

FIȘA 14. Subprograme

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

1. Parametri utilizați la apelul unui subprogram, se numesc:
a. parametri actuali b. parametri locali c. parametri globali d. parametri formali
2. Funcția `s` realizează interschimbarea valorilor variabilelor `a` și `b` prin apelul `s(a,b)`. Care este definiția corectă a funcției `s`?
a. `void s(int x,int y) {int z=y; y=x; x=y; }`
b. `void s(int x,int y) { x=y+x; x=y-x; y=y-x; }`
c. `void s(int &x,int &y){ x=x-y; y=x+y; x=y-x; }`
d. `void s(int &x,int &y) { x=y+x; x=y-x; y=y-x; }`
3. Care este antetul corect al unui subprogram care primește prin doi parametri `a` și `b` două numere întregi cu maximum 6 cifre fiecare și returnează prin al treilea parametru `c` media aritmetică dintre `a` și `b` cu exact două zecimale?
a. `void f(int a, int b, float c)` b. `void f(long a, long b, float &c)`
c. `float f(int a, int b)` d. `float f(long a, long b)`
4. Știind că funcția `cmmdc`, cu doi parametri întregi `a` și `b`, returnează valoarea celui mai mare divizor comun al numerelor `a` și `b`, stabiliți care dintre următoarele variante atribuie variabilei `m` valoarea celui mai mare divizor comun al numerelor `a`, `b` și `c`.
a. `m=cmmdc(cmmdc(a,b),c);` b. `m=cmmdc(a,b)*c;`
c. `m=cmmdc(a,b,c);` d. `m=cmmdc(a,b)/c;`
5. Funcția `f` cu parametrii `x` și `y` determină numărul de numere întregi pare din intervalul `[x,y]`, (`x<y`). Care din următoarele expresii este adevărată pentru orice numere `x` și `y` care nu au aceeași paritate?
a. `f(x,y)==y-x` b. `f(x,y)==(y-x+1)/2`
c. `f(x,y)==(y-x)/2` d. `f(x,y)==f(x,y+1)`
6. Care dintre următoarele reprezintă un antet corect folosit la definirea subprogramului `P`, astfel încât programul principal alăturat să afișeze valorile 1 9?
a. `void P (int a, int b)` b. `int P (int a, int b)`
c. `int P (int &x, int y)` d. `void P(int x, int &y)`

```
void main()
{int a=1,b=5;
 P(a,b);
 cout<<a<<' '<<b; | printf("%d %d",a,b);}
```
7. Subprogramul `maxim(x,y)` returnează cea mai mare valoare dintre `x` și `y`. Care dintre apelurile următoare ale funcției `maxim` NU returnează valoarea absolută a numărului real `x`?
a. `maxim(x,0)` b. `maxim(x,maxim(-x,0))`
c. `maxim(x,-x)` d. `maxim(-x,maxim(x,0))`
8. Subprogramul `complement` schimbă cifrele unui număr natural `n`, astfel încât fiecare cifră `x` devine egală cu `9-x`. Care este antetul corect al unui astfel de subprogram?
a. `void complement(long &n)` b. `void complement(float n)`
c. `void complement(real &n)` d. `void complement(long n)`
9. Antetul corect al unui subprogram care are ca parametri de intrare două numere întregi `a`, `b` și returnează prin parametrii `M` și `P`, media aritmetică și respectiv produsul celor două numere, este :
a. `int calcul(int a, int b, int &P);`
b. `void calcul(int a, int b, float M, int P);`
c. `void calcul(int a, int b, float &M, int &P);`
d. `float calcul(int a, int b);`

