

**Exemen de admitere sesiunea iunie 2014**  
**Informatică**  
**MODEL 1**

**Toate subiectele sunt obligatorii.**  
**Se acordă 10 puncte din oficiu.**  
**Timpul efectiv de lucru este de 3 ore.**

**I. SUBIECTUL I (30 puncte)**

**1. (4p.)** Se consideră avea expresia  $x \% 10 + x / 10$ , unde  $n$  este o variabilă de tip întreg, ce poate memora valori strict pozitive. Pentru câte valori distincte ale lui  $n$ , expresia are valoarea minimă?

- a) o valoare
- b) două valori
- c) 4 valori
- d) 10 valori

**2. (26p.)** Se consideră algoritmul următor, unde  $x \% y$  reprezintă restul împărțirii numerelor naturale  $x$  și  $y$ , iar  $[x]$  reprezintă partea întreagă a numărului real  $x$ .

```
citeste m, n
daca m < n atunci
|   a ← m
|   b ← n
altfel
|   a ← n
|   b ← m
□
p ← 1
k ← 0
cat timp m > 0 executa
|   k ← k*p + m%10
|   p ← p*10
|   m ← [m/10]
□
daca k = n atunci
|   scrie a
altfel
|   scrie b
□
```

- a) (4p.)** Scrieți valoarea afișată la ieșire, dacă de la intrare se citesc numerele 126 și 221
- b) (6p.)** Dacă primul număr citit este 252, cât trebuie să fie al doilea număr citit, așa încât algoritmul să afișeze 252?
- c) (6p.)** Rescrieți algoritmul, folosind în loc de instrucțiunea repetitivă cu test initial, o instrucțiune repetitivă cu test final.
- d) (10p.)** Scrieți programul în limbajul C / C++ corespunzător algoritmului dat.

## II. SUBIECTUL al II-lea (30 puncte)

1. (6p.) Care dintre expresiile următoare nu au valoarea zero, știind că valoarea variabilei reale  $x$ , de tip `double`, se află în mulțimea numerelor reale  $[-7, 3) \cap [-3, 7)$ ?

- a) `fabs(x) < 3`
- b) `(x > -3) && (x < 3)`
- c) `(x >= -3) && (x < 3)`
- d) `(x > -3) && (x <= 3)`
- 3) `fabs(x) <= 3`

2. (10p.) Se consideră variabilele  $n$  și  $k$  de tip `int`. Considerând că valoarea lui  $n$  este pozitivă, să se scrie o secvență de instrucțiuni C, în urma executării căreia, variabila  $k$  să memoreze numărul de cifre ale valorii variabilei  $n$ .

3. (14p.) Dându-se un număr întreg  $n$ , să se determine numărul de divizori primi ai lui  $n$ .

a) (10p.) Scrieți algoritmul de rezolvare al problemei în pseudocod.

b) (4p.) Descrieți rolul tuturor variabilelor din algoritm și indicați datele de intrare și pe cele de ieșire.

## III. SUBIECTUL al III-lea (30 puncte)

1. (5p.) Se consideră șirul  $A = (4, 2, 6, 3)$ . Descrieți succesiunea etapelor prin care trece șirul  $A$ , în cazul sortării crescătoare folosind metoda bulelor, după fiecare trecere.

2. (5p.) În secvența următoare, toate variabilele sunt întregi.

```
gasit = 0;
scanf("%d", &n);
...
for(i=1; i<=n; ++i) {
    scanf("%d", &x);
    ...
}
if(gasit) {
    printf("Cel mai mare numar negative este pe pozitia: ");
    ...
}
```

Rescrieți secvența anterioară, înlocuind punctele de suspensie, astfel încât în urma executării secvenței, să se afișeze poziția celui mai mare număr negativ citit.

3. (10p.) Dându-se un număr întreg  $n$ , să se determine dacă  $n$  se poate scrie ca un produs de numere prime distincte. De exemplu, dacă  $n=21$ , atunci  $n=3*7$ , se poate scrie ca produs de două numere prime distincte, pe când,  $n=12$  nu se poate scrie ca un produs de numere prime distincte ( $n=2*2*3$ ). Scrieți un program C pentru rezolvarea problemei. Programul citește un număr întreg pozitiv și afișează șirul de caractere "DA", în cazul în care numărul se poate scrie ca un produs de numere prime distincte, sau "NU" în caz contrar.

