

Varianta 37

Subiectul I

ex.1

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Care este valoarea pe care trebuie să o aibă inițial variabila întreagă x pentru ca, în urma executării secvenței alăturate, să se afișeze șirul de mai jos?

HHHHHH

(4p.)

```
while (x!=3){  
    x=x-1;  
    cout<<"HH";    |    printf("HH");  
}
```

a. 0

b. 4

c. 6

d. 5

Raspuns corect: c) 6

ex.2

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Se consideră algoritmul alăturat descris în pseudocod.

S-a notat cu $x \div y$ restul împărțirii numărului natural x la numărul natural nenul y și cu $[z]$ partea întreagă a numărului real z .

- a) Scrieți numărul afișat dacă se citesc valorile $n=1232$ și $k=2$. (6p.)
- b) Scrieți toate perechile de valori care pot fi citite pentru n și k , cu $n < 100$, astfel încât în urma executării algoritmului valoarea afișată să aibă 4 cifre. (4p.)
- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)
- d) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)

```
citește n, k  
(numere naturale,  $k \leq 9$ )  
nr ← 0; p ← 1  
cât timp  $n \neq 0$  execută  
    c ←  $n \div 10$   
    nr ←  $nr + c * p$   
    p ←  $p * 10$   
    dacă  $c = k$  atunci  
        nr ←  $nr + c * p$   
        p ←  $p * 10$   
    ■  
    n ←  $[n / 10]$   
■  
n ← nr  
scrie n
```

a)

$n = 1232$

$k = 2$

-> se afiseaza 122322

Daca ne uitam la program ne dam seama ca rolul acestuia este sa il rescrie pe n astfel incat fiecare cifra aparitie a cifrei k sa fie dublata

b) Scrieți toate perechile de valori care pot fi citite pentru n și k , cu $n < 100$, astfel încât în urma executării algoritmului valoarea afișată să aibă 4 cifre. (4p.)

b)

Raspuns:

- 11 1
- 22 2
- 33 3
- 44 4
- 55 5
- 66 6
- 77 7
- 88 8
- 99 9

c)

c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura `cât timp...execută` cu o structură repetitivă de un alt tip. (6p.)

citește n, k

(numere naturale, $k \leq 9$)

$nr \leftarrow -0$; $p \leftarrow -1$

dacă $n \neq 0$ atunci

┌execută

| $c \leftarrow n \% 10$

| $nr \leftarrow nr + c * p$

| $p \leftarrow p * 10$

| ┌dacă $c = k$ atunci

| | $nr \leftarrow nr + c * p$

| | $p \leftarrow p * 10$

| └■

| $n \leftarrow [n / 10]$

└■ cât timp $n \neq 0$

$n \leftarrow -nr$

scrie n

d)

```
main() : int
1 //Varianta 37, Subiectul I, ex.2 d)
2
3 #include <bits/stdc++.h>
4 using namespace std;
5 int main()
6 {
7     int n,k;
8     cin >> n >> k;
9     int nr = 0;
10    int p = 1;
11    while(n!=0){
12        int c = n%10;
13        nr = nr + c*p;
14        p = p*10;
15        if(c==k){
16            nr=nr+c*p;
17            p=p*10;
18        }
19        n/=10;
20    }
21    n=nr;
22    cout << n;
23 }
```

Subiectul II

ex.1

1. Fie declarațiile alăturate. Dacă variabila **x** reține informații despre un elev, precizați care este varianta corectă ce afișează prima literă din numele acestuia? (4p.)
- | | |
|---|--|
| <p>a. <code>cout<<nume; printf("%c",nume);</code></p> <p>b. <code>cout<<x; printf("%c",x);</code></p> <p>c. <code>cout<<x.nume; printf("%c",x.nume);</code></p> <p>d. <code>cout<<x.nume[0]; printf("%c",x.nume[0]);</code></p> | <pre>struct elev{ char nume[30]; float nota;}; elev x;</pre> |
|---|--|

Raspuns: d)

ex.2

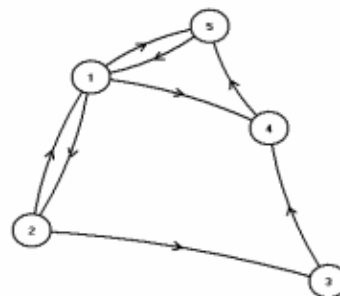
2. Într-o listă liniară simplu înlănțuită, alocată dinamic, fiecare nod reține în câmpul **adr** adresa următorului nod din listă sau **NULL** dacă nu există un nod următor, iar în câmpul **info** un număr întreg. Variabilele **d** și **q** rețin adresele câte unui nod (nodul referit de **d** este inclus în listă, iar cel referit de **q** nu aparține listei). Să se identifice secvența de instrucțiuni care realizează inserarea corectă, în listă, a nodului referit de variabila **q**, ca succesor al nodului referit de variabila **d**. (4p.)
- a. **d->adr=q; q->adr=d;**
- b. **q->adr=d->adr; d->adr=q;**
- c. **d->adr=q; q->adr=d->adr;**
- d. **d=q; q->adr=d->adr;**

Raspuns: b)

ex.3

Scrieti pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

3. Se consideră un graf orientat cu 5 vârfuri reprezentat în figura alăturată.
- a) Care este matricea de adiacență corespunzătoare grafului? (6p.)
- b) Scrieți vârfurile care au gradul intern maxim. (6p.)



a)

```
0 1 0 1 1
1 0 1 0 0
0 0 0 1 0
0 0 0 0 1
1 0 0 0 0
```

b)

1,5,4

ex.4

4. Un șir cu maximum 255 de caractere conține cuvinte separate prin unul sau mai multe spații. Cuvintele sunt formate numai din litere mici ale alfabetului englez. Scrieți un program C/C++ care citește un astfel de șir și îl afișează modificat, prima și ultima literă a fiecărui cuvânt fiind afișată ca literă mare.