10. Fie funcției f definită alăturat. Se știe că x , y și z sunt variabile întregi. Inițial $x=5$ și $y=12$, iar după apelul $f(x,y,z)$, valorile celor trei variabile sunt $x=6$, $y=12$ și $z=17$. Care este antetul complet al funcției f ?
- ```
void f(....)
{
 a++;
 b--;
 c=a+b;
}
```
- a. `void f(int &a,int b,int &c)`                      b. `void f(int a,int &b,int &c)`  
c. `void f(int a,int b,int c)`                         d. `void f(int &a,int &b,int &c)`
11. Subprogramul `putere(n)` returnează numărul întreg  $10^{x-1}$ , unde  $x$  reprezintă numărul de cifre ale numărului  $n$  primit ca parametru ( $n > 99$ ). Atunci expresia  $n/putere(n)$  reprezintă:
- a. prima cifră a numărului  $n$                       b. ultima cifră a numărului  $n$   
c. numărul obținut prin eliminarea ultimei cifre a numărului  $n$                       d. numărul obținut prin eliminarea primei cifre a numărului  $n$
12. Care dintre următoarele secvențe de instrucțiuni este corectă, știind că se folosește funcția cu antetul `void f(int a, int &b)?`
- a. `a=1; b=1; f(a,b);`  
b. `b=1; f(b,1);`  
c. `a=1; f(a,1);`  
d. `a=1;b=1; printf("%d",f(a,b)); |cout<<f(a,b);`
13. Subprogramul `s` este utilizat pentru interschimbarea valorilor a două variabile întregi. Antetul corect al acestuia este:
- a. `void s(int x, int &y)`                              b. `void s(int x, int y)`  
c. `void s(int &x, &y)`                                 d. `void s(int &x, int &y)`
14. Fie subprogramul `f` definit alăturat și variabilele  $a$  și  $b$  de tip `int`, cu valorile  $a=1$  și  $b=2$ . Ce valori vor avea variabilele  $a$  și  $b$  în urma apelului `f(a,b+1)?`
- ```
void f(int &x,int y){
  x=x+y;
  y=x-y;
  x=x-y;
}
```
- a. `a=2,b=1` b. `a=3,b=2` c. `a=1,b=2` d. `a=2,b=2`
15. Subprogramul `s` returnează în parametrii d și m , cel mai mare divizor comun și respectiv cel mai mic multiplu comun a două numere întregi transmise prin parametrii a și b . Antetul corect al subprogramului `s` este:
- a. `int S(int a,int b,int d, int m)`
b. `void S(int a,int b,int &d,int &m)`
c. `void S(int a,b,int &d,&m)`
d. `float S(int a,b)`

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

1. Scrieți definiția completă a subprogramului `prime` care are ca parametri două numere naturale x și y (având cel mult 4 cifre fiecare) și afișează pe ecran toate numerele prime din intervalul închis, având unul din capete în x și celălalt în y . Numerele afișate vor fi separate prin câte un spațiu.
Exemplu: pentru $x=32$ și $y=18$ valorile afișate vor fi `19 23 29 31`.
2. Scrieți definiția completă a subprogramului `impar` care primește prin parametrul x un tablou unidimensional cu cel mult 100 de elemente numere naturale, fiecare având cel mult 9 cifre, iar prin parametrul n o valoare naturală reprezentând numărul efectiv de elemente ale tabloului ($1 \leq n \leq 100$) și afișează mesajul `DA` în cazul în care printre elementele tabloului x se află cel puțin un număr impar, sau afișează mesajul `NU` în caz contrar.

3. Scrieți în limbajul C/C++ definiția completă a subprogramului `ordonare` care primește ca parametru un tablou unidimensional `x` cu cel mult 100 de elemente, numere reale, și un număr natural `n` ($n \geq m$), ce reprezintă numărul efectiv de elemente ale tabloului `x`. Subprogramul va afișa tabloul obținut în urma schimbării poziției doar a elementelor impare din tablou astfel încât acestea să apară în ordinea crescătoare a valorilor lor.

Exemplu: pentru $n=6$ și $x=(7, 11, 2, -8, -3, 10)$

subprogramul va afișa $-3, 7, 2, -8, 11, 10$

4. Scrieți definiția completă a subprogramului `Cifra`, cu doi parametri, `n` și `x`, care primește prin intermediul parametrului `n` un număr natural de cel mult nouă cifre și furnizează prin parametrul `x` cea mai mare cifră a numărului transmis prin parametrul `n`.

5. Scrieți în limbajul C/C++ definiția completă a subprogramului `Invers`, cu doi parametri, `n` și `x`, care primește prin intermediul parametrului `n` un număr natural de cel mult nouă cifre și furnizează prin parametrul `x` numărul obținut prin inversarea ordinii cifrelor sale, ca în exemplu.

Exemplu: dacă $n=78904$ atunci, după apel, $x=40987$.

6. Scrieți definiția completă a unui subprogram `s` care primește prin intermediul parametrului `n` un număr natural de maximum 9 cifre, prin intermediul parametrului `c` o cifră și furnizează prin intermediul parametrului `k` numărul de cifre ale lui `n` care aparțin intervalului $[c-1, c+1]$.

Exemplu: pentru $n=1233$ și $c=3$, `k` va avea valoarea 3, iar pentru $n=650$ și $c=3$, `k` va avea valoarea 0.

7. Să se scrie în limbajul C/C++ definiția completă a subprogramului `calcul`, care primește prin parametrul `n` un număr natural nenul de cel mult 9 cifre și furnizează prin parametrul `x` numărul obținut prin citirea cifrelor pare ale lui `n` de la dreapta la stânga. Dacă `n` nu conține nicio cifră pară `x` primește valoarea 0.

Exemplu: în urma apelului `calcul(9278, x)`, `x` primește valoarea 82.