4. (10p.) Se consideră un șir de numere reale  $x_1, x_2, \dots, x_n$  ordonat crescător. Considerându-se în plus un număr real  $x_0$ , să se insereze acest număr printre elementele șirului inițial, așa încât el să rămână ordonat și după inserare. Să se scrie un program C care rezolvă această problemă: citește întâi de la tastatură lungimea șirului, valorile elementelor șirului, presupus ordonat crescător, preum și termenul  $x_0$ , iar apoi înserează  $x_0$  printre termenii șirului, conform specificațiilor problemei și afișează la terminalul standard de ieșire șirul rezultat după inserare.

# Exemen de admitere sesiunea iunie 2014

## Informatică

### MODEL 2

Toate subiectele sunt obligatorii.

Se acordă 10 puncte din oficiu.

Timpul efectiv de lucru este de 3 ore.

#### I. SUBIECTUL I (30 puncte)

1. (4p.) Considerând  $n$  o variabilă de tip întreg, care dintre expresiile următoare determină dacă valoarea lui  $n$  este impară?

- a)  $n \% 2 == 1$
- b)  $(n+1) \% 2 == 0$
- c)  $n \% 2 == 0$
- d)  $(n+1) \% 2 == 1$

2. (26p.) Se consideră algoritmul următor, unde  $x \% y$  reprezintă restul împărțirii numerelor naturale  $x$  și  $y$ , iar  $[x]$  reprezintă partea întreagă a numărului real  $x$ .

```
citeste m, n
daca m < n atunci
|   a ← m
|   b ← n
altfel
|   a ← n
|   b ← m
□
p ← 1
k ← 0
cat timp a > 0 executa
|   k ← k*p + a%10
|   p ← p*10
|   a ← [a/10]
□
cat timp b > 0 executa
|   k ← k*p + b%10
|   p ← p*10
|   b ← [b/10]
□
scrie k
```

- a) (4p.) Scrieți valoarea afișată la ieșire, dacă de la intrare se citesc numerele 735 și 121
- b) (6p.) Dacă primul număr citit este 322, cât trebuie să fie al doilea număr citit, așa încât algoritmul să afișeze 321322?
- c) (6p.) Rescrieți algoritmul, folosind în loc de instrucțiuni repetitive cu test initial, instrucțiuni repetitive cu test final.
- d) (10p.) Scrieți programul în limbajul C / C++ corespunzător algoritmului dat.

#### II. SUBIECTUL al II-lea (30 puncte)

1. (6p.) Variabila  $n$  este de tip întreg și poate memora valori din intervalul  $[1, 99]$ . Care este valoarea minimă și valoarea maximă, pe care o poate avea expresia  $\text{abs}(n/5 + n\%5)$ ?

- a) 0; 98
- b) 1; 99

- c) 1; 23  
d) 4; 44

**2. (10p.)** Se consideră că un tablou de maxim 100 de valori de tip `int` este memorat în variabila  $v$ , iar dimensiunea efectivă a acestuia este memorată în variabila  $n$  ( $n < 100$ ). Să se scrie o secvență de instrucțiuni C, în urma executării căreia, valorile din tabloul  $v$  să fie în ordine inversă. De exemplu, dacă la început,  $v$  conținea valorile  $[2, 7, 9, 1, 4, 3, 5]$ , după execuția secvenței, el trebuie să conțină valorile  $[5, 3, 4, 1, 9, 7, 2]$ .

**3. (14p.)** Dându-se un număr întreg pozitiv  $n$ , să se determine cifrele sale în baza de numerație 2.

a) **(10p.)** Scrieți algoritmul de rezolvare al problemei în pseudocod.

b) **(4p.)** Descrieți rolul tuturor variabilelor din algoritm și indicați datele de intrare și pe cele de ieșire.

### III. SUBIECTUL al III-lea (30 puncte)

**1. (5p.)** Se consideră șirul  $A = (8, 7, 3, 5)$ . Scrieți succesiunea etapelor prin care trece șirul  $A$ , în cazul sortării crescătoare folosind metoda selecției.