Exemplu: pentru şirul: maine este proba la informatica se va afişa:

MainE EstE ProbA LA InformaticaA

(10p.)

Raspuns:

```
1 //Varianta 37, Subiectul II, ex.4
2 #include <bits/stdc++.h>
3 using namespace std;
4 int main()
5 {
6     char s[256];
7     cin.get(s,255);
8     for(int i=0;i<strlen(s);i++)
9     {
10         if(i==0 || i==strlen(s)-1 || s[i-1]==' ' && s[i]!=' ' || s[i+1]==' ' && s[i]!=' ')
11             s[i]--32;
12     }
13     cout << s;
14 }
```

Subiectul III

ex.1

1. Se utilizează un algoritm pentru a genera în ordine lexicografică inversă toate permutările mulțimii {1,2,3,4,5}. Primele patru permutări generate sunt: 54321, 54312, 54231, 54213. A cincea permutare este: (4p.)
- a. 53421 b. 54321 c. 54132 d. 54123

Raspuns: c)

ex.2

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Considerăm subprogramul f definit alăturat. Ce valoare are $f(7,11)$? Dar $f(11,7)$? (6p.)
- ```
int f(int x,int y)
{
 if(x<=y) return x-y;
 return f(y-x,x-1)+3;
}
```

Raspuns: -4, respectiv -11

1.

pentru  $f(7,11) \rightarrow x = 7, y = 11$

$7 \leq 11$  adv

**La final se returneaza -4**

2.

pentru  $f(11,7) \rightarrow x = 11, y = 7$

$11 \leq 7$  fals

returneaza  $f(-4,10)+3$

$f(-4, 10)$   
 $-4 \leq 10$  adv  
 returneaza  $-4 - 10 = -14$   
 intoarcere la  $f(11, 7) \rightarrow$  returneaza  $-14 + 3 = -11$

**La final se returneaza -11**

ex.3

3. Scrieți un program C/C++ care citește numerele naturale nenule  $n$  și  $k$  ( $k \leq n \leq 100$ ) și un tablou unidimensional cu  $n$  elemente numere întregi, fiecare având cel mult 4 cifre. Programul modifică tabloul, permutând circular, cu  $k$  poziții spre stânga, elementele acestuia și afișează pe ecran, separate prin câte un spațiu, elementele tabloului obținut.

**Exemplu:** dacă  $n=4$ ,  $k=3$  și tabloul  $v=(1,2,3,4)$ , atunci se vor afișa în ordine elementele: 4 1 2 3. (10p.)

Raspuns:

```

1 //Varianta 37, Subiectul III, ex.3
2 #include <bits/stdc++.h>
3 using namespace std;
4 int v[101], v2[101];
5 int main()
6 {
7 int n, k;
8 cin >> n >> k;
9 for(int i=1; i<=n; i++)
10 {
11 cin >> v[i];
12 int q = i-k;
13 if(q<=0)
14 v2[n+q]=v[i];
15 else v2[q]=v[i];
16 }
17 for(int i=1; i<=n; i++)
18 cout << v2[i] << " ";
19 }
```

#### ex.4

4. a) Scrieți doar antetul subprogramului `nrdiv`, care primește prin intermediul parametrului `x` un număr natural nenul cu cel mult 4 cifre, și returnează numărul de divizori primi ai lui `x`. (4p.)

b) Pe prima linie a fișierului `bac.in` se află un număr natural nenul `n` ( $n \leq 1000$ ), iar pe a doua linie a fișierului se află un șir format din `n` numere naturale nenule, despărțite prin câte un spațiu, fiecare număr fiind format din cel mult 4 cifre. Scrieți un program C/C++ care citește numerele din fișier și care afișează pe ecran, folosind apeluri utile ale subprogramului `nrdiv`, prima și ultima valoare din șirul celor `n` numere citite, care au un număr par de divizori primi. Numerele afișate vor fi separate printr-un spațiu.

**Exemplu:** dacă fișierul `bac.in` are conținutul alăturat, pe ecran se va afișa: 20 10

(6p.)

- a)  
`unsigned int nrdiv(unsigned int x)`  
 b)

```

1 //Varianta 37, Subiectul III, ex.4 b)
2 #include <bits/stdc++.h>
3 using namespace std;
4 int x,n;
5 ifstream fin("bac.in");
6
7 int prim(int x){
8 if(x==1 || x==0)
9 return 0;
10 if(x==2 || x==3)
11 return 1;
12
13 for(int i=2;i<=sqrt(x);i++)
14 if(x%i==0)
15 return 0;
16 return 1;
17
18 }
19
20 unsigned int nrdiv(unsigned int x){
21 if(prim(x))
22 return 1;
23 unsigned int nr=0;
24 for(int i=2;i<=x/2;i++)
25 if(x%i==0 && prim(i))
26 nr++;
27 return nr;
28 }

```

```
28 }
29 int main()
30 {
31 fin >> n;
32 int u_nr_div_pari = -1;
33 for(int i=1; i<=n; i++){
34 fin >> x;
35 if(nrdiv(x)%2==0 && u_nr_div_pari==-1){
36 cout << x << " ";
37 u_nr_div_pari = x;}
38 else if(nrdiv(x)%2==0) u_nr_div_pari = x;
39 }
40 cout << u_nr_div_pari;
41 }
```