8. Scrieți definiția completă a subprogramului `numar`, cu patru parametri, care primește prin intermediul parametrului `n` un număr natural format din cel mult 9 cifre, iar prin intermediul parametrilor `c1` și `c2` câte o cifră, cu proprietatea $c1 < c2$; subprogramul furnizează prin intermediul celui de al patrulea parametru, `x`, o valoare egală cu suma cifrelor care aparțin intervalului închis $[c1, c2]$.

Exemplu: pentru $n=123$, $c1=2$ și $c2=7$, valoarea furnizată prin `x` va fi 6.

9. Subprogramul `Nr` are un singur parametru, `k`, prin intermediul căruia primește un număr natural de cel puțin 3 cifre și cel mult 9 cifre, cu toate cifrele nenule. Subprogramul furnizează tot prin intermediul parametrului `k`, valoarea obținută prin eliminarea primei cifre a numărului transmis la apel.

Exemplu: dacă subprogramul primește prin intermediul parametrului `k` valoarea 12438, atunci în urma apelului subprogramului `Nr`, `k` va primi valoarea 2438.

Scrieți, în limbajul C/C++, definiția completă a subprogramului `Nr`.

10. Scrieți definiția completă a subprogramului `numar`, cu trei parametri, care primește prin intermediul parametrului `n` un număr natural format din cel mult 9 cifre, iar prin intermediul parametrilor `c1` și `c2` câte o cifră nenulă. Subprogramul caută fiecare apariție a cifrei `c1` în `n`, și dacă aceasta apare, o înlocuiește cu `c2`. Subprogramul furnizează tot prin `n` numărul astfel obținut. Dacă cifra `c1` nu apare în `n`, atunci valoarea lui `n` rămâne nemodificată.
Exemplu: pentru `n=149448`, `c1=4` și `c2=2`, valoarea furnizată prin `n` va fi `129228`.
11. Subprogramul `suma`, cu trei parametri, primește prin intermediul parametrilor `x` și `y` două numere naturale de cel mult 4 cifre fiecare și furnizează, prin intermediul parametrului `z`, suma numerelor pare cuprinse între `x` și `y` inclusiv.
Exemplu: dacă la apel sunt transmise valorile `x=12` și `y=23`, atunci subprogramul `suma` va returna, prin intermediul parametrului `z`, valoarea `102`.
Scrieți, în limbajul C/C++, definiția completă a subprogramului `suma`.
12. Să se scrie în limbajul C/C++ definiția completă a subprogramului `ca1cu1`, care primește prin intermediul parametrului întreg `n`, un număr natural de cel mult 9 cifre și returnează valoarea absolută a diferenței dintre numărul de cifre pare și numărul de cifre impare conținute de `n`.
Exemplu: în urma apelului `ca1cu1(92465)` se va returna valoarea `1`.
13. Scrieți în limbajul C/C++ definiția completă a subprogramului `Cifre` care primește prin parametrii `a` și `b` două numere naturale ($0 < a < 200000000$, $0 < b < 200000000$) și returnează numărul de cifre comune ale celor două numere.
Exemplu: dacă `a = 123446` și `b = 248766` atunci subprogramul va returna `3`, iar dacă `a = 1244` și `b = 4456` subprogramul va returna `1`.
14. Scrieți definiția completă a subprogramului `numar`, cu doi parametri, care primește prin intermediul parametrului `n` un număr natural format din cel mult 9 cifre, iar prin intermediul parametrului `c` o cifră nenulă; subprogramul returnează numărul obținut prin înlocuirea fiecărei apariții a cifrei `c` în numărul `n` cu partea întreagă a valorii `c/2`. Dacă `c` nu apare în `n`, subprogramul returnează valoarea `n`.
Exemplu: pentru `n=1525735`, `c=5` valoarea returnată va fi `1222732`.
15. Scrieți definiția completă a subprogramului `numar`, cu doi parametri, care primește prin intermediul parametrilor `a` și `b` două numere naturale, `a` fiind format din cel mult 9 cifre, iar `b` fiind un număr natural strict mai mic decât numărul de cifre ale lui `a`. Subprogramul înlocuiește cu `0` ultimele `b` cifre ale numărului `a` și returnează valoarea astfel obținută.
Exemplu: pentru `a=184465709` și `b=5`, valoarea returnată va fi `184400000`.
16. Scrieți în limbajul C/C++ definiția completă a subprogramului `numar`, cu exact doi parametri, care primește prin intermediul parametrului `x` un număr natural de cel mult 2 cifre, și prin intermediul parametrului `y` un număr natural nenul de cel mult 9 cifre. Subprogramul returnează cel mai mare număr natural `z` pentru care există un număr natural `k` astfel încât $z = x^k$ și $z < y$.
Exemplu: pentru `y=18` și `x=2` subprogramul va returna valoarea `16` ($=2^4 < 18$).
17. Scrieți definiția completă a subprogramului `divizor`, cu trei parametri, prin care primește 3 numere naturale nenule cu cel mult 9 cifre fiecare și returnează numărul divizorilor comuni tuturor celor 3 numere.
Exemplu: dacă numerele primite ca parametri sunt `24`, `20` și `12`, subprogramul returnează valoarea `3` (divizorii comuni sunt `1`, `2` și `4`).