**2. (5p.)** În secvența următoare, toate variabilele sunt întregi.

```
...
while (...) {
    scanf("%d", &x);
    ...
}
printf("max=%d, min=%d\n", max, min);
```

Rescrieți secvența anterioară, înlocuind punctele de suspensie, astfel încât în urma executării secvenței, după citirea unei secvențe de numere pozitive terminate cu zero, să se afișeze la terminalul standard de ieșire valoarea maximă și valoarea minimă a șirului de numere (valoarea zero nu se consideră printre elementele șirului).

**3. (10p.)** Dându-se un număr întreg  $n$ , precum și un tablou unidimensional de  $n$  valori întregi, să se scrie un program C, care realizează următoarele operații:

- crează două tablouri de numere întregi, unul conținând numerele pare din tabloul inițial, iar celălalt numerele impare;
- afișează la ieșirea standard cele două tablouri, fiecare pe o linie separată, numerele fiind separate de câte un spațiu;
- înlocuiește în primul tablou, toate numerele mai mari decât maximumul valorilor impare cu zero, iar în al doilea tablou toate numerele mai mari decât maximumul valorilor pare cu zero;
- afișează pe o linie nouă, suma elementelor din cele două tablouri, după ultima operație, separate printr-un spațiu.

**4. (10p.)** Se consideră șirurile  $a_n$  și  $b_n$  definite astfel:

$$a_n = \begin{cases} 0, n = 1 \\ 1, n = 2 \\ b_{n-1} + b_{n-2}, n > 2 \end{cases} \quad b_n = \begin{cases} 1, n = 1 \\ 2, n = 2 \\ a_{n-1} + a_{n-2}, n > 2 \end{cases}$$

Să se scrie un program C, care citește de la terminalul standard de intrare o valoare întreagă pozitivă pentru  $n$  și afișează la terminalul standard de ieșire valorile termenilor  $a_n$  și  $b_n$ .

# Exemen de admitere sesiunea iunie 2015

## Informatică

### MODEL 3

Toate subiectele sunt obligatorii.

Se acordă 10 puncte din oficiu.

Timpul efectiv de lucru este de 3 ore.

#### I. SUBIECTUL I (30 puncte)

1. (4p.) Care este valoarea expresiei  $2/1*3/2*4/3*5/4$ ?

- a) 1
- b) 2
- c) 4
- d) 5

2. (26p.) Se consideră algoritmul următor, unde  $x\%y$  reprezintă restul împărțirii numerelor naturale  $x$  și  $y$ , iar  $[x]$  reprezintă partea întreagă a numărului real  $x$ .

```
citeste n
pentru k ← 2, n executa
|   citeste m
|   s ← 0
|   cat timp m > 0 executa
|   |   s ← s + m%k
|   |   m ← [m/k]
|   □
|   daca s = k atunci
|   |   scrie k
|   □
□
```

a) (4p.) Scrieți valorile afișate la ieșire, dacă de la intrare se citește întâi numărul 5, iar apoi succesiv numerele 3, 11, 2000, 4000. Se consideră că fiecare operație de scriere afișează valorile pe o linie nouă.

b) (6p.) Se consideră că se citește întâi numărul 4, iar apoi cele 3 numere din mulțimea  $M = \{5, 10, 3000\}$ . Scrieți în ce ordine trebuie citite numerele din mulțimea  $M$ , astfel încât algoritmul să efectueze un număr maxim de afișări. Justificați răspunsul.

c) (6p.) Rescrieți algoritmul, folosind în loc de instrucțiunea repetitivă cu test initial, o instrucțiune repetitivă cu test final.

d) (10p.) Scrieți programul în limbajul C / C++ corespunzător algoritmului dat.

#### II. SUBIECTUL al II-lea (30 puncte)

1. (6p.) Se consideră variabilele  $x$  și  $y$  de tip `double`. Care dintre expresiile C de mai jos reprezintă o transcriere corectă a expresiei algebrice  $\sqrt[4]{x^3 y^5}$ ?

