i--; j++;}
if(i>=j) cout<<"DA";
else cout<<"NU";`

b) `i:=0; j:=strlen(a)-1;
while ((a[i]!=a[j])&&(i<j))
{ i++; j--; }
if(i<=j) cout<<"DA";
else cout<<"NU";`

c) `i:=0; j:=strlen(a)-1;
while ((a[i]==a[j])&&(i<j))
{ i++; j--; }
if (i>=j) cout<<"DA";
else cout<<"NU";`

d) `i:=0; j:=strlen(a)-1;
while ((a[i]!=a[j])&&(i<j))
{ i--; j++; }
if (i>=j) cout<<"DA";
else cout<<"NU";`

26. Știind că variabila a este folosită pentru a memora, ca șir de caractere, numele unei discipline studiate în liceu (maximum 50 caractere), identificați o declarație corectă a sa:

a) `var a=string;`

b) `var a:string[39];`

c) `var a:string[50];`

d) `var a:array[1..50] of string;`

a) `char a;`

b) `char a[39];`

c) `char a[50];`

d) `char* a[20];`

27. Știind că variabila a este utilizată pentru a memora numele celor 7 zile ale săptămânii, cum trebuie ea declarată:

a) `var a:string[7][7];`
 b) `var a: array[1..7] of string;`
 c) `var a:string[7];`
 d) `var a: array[1..7] of char;`

a) `char a[7];`
 b) `char a[7][17];`
 c) `char a;`
 d) `char** a[7];`

28. Considerăm secvența de instrucțiuni următoare în care variabila *s* este un șir de caractere, *i* și *k* variabile întregi, *x* o variabilă din tipul *char* iar *ok* este o variabilă din tipul *boolean*(Pascal) - *int*(pentru C/C++):

```

i := 1; ok := true; k := 0;
while (i<=length(s))and ok do
begin
  if s[i]=x then begin
    k:=i; ok:=false;
  end;
  inc(i);
end;

```

```

i = 0;
ok = 1; k = 0;
while(i<strlen(s) && ok){
  if(s[i]==x)
  {
    k = i;
    ok = 0;
  }i++;}

```

Care dintre atribuirile următoare conduc la obținerea aceleiași valori pentru *k*, ca cea obținută în urma execuției secvenței prezentate ?

a) `k := concat(s,x)`
 b) `k := length(s)`
 c) `k := pos(x,s)`
 d) `k := pos(s,x)`

a) `k = strcat(s,x)`
 b) `k = strlen(s)`
 c) `k = strchr(s,x)-s`
 d) `k = strchr(s,x)-x`

1.1.2 Probleme rezolvate

1. Se consideră un text în care unicul separator este spațiul. Știind că între oricare două cuvinte pot exista mai mulți separatori, să se determine numărul de cuvinte din text.

Exemplu: Pentru textul 'Am venit repede ' se va afișa 3.

Soluție

Algoritmul presupune parcurgerea caracter cu caracter a textului și identificarea numărului de perechi de caractere alăturate care pot reprezintă finalul unui cuvânt (caracter diferit de spațiu urmat de un separator).

```

1  var s:string;
2      i,nr:integer;
3  begin
4      readln(s);
5      s:=s+' '; nr:=0;
6      for i:=1 to length(s)-1 do
7          if(s[i]<>' ')and
8              (s[i+1]=' ')then inc(nr);
9      writeln(nr);
10 end.
11

```

```

#include <stdio.h>
#include <string.h>
char s[256];int nr,i;
void main() {
  gets(s);
  strcat(s," "); nr=0;
  for (i=0;i+1<strlen(s);i++)
    if (s[i]!=' '&&s[i+1]==' ')
      nr++;
  printf("%d\n",nr);
}

```