18. Scrieți definiția completă a unui subprogram `i_prim` care primește prin singurul său parametru. `n` un număr natural din intervalul $[2, 30000]$ și returnează diferența minimă $p_2 - p_1$ în care p_1 și p_2 sunt numere prime și $p_1 \leq n \leq p_2$.

Exemplu: dacă $n=20$ atunci `i_prim(10)=4`, valoare obținută pentru $p_1=19$ și $p_2=23$.

19. Un număr natural se numește palindrom dacă numărul citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga.

Scrieți definiția completă a subprogramului `Palindrom` care primește prin intermediul parametrului `n` un număr natural de cel mult nouă cifre și returnează 1 dacă acesta este palindrom și 0 în caz contrar.

20. Scrieți o funcție care primește prin intermediul parametrului `v` un vector cu cel mult 50 elemente numere reale și prin parametrul `n` o valoare naturală ($2 \leq n \leq 50$). Funcția returnează valoarea 1 dacă primele `n` elementele ale vectorului sunt în ordine strict crescătoare, 2 dacă primele `n` elementele ale vectorului sunt în ordine strict descrescătoare și 0 dacă primele `n` elemente ale vectorului nu sunt nici în ordine strict crescătoare, nici în ordine strict descrescătoare.

21. Realizați următoarele cerințe utilizând limbajul C/C++:

a) Scrieți definiția completă a unui subprogram `sub` cu doi parametri care:

- primește prin intermediul parametrilor `a` și `b` două numere naturale formate fiecare din cel mult 8 cifre;
- elimină cifra unităților numărului conținut de parametrul `a`, mutând-o la sfârșitul numărului conținut de parametrul `b`;
- returnează prin intermediul parametrilor `a` și `b` noile numere obținute.

De exemplu, pentru valorile 1234 și 56 ale parametrilor `a` și `b`, în urma executării subprogramului `sub` valorile returnate prin intermediul parametrilor vor fi 123 și 564.

b) Scrieți programul care citește de la tastatură un număr natural `n` cu cel mult 8 cifre și care verifică dacă numărul `n` este un palindrom, folosind apeluri ale subprogramului `sub`. Programul va afișa pe ecran mesajul **DA** dacă numărul este palindrom, altfel va afișa mesajul **NU**.

De exemplu, dacă $n=12321$ atunci se va afișa pe ecran mesajul **DA**, iar dacă $n=124321$ atunci se va afișa pe ecran mesajul **NU**.

22. Știind că sunt definite subprogramele:

- `s1` care primește prin intermediul parametrului `n` un număr natural de cel mult 8 cifre și care returnează numărul de cifre ale numărului `n`;
- `s2` cu doi parametri, care primește prin intermediul parametrului `n` un număr natural de cel mult 4 cifre și prin intermediul parametrului `k` o cifră. Subprogramul dublează fiecare apariție a cifrei `k` în numărul `n` și returnează numărul nou format prin intermediul parametrului `n`. (De exemplu, pentru $n=1232$ și $k=2$ valoarea returnată este 122322, iar pentru $n=1232$ și $k=4$ valoarea returnată este 1232.)

a) Scrieți numai antetul fiecăruia dintre cele două subprograme.

b) Scrieți în limbajul C sau C++ programul în care se citesc de la tastatură un număr natural `n` de cel mult 8 cifre și două cifre distincte `a` și `b`. Programul va determina și va afișa câte dintre cifrele numărului `n` sunt diferite atât de `a` cât și de `b`, fără a accesa cifrele numărului `n`, folosind apeluri ale subprogramelor `s1` și `s2`. Programul va afișa pe ecran numărul determinat.

De exemplu, dacă $n=121532$, $a=3$ și $b=2$, atunci numărul determinat este 3, deoarece `n` are trei cifre diferite de 3 și de 2

23. Considerăm definite următoarele două subprograme:

- `pozmax` – cu doi parametri: `v` (un tablou unidimensional format din n ($1 \leq n \leq 100$) numere întregi de cel mult 3 cifre fiecare) și `p` (un număr natural cuprins între 1 și n); subprogramul returnează poziția valorii maxime din secvența `v[1], ..., v[p]`.
- `schimb` – cu doi parametri: `a` și `b` (numere întregi formate din cel mult trei cifre fiecare); subprogramul interschimbă valorile a două variabile transmise prin intermediul parametrilor `a` și `b`.

a) Scrieți numai antetul fiecăruia dintre cele două subprograme.

b) Scrieți un program care citește de la tastatură un număr natural n ($1 \leq n \leq 100$) și apoi cele n elemente (numere întregi formate din cel mult 3 cifre fiecare) ale unui tablou unidimensional, sortează crescător tabloul apelând subprogramele descrise mai sus și apoi îl afișează pe ecran, pe o singură linie, elementele tabloului fiind despărțite prin câte un spațiu.