- a) `pow(pow(x,3)*pow(y,5), 1/4)`
- b) `pow(pow(x,3)*pow(y,5), 0.25)`
- c) `pow(pow(x,3)*pow(y,5), 1.0/4)`
- d) `pow(x,0.75)*pow(y,1.25)`

2. (10p.) Se consideră un număr întreg strict pozitiv memorat în variabila  $n$  de tip `int`, a cărui valoare poate fi maximum 39. Să se declare o variabilă  $r$ , care să memoreze un șir de caractere și să se scrie o secvență de instrucțiuni C, în urma executării căreia, variabila  $r$  să memoreze cifrele valorii lui  $n$  în scriere latină. De exemplu, dacă  $n$  are valoarea 38, atunci  $r$  trebuie să aibă valoarea "XXXVIII".

**3. (14p.)** Dându-se un număr întreg pozitiv  $n$ , care nu conține nici o cifră zero, să se determine un alt număr întreg pozitiv  $m$ , fiecare cifră a lui  $m$  fiind complementul față de 10 al cifrei corespunzătoare din numărul  $n$ . Complementul față de 10 al unei cifre nenule  $c$  este cifra  $c_1 = 10 - c$ . De exemplu, complementul lui 9 este 1, al lui 8 este 2, etc. Astfel, dacă  $n$  este, de exemplu 257, atunci numărul  $m$  va fi 853.

**a) (10p.)** Scrieți algoritmul de rezolvare al problemei în pseudocod.

**b) (4p.)** Descrieți rolul tuturor variabilelor din algoritm și indicați datele de intrare și pe cele de ieșire.

### III. SUBIECTUL al III-lea (30 puncte)

**1. (5p.)** Se consideră șirurile  $A = (1, 3, 6, 30)$  și  $B = (5, 7, 19, 33, 71)$ . Notând cu  $C$  șirul rezultat prin interclasarea lui  $A$  și  $B$ , scrieți succesiunea etapelor prin care trece șirul  $C$ , considerând că inițial  $C$  este șirul vid,  $C = ()$ .

**2. (5p.)** În secvența următoare, toate variabilele sunt întregi.

```
gasit = 0;
scanf("%d", &n);
...
while (...) {
    ...
}
if(gasit) {
    printf("Numarul %d contine cifra 9\n", n);
}
```

Rescrieți secvența anterioară, înlocuind punctele de suspensie, astfel încât în urma executării secvenței, să se afișeze mesajul respectiv, în cazul în care numărul  $n$  are valoarea 275913.

**3. (10p.)** Dându-se două mulțimi de numere întregi,  $A = \{a_1, \dots, a_n\}$  și  $B = \{b_1, \dots, b_m\}$ , să se determine mulțimea diferență,  $C = A - B$ . Să se scrie un program  $C$  care citește de la terminalul standard de intrare numărul de elemente ale celor două mulțimi, precum și elementele acestora, calculează mulțimea diferență și o afișează la terminalul standard de ieșire.

**4. (10p.)** Se consideră două funcții reale de variabilă reală,  $f$  și  $g$ . Să se determine valorile funcției compuse  $h(x) = f(g(x))$  într-un interval dat  $[a, b]$ , în puncte echidistante aflate la distanța  $dx$ , începând cu punctul  $a$ . Se consideră că  $dx < (b-a)/10$ .

$$f(x) = \begin{cases} x-1, & x < 1 \\ x+1, & x \geq 1 \end{cases} \quad g(x) = \begin{cases} -x^2, & x < -1 \\ 1+x^2, & x \geq -1 \end{cases}$$

Să se scrie un program  $C$ , care citește de la terminalul standard de intrare valorile reale  $a$ ,  $b$  și  $dx$  și afișează la terminalul standard de ieșire valorile funcției  $h$  pentru valori crescătoare ale variabilei  $x$ , conforme cu specificațiile problemei. Valorile funcției sunt separate prin spații.

**Exemen de admitere sesiunea iunie 2014**  
**Informatică**  
**MODEL 4**

**Toate subiectele sunt obligatorii.**  
**Se acordă 10 puncte din oficiu.**  
**Timpul efectiv de lucru este de 3 ore.**

**SUBIECTUL I (30 de puncte)**

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

**1. (4p.)** Indicați expresia care are valoarea 1 dacă și numai dacă numărul natural memorat în variabila întreagă  $x$  are exact trei cifre și este par.

- a)  $x / 1000 == 0 \ \&\& \ x > 99 \ \&\& \ x \% 2 == 0$       b)  $x / 1000 == 0 \ || \ x < 1000 \ || \ x \% 2 == 0$   
c)  $x \% 1000 == 0 \ \&\& \ x < 1000 \ \&\& \ x / 2 == 0$       d)  $x \% 1000 == 0 \ || \ x > 99 \ || \ x / 2 == 0$

**2. (26p.)** Se consideră algoritmul alăturat, prezentat în pseudocod

```
citește n, k
| (numere naturale nenule,  $n \geq k$ )
|_#
m <- 0
pentru (i=1, i <= n, i++)
|   citește ch
|   m = m + 1
|   x[m] = ch - '0'
|   atâta timp cat ( x[m] > x[m-1] și m > 1 și k != 0 )
|   execută
|       |   x[m-1] = x[m]
|       |   k = k - 1
|       |   m = m - 1
|       |_#
|   |_#
|   pentru (i=1, i <= m-k, i++)
|       scrie x[i]
|   |_#
```

- a) (6p.) Scrieți ce se afișează dacă pentru variabila  $n$  se citește valoarea 7, pentru  $k$  se citește valoarea 3 iar pentru variabila  $ch$  se citesc valorile 1, 2, 7, 6, 1, 5, 4.  
b) (10p) Dacă pentru variabila  $n$  se citește valoarea 10, pentru  $k$  se citește valoarea 5 ce valori pot fi date variabilei  $ch$  pentru a se obține rezultatul 12983. Dar pentru a obține 98123?  
c) (10p) Scrieți programul C/C++ corespunzător algoritmului dat.

**SUBIECTUL al II-lea (30 de puncte)**

**Pentru fiecare dintre itemii 1, 2, 3 și 4 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

**1. (4p.)** Variabila  $a$  memorează elementele unui tablou unidimensional cu 100 de elemente, numerotate de la 0 la 99. În limbajul C/C++ un element aflat pe poziția 25 poate fi accesat prin:

- a)  $a(25)$                       b)  $\&a[25]$                       c)  $a[25]$                       d)  $*a[25]$

**2 (4p.)** In C/C++, transmiterea ca parametru al unei funcții a unui tablou de elemente are ca efect transmiterea:

- a) valorilor elementelor din tablou
- b) valorii primului element din tablou
- c) adresei tabloului
- d) adresei ultimului element din tablou

**3. (4p.)** Este vreo diferență între următoarele declarații ale funcțiilor în ceea ce privește semnătura lor?

```
int welcome(int arr[]);  
int welcome(int arr[2]);
```

- a) Da
- b) Nu

**4. (4p.)** Sunt expresiile **arr** și **&arr** aceleași pentru un tablou cu 10 elemente întregi?

- a) Da
- b) Nu

**5. (14p.)** Se consideră un fișier text care conține doar litere mari și mici, cifre, spații, precum și simbolurile . , : - ? ! ( ). Scrieți un program C/C++ care deschide fișierul de intrare *input.txt* în care se află pe prima linie textul dat și tipăriți în fișierul *output.txt* un singur întreg, reprezentând partea întreagă a lungimii medii a cuvintelor textului. Se considera ca un cuvânt se termină cu spațiu, unul dintre caracterele specificate anterior, sau caracterul de sfârșit de linie \n.

*Exemplu:*

**input.txt**

- Du-te acasa, ca ca nu mai am nici o treaba cu tine, ai inteles?

**output.txt**

3

*Explicație:*

Lungimea totală a cuvintelor textului = 46, numărul de cuvinte = 15,  $[46 / 15] = 3$ .