24. Se definește subprogramul `sub` cu doi parametri, subprogram care primește prin intermediul parametrului `n` un număr natural de cel mult opt cifre și prin intermediul parametrului `k` o cifră. Subprogramul determină eliminarea primei apariții a cifrei `k` în numărul `n` și returnează numărul obținut prin intermediul parametrului `n`. De exemplu, pentru valorile $n=152422$ și $k=2$ ale parametrilor, în urma apelului subprogramului `sub`, parametrul `n` va returna valoarea 15422. Dacă cifra `k` nu apare în scrierea numărului `n`, atunci valoarea lui `n` nu se modifică.

a) Scrieți definiția completă a subprogramului `sub`.

b) Scrieți în limbajul C sau C++ programul în care se citește de la tastatură două numere naturale `a` și `b` cu cel mult opt cifre. Programul va determina și va afișa pe ecran numărul de cifre distincte ce intră atât în scrierea lui `a` cât și în scrierea lui `b`, fără a accesa cifrele numerelor `a` și `b`, folosind apeluri ale subprogramului `sub`.

De exemplu, pentru valorile $a=1237248$ și $b=1245823$ programul va determina afișarea pe ecran a valorii 5 deoarece sunt cinci cifre distincte (1, 2, 3, 4 și 8) ce apar atât în scrierea lui `a` cât și în scrierea lui `b`.

25. Realizați următoarele cerințe utilizând limbajul C/C++:

a) Scrieți definiția completă a unei funcții `produs` cu un parametru, funcție care:

- primește prin intermediul parametrului `a` un număr natural cu cel mult 8 cifre ($a > 1$);
- returnează produsul divizorilor primi ai valorii parametrului `a`.

De exemplu, pentru valoarea 300 a parametrului `a`, funcția va returna valoarea 30 deoarece $a=2^2 \cdot 3 \cdot 5^2$ și $2 \cdot 3 \cdot 5=30$.

b) Scrieți programul în care se citește de la tastatură numerele naturale nenule `a`, `b` și `c` de cel mult 8 cifre fiecare ($a > 1$, $b > 1$, $c > 1$) și în care se verifică dacă numerele citite au exact aceiași factori în descompunerea lor în factori primi, folosind apeluri ale subprogramului `produs`. Dacă cele trei numere au exact aceiași factori (exponenții lor putând să difere), programul va afișa pe ecran mesajul DA, în caz contrar va afișa mesajul NU.

De exemplu, dacă $a=300$, $b=1500$ și $c=30$, atunci se va afișa mesajul DA, iar pentru numerele $a=300$, $b=700$ și $c=140$, se va afișa mesajul NU.

26. Se consideră subprogramul `NRDIV`, cu 3 parametri, care:

- primește prin intermediul parametrului `n` un număr natural nenul, $2 \leq n \leq 10000$;
- returnează prin intermediul parametrilor `d1` și `d2` cel mai mic, respectiv cel mai mare divizor prim al său ($1 < d1 \leq d2 \leq n$).

Pentru $n=6$, se obțin $d1=2$ și $d2=3$, pentru $n=8$, se obțin $d1=2$ și $d2=2$, iar pentru $n=7$ se obțin $d1=7$ și $d2=7$.

a) Scrieți definiția completă a subprogramului `NRDIV`

b) Scrieți un program C/C++ care citește două numere naturale `a` și `b`, de cel mult 4 cifre fiecare, și determină prin apeluri utile ale subprogramului `NRDIV` toate numerele naturale cuprinse în intervalul `[a, b]` cu proprietatea că toți divizorii lor primi au exact două cifre. Numerele determinate vor fi afișate pe ecran, cu spații între ele.

Exemplu: dacă $a=120$, $b=200$ se vor afișa numerele: 121 143 169 187

27. Subprogramul `cifre_pare` primește ca parametru un număr natural n având cel mult 9 cifre și verifică dacă numărul n are toate cifrele pare.

a) Scrieți definiția completă a subprogramului `cifre_pare`.

b) Scrieți programul C/C++ care folosind apelul subprogramului `cifre_pare`, afișează pe ecran numerele naturale mai mici decât 10000, care au toate cifrele pare.

28. a) Scrieți definiția completă a unui subprogram recursiv `sum` care primește prin parametrul x un număr natural de cel mult 4 cifre și returnează suma divizorilor numărului x , diferiți de 1 și de el însuși.