46 litere

## SUBIECTUL al III-lea (30 de puncte)

**1. (15p.)** Se consideră subprogramul definit mai jos:

```
int fp[30005];  
for(int i = 0; i < n; ++i) {  
    d = 2;  
    while(x[n] % d == 0) {  
        fp[d] ++;  
        x[n] /= d;  
    }  
    d = 3;  
    while(x[n] > 0 && d <= x[n]) {  
        while(x[n] % d == 0) {  
            fp[d] ++;  
            x[n] /= d;  
        }  
        d += 2;  
    }  
}  
for(int i = 0; i < fp.size(); ++i) {  
    if(fp[i] != 0) {  
        printf("%d %d\n", i, fp[i]);  
    }  
}
```



}

a) (7p.) Ce afișează apelul funcției *calcul()* cu parametrii  $n = 4$  și  $x = \{32, 81, 100, 19\}$ ?

b) (8p.) Scrieți un program C/C++ care citește o variabilă întreagă  $m$  care poate lua valorile 2, 3 sau 4 și care determină dacă valoarea  $\sqrt[m]{x_1 * x_2 * \dots x_n}$  este întreagă.

2. (15p.) Se numește *prefix* al unui număr natural format din  $n$  cifre, un alt număr natural format din primele  $k$  cifre ale numărului inițial,  $1 \leq k \leq n$ . Un număr format din  $n$  cifre are  $n$  prefixe. De exemplu, prefixele numărului 2714 sunt: 2, 27, 271 și 2714. Un număr se numește *superprim*, dacă toate prefixele sale sunt numere prime. De exemplu, numărul 179 este număr superprim, deoarece numerele toate numerele 1, 17 și 179 sunt numere prime. Dându-se un număr natural  $m$ , să se determine toate numerele superprime mai mici sau egale cu  $m$ . Să se scrie un program, care citește de la terminalul standard de intrare o valoare întreagă pozitivă pentru variabila întreagă  $m$  și afișează la terminalul standard de ieșire toate numerele superprime mai mici sau egale cu  $m$ , fiecare pe o linie nouă. În cazul în care nu sunt asemenea numere, se va afișa mesajul “Nu sunt numere superprime”.

# Exemen de admitere sesiunea iunie 2014

## Informatică

### MODEL 5

Toate subiectele sunt obligatorii.

Se acordă 10 puncte din oficiu.

Timpul efectiv de lucru este de 3 ore.

#### SUBIECTUL I (30 puncte)

##### 1. (4p.) Expresia

$$((2*a + b) \% 2) \&\& ((b+1) \% 2)$$

este egală cu zero, dacă și numai dacă:

- a)  $a$  și  $b$  sunt ambele pare
- b)  $a$  este par și  $b$  este impar
- c)  $a$  este impar și  $b$  este par
- d) oricare ar fi  $a$  și  $b$

##### 2. (26p.) Se consideră algoritmul alăturat, reprezentat în pseudocod.

```
citeste n
suma ← 0
cat timp n! = 0 executa
    suma ← suma + n % 10
    n ← n / 10
scrie suma
```

a) (4p.) Scrieti numărul afișat dacă pentru variabila  $n$  se citește valoarea 475.

b) (6p.) Scrieti două numere distincte ce pot fi citite pentru variabila  $n$  astfel încât, în urma executării algoritmului, pentru fiecare dintre cele două numere, rezultatul afișat să fie 22.

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

d) (10p.) Scrieti programul C/C++ corespunzător algoritmului dat.

#### SUBIECTUL al II-lea (30 puncte)

##### 1. (4p.) Se considera urmatoarea secventa repetitiva:

```
i=0;
while (i+j<=10) {
    i=i+1;
    j=j-2;
}
```

Valoarea minima posibila pentru care variabila  $j$  astfel incat instructiunea repetitiva de mai sus sa nu se execute la infinit (necontrolat) este:

- a) 1      b) 5      c) 6      d) 17      e) 10      f) 2

2. (6p.) Variabile  $a$  și  $b$  sunt de tip întreg și memorează numere naturale nenule. Expresia care poate înlocui punctele de suspensie astfel încât la finalul executării secvenței obținute variabila  $r$  să memoreze cel mai mic multiplu comun al valorilor memorate inițial în variabilele  $a$  și  $b$  este:

```
r=a%b;
while(r!=0) {
```

```

    a=b;
    b=r;
    r=a % b;
}
... . ;

```

- a)  $r=(a*b)/r$
- b)  $r=a*b*r$
- c)  $r=(a+b)*r$
- d)  $r=(a+b)/r$

**3. (4p.)** Se dau variabilele (pointeri la caractere):  $x=$  “Ana”,  $y=$ “are”,  $z=$ “multe” si  $t=$ “mere”. Scrieti o singura linie de cod astfel incat, folosind cele 4 variabile, pe ecran sa apara:

*Ana are multe  
mere*

**4. (14p.)** Se da urmatoarea problema: daca se citesc doua numere de la tastatura,  $a$  si  $b$ , cu  $a \leq b$ , sa se calculeze suma produselor numerelor plecand, pe de o parte de la 1 si mergand pana la  $a$ , iar pe de cealalta parte plecand de la  $b$  si scazand 1 de fiecare data. (de exemplu,  $a=3$  si  $b=9$ , trebuie calculata suma produselor  $1*9, 2*8, 3*7$ ).

- a) (10p.) Scrieti, în pseudocod, algoritmul de rezolvare pentru problema enuntată.
- b) (4p.) Mentionati rolul tuturor variabilelor care au intervenit în algoritmul realizat la punctul a) și indicati datele de intrare, respectiv datele de ieșire ale problemei enuntate.

### SUBIECTUL al III-lea (30 puncte)

**1. (4p.)** Se da secventa de cod:

```

for i ← 1, n – 1 do
    for j ← n, i+1, –1do
        if ( $a_j < a_{j-1}$ ) then
             $a_{j-1} \leftrightarrow a_j$ 

```

Unde instructiunea  $a_{j-1} \leftrightarrow a_j$  inseamna interschimbarea valorilor  $a_{j-1}$  si  $a_j$ .

Care vor fi valorile vectorului  $a$  dupa terminarea pasului  $i = 5$ , stiind ca la intrare avem valorile  $n = 7$  și  $a = (7,5,3,1,2,4,6)$ ?

- a) (1,2,3,7,6,5,4)
- b) (1,2,3,4,5,6,7)
- c) (1,2,3,4,7,6,5)
- d) (1,2,3,4,5,7,6)
- e) (1,5,3,2,4,6,7)

**2. (6p.)** Structura unui fisier text este urmatoarea:

*Numele utilizatorului  
Numarul de numere din fisier  
Primul numar  
Al doilea numar  
...  
Ultimul numar*

Cititi datele din fisier si afisati pe ecran numerele in ordinea inversa din fisier si apoi pe randul urmator numele utilizatorului.

**3. (10p.)** Scrieti un program C/C++ care citeste de la tastatura un numar natural  $n$  cuprins intre 4 si 100 si cele  $n$  numere ale unui sir. Fiecare numar trebuie sa aiba minim 2 cifre. Creati un nou sir care contine inversele numerelor citite, apoi, pentru fiecare numar din sirul initial, daca acesta este mai mic decat inversul sau, adaugati in sirul initial dupa acest numar si inversul acestuia. Afisati pe ecran sirul ce contine numerele initiale la care s-au adaugat aceste inverse.

**4. (10p.)** Fie data o pereche de iepuri. Se stie ca fiecare pereche de iepuri produce in fiecare luna o noua pereche de iepuri, care la randul sau devine productiva la varsta de o luna. Sa se scrie un program C/C++ care afiseaza cate perechi de iepuri vor fi dupa  $n$  luni (unde  $n$  se citeste de la tastatura) si sa se explice algoritmul folosit.

# Exemen de admitere sesiunea iunie 2014

## Informatică

### MODEL 6

**Toate subiectele sunt obligatorii.**

**Se acordă 10 puncte din oficiu.**

**Timpul efectiv de lucru este de 3 ore.**

#### SUBIECTUL I (30 puncte)

**Marcați varianta corectă (sau variantele corecte).**

**1. (5p.)** Care este rezultatul evaluării expresiei

$a < b < c$

dacă  $a=-2$ ,  $b=-1$  și  $c=0$ ? Dar dacă  $a=0$ ,  $b=1$ ,  $c=2$ ?

- a) 1 în ambele cazuri
- b) 0 în ambele cazuri
- c) 0, respectiv 1
- d) 1, respectiv 0.

**2. (5p.)** Presupunem că valorile expresiilor