Exemplu: dacă $x=10$ se va returna valoarea 7 ($7=2+5$).

b) Scrieți programul C/C++ care citește de la tastatură un număr natural n ($0 < n < 100$), apoi n numere naturale (cu cel mult 4 cifre fiecare). Programul determină, folosind apeluri utile ale subprogramului `sum`, pentru fiecare număr natural, suma divizorilor săi proprii și afișează pe ecran sumele determinate, în ordinea crescătoare a valorilor lor, separate prin câte un spațiu.

Exemplu: dacă $n=5$ și numerele citite sunt 10 2 33 6 11

valorile afișate pe ecran vor fi: 0 0 5 7 14

deoarece suma divizorilor lui 10 este 7, suma divizorilor lui 2 este 0, suma divizorilor lui 33 este 14, suma divizorilor lui 6 este 5, suma divizorilor lui 11 este 0.

29. Se consideră definite următoarele subprograme:

- `p1` care primește prin intermediul parametrului n un număr natural cu cel mult 8 cifre și returnează suma cifrelor numărului primit prin parametrul n

Exemplu: dacă n este egal cu 1234 valoarea returnată de subprogram va fi 10.

- `p2` care primește prin intermediul parametrului n un număr natural cu cel mult 8 cifre, elimină ultima cifră a acestui număr și returnează noul număr obținut.

Exemplu: dacă n este egal cu 1234 valoarea returnată de subprogram va fi 123.

a) Scrieți numai antetul subprogramelor `p1` și `p2`.

b) Scrieți un program C/C++ care citește de la tastatură un număr natural nenul n cu cel mult 8 cifre și determină, prin apeluri utile ale subprogramelor `p1` și `p2`, numărul de cifre egale cu 0 din scrierea lui n . Programul va afișa pe ecran numărul obținut.

Exemplu: dacă n este 102030, programul va afișa valoarea 3.

30. a) Scrieți definiția completă a unui subprogram `sterge`, cu trei parametri, care:

- primește prin parametrii:

- v un tablou unidimensional cu maximum 100 de elemente întregi din intervalul $[-1000; 1000]$

- n o valoare naturală reprezentând numărul de elemente din tabloul v

- i o valoare naturală cu $1 \leq i \leq n$

- elimină din tabloul v elementul $v[i]$ și actualizează valoarea lui n .

Tabloul modificat este furnizat tot prin parametrul v .

b) Fișierul text `NUMERE.IN` conține pe prima linie un număr natural nenul n ($1 \leq n \leq 100$) și pe următoarea linie n numere întregi din intervalul $[-1000; 1000]$, separate prin câte un spațiu. Scrieți un program C/C++ care citește din fișierul `NUMERE.IN` numărul natural n , construiește în memorie un tablou unidimensional v cu cele n numere întregi aflate pe linia a doua în fișier și utilizează apeluri utile ale subprogramului `sterge` pentru a elimina din tablou un număr minim de elemente astfel încât să nu existe două elemente alăturate cu aceeași valoare. Elementele tabloului obținut se afișează pe ecran, separate prin câte un spațiu.

Exemplu: Dacă fișierul `NUMERE.IN` are conținutul:

12

10 10 2 2 19 9 9 9 9 15 15 15 atunci se afișează 10 2 19 9 15.

31. Se consideră subprogramul `nr` cu doi parametri, care primește prin parametrul `n` un număr natural cu maximum 8 cifre, și prin parametrul `c` o cifră zecimală. Subprogramul va returna numărul de apariții ale cifrei `c` în scrierea numărului `n`.

Exemplu: dacă `n=15356`, iar `c=5`, subprogramul va returna valoarea 2.

a) Scrieți doar antetul subprogramului `nr`.

b) Scrieți programul C/C++ care citește de la tastatură un număr natural `n`, cu cel mult 8 cifre, și afișează pe ecran numărul de cifre distincte ale numărului `n`. Se vor folosi apeluri utile ale funcției `nr`.

Exemplu: pentru `n=15356` se va afișa valoarea 4 deoarece numărul conține 4 cifre distincte și anume 1, 3, 5 și 6.

32. Se consideră definite următoarele subprograme:

- `s1`, cu doi parametri: `a`, `b` două numere întregi cu cel mult 4 cifre fiecare; subprogramul interschimbă valorile a două variabile transmise prin intermediul parametrilor `a` și `b`.

- `s2`, cu trei parametri: `a`, un tablou unidimensional cu exact 100 de elemente, numere întregi cu cel mult 4 cifre fiecare, `p`, un număr natural ($p \leq 100$), `q`, un număr natural ($q \leq 100$). Subprogramul caută primul element divizibil cu 5 în secvența a_p, a_{p+1}, \dots, a_q , și returnează poziția acestuia, dacă există un astfel de element, sau valoarea -1 în caz contrar.

a) Scrieți numai antetul subprogramului `s1`.

b) Scrieți definiția completă a subprogramului `s2`.

c) Scrieți programul C/C++ care citește de la tastatură o valoare naturală `n` ($0 < n \leq 100$) și apoi un tablou unidimensional `a`, cu `n` elemente, numere întregi cu cel mult 4 cifre fiecare. Programul determină, folosind apeluri utile ale subprogramului `s2`, primul element divizibil cu 5 (dacă există) și ultimul element divizibil cu 5 (dacă există) al tabloului `a`, interschimbă valorile elementelor găsite, folosind apelul subprogramului `s1`, și apoi scrie pe prima linie a fișierului text `BAC.TXT` elementele tabloului `a`, astfel transformat, separate prin câte un spațiu, sau valoarea 0 dacă tabloul conține mai puțin de două elemente divizibile cu 5.

Exemplu: pentru `n=7` și tabloul `a = (6, 10, 4, 15, 2, 5, 8)`, programul va scrie în fișier:

```
6 5 4 15 2 10 8
```

33. Subprogramul `sub`, cu trei parametri, primește prin intermediul parametrilor:

- `v` un tablou unidimensional cu cel mult 100 de componente ce memorează numere întregi de cel mult 4 cifre fiecare;

- `n` un număr natural nenul mai mic sau egal cu 100 ce reprezintă numărul efectiv de componente ale tabloului primit prin intermediul parametrului `v`;

- `a` un număr întreg cu cel mult 4 cifre.

Subprogramul `sub` returnează numărul componentelor tabloului primit prin intermediul parametrului `v` ale căror valori sunt egale cu valoarea parametrului `a`.

Exemplu: pentru valorile `n=5`, `v=(1,21,9,21,403)`, `a=21` ale parametrilor, în urma apelului, subprogramului `sub` va returna valoarea 2.

a) Scrieți definiția completă a subprogramului `sub`.

b) Scrieți un program C/C++ care să citească de la tastatură un număr natural nenul `n` ($n \leq 100$) și `n` numere întregi, fiecare având cel mult 4 cifre, și care, folosind apeluri utile ale subprogramului `sub`, să afișeze pe ecran mesajul `DA` dacă oricare două dintre cele `n` numere citite sunt distincte două câte două, sau mesajul `NU` în caz contrar.

Exemplu: pentru `n=6` și cele `n` numere citite de la tastatură: 47 183 69 8 134 -56 se va afișa pe ecran mesajul `DA`

34. Subprogramul f , cu un parametru:

- primește prin intermediul parametrului a un număr natural cu cel mult 8 cifre ($a > 1$);
- returnează cel mai mic divizor prim al valorii parametrului a .

Exemplu: pentru valoarea 45 a parametrului a , subprogramul va returna valoarea 3 deoarece $a = 3^2 \cdot 5$, iar cel mai mic divizor prim al său este 3.

a) Scrieți definiția completă a subprogramului f .

b) Scrieți un program C/C++ care să citească de la tastatură un număr natural nenul n ($n \leq 100$) și un șir de n numere naturale de cel mult 8 cifre fiecare, toate numerele din șir fiind strict mai mari decât 1. Folosind apeluri utile ale subprogramului f , programul va determina și va afișa pe ecran toate numerele prime din șirul citit. Numerele determinate se vor afișa pe ecran, separate prin câte un spațiu. Dacă nu există astfel de numere se va afișa pe ecran mesajul **NU EXISTA**.

Exemplu: pentru $n=8$, șirul: 1125, 2, 314, 101, 37, 225, 15, 12 se va afișa:

2 101 37

35. Un număr n se numește *extraprim* dacă atât el, cât și orice număr obținut prin permutarea circulară a cifrelor lui n , sunt numere prime. De exemplu, numărul 197 este un număr *extraprim* deoarece 197, 971, 719 sunt numere prime. Numărul 23 nu este *extraprim* deoarece 32 nu este prim.

a) Scrieți definiția completă a unui subprogram f , cu un parametru, subprogram care:

- primește prin intermediul parametrului a un număr natural cu cel mult 2 cifre ($a > 1$);
- returnează suma tuturor exponenților din descompunerea în factori primi a valorii parametrului a .

Exemplu: pentru $a=90$ subprogramul va returna valoarea 4, deoarece $a = 2 \cdot 3^2 \cdot 5$ și $1+2+1=4$.

b) Scrieți un program C/C++ care citește de la tastatură un număr natural n , $2 \leq n \leq 99$, și care, folosind apeluri utile ale subprogramului f , verifică dacă n este un număr *extraprim* și afișează pe ecran, în caz afirmativ mesajul **DA**, iar altfel mesajul **NU**.

36. Subprogramul *ordonat* are 4 parametri:

- n , prin care primește un număr natural ($1 \leq n \leq 100$);
- v , prin care primește un tablou unidimensional cu n elemente, numerotate de la 1 la n , numere naturale cu cel mult patru cifre fiecare;
- k_1 și k_2 – două numere întregi ($1 \leq k_1 \leq k_2 \leq n$).

Subprogramul returnează valoarea 1 dacă elementele din tablou, cu indici în intervalul $[k_1, k_2]$, se află în ordine crescătoare, și 0 în caz contrar.

a) Scrieți numai antetul subprogramului *ordonat*.

b) Scrieți un program C/C++ care citește de la tastatură un număr natural n ($1 \leq n \leq 100$) și cele n elemente, numerotate de la 1 la n , ale unui tablou unidimensional. Elementele sunt numere naturale, cu cel mult 4 cifre fiecare. Programul determină și afișează pe ecran, separate printr-un spațiu, două valori, k_1 și k_2 , astfel încât secvența formată din elementele cu indici în intervalul $[k_1, k_2]$, pe poziții consecutive în tablou, să fie ordonată crescător și să aibă o lungime maximă. Dacă sunt mai multe secvențe de lungime maximă se vor afișa indicii corespunzători unei valori minime a lui k_1 . Se vor utiliza apeluri utile ale subprogramului *ordonat*.

Exemplu: dacă $n=7$ și tabloul este (4, -6, 1, 2, 1, 4, 5) se va afișa 2 4.

37. Subprogramul `dist` primește prin intermediul parametrului `a` un număr natural cu maximum 8 cifre și returnează 1 dacă cifrele numărului `a` sunt distincte, altfel returnează 0.

a) Scrieți definiția completă a subprogramului `dist`.

b) Fișierul text `date.in` conține pe prima linie un număr natural nenul n ($n \leq 100$), iar pe a doua linie n numere naturale, separate prin câte un spațiu, fiecare număr având maximum 8 cifre. Scrieți un program C/C++ care citește toate numerele din fișier și afișează pe ecran, despărțite prin câte un spațiu, numerele de pe a doua linie a fișierului `date.in` ce au toate cifrele distincte, folosind apeluri utile ale subprogramului `dist`. În cazul în care nu există niciun astfel de număr se va afișa, pe ecran, valoarea 0.

Exemplu: dacă fișierul `date.in` are conținutul

alăturat, atunci se vor afișa numerele:
16 123

```
6  
16 1775 333 242477 123 55566
```

38. Subprogramul `cif`, cu doi parametri, primește prin intermediul parametrului `a` un număr natural cu cel mult 8 cifre și prin intermediul parametrului `b` o cifră; subprogramul returnează numărul de apariții ale cifrei `b` în scrierea numărului `a`.

Exemplu: pentru $a=125854$ și $b=5$, subprogramul va returna valoarea 2.

a) Scrieți definiția completă a subprogramului `cif`.

b) Scrieți un program C/C++ care citește de la tastatură un număr natural n cu cel mult 8 cifre, dintre care cel puțin una impară, și care determină și afișează pe ecran, folosind apeluri utile ale subprogramului `cif`, cel mai mic număr natural care poate fi obținut utilizând toate cifrele impare ale numărului n .

Exemplu: dacă $n=2152331$ atunci se va afișa pe ecran numărul 11335.

39. a) Scrieți definiția completă a unui subprogram `primul`, care

- primește prin singurul său parametru, `a`, o valoare naturală din intervalul $[2, 10000]$

- returnează o valoare naturală reprezentând cel mai mic divizor al numărului `a` mai mare strict decât 1.

b) Fișierul text `NUMERE.IN` conține pe prima linie un număr natural nenul n ($1 \leq n \leq 100$) și pe următoarea linie n numere naturale din intervalul $[2, 10000]$ separate prin câte un spațiu.

Un număr natural n se numește „aproape prim” dacă este egal cu produsul a două numere prime distincte. De exemplu, numărul 14 este „aproape prim” pentru că este egal cu produsul numerelor prime 2 și 7.

Scrieți un program C/C++ care determină și afișează pe ecran (dacă există), separate prin câte un spațiu, folosind apeluri utile ale subprogramului `primul`, numerele „aproape prime” de pe linia a doua a fișierului `NUMERE.IN`.

Exemplu: dacă fișierul `NUMERE.IN` are conținutul:

```
6  
100 14 21 8 77 35 atunci se afișează pe ecran 14 21 77 35
```

40. Se consideră o funcție f care are ca parametri un vector `a` cu maximum 50 de elemente întregi, numărul de elemente al vectorului `n`, număr natural ($n \leq 50$) și un număr întreg `x`. Funcția va returna numărul de elemente din vectorul `a` care sunt strict mai mici decât `x`.

a) Scrieți definiția completă a funcției f .

b) Scrieți un program care citește un vector `a` cu n elemente întregi ($n \leq 50$) și, utilizând apeluri utile ale funcției f , numără și afișează pe ecran câte elemente diferite de 0 sunt în vectorul `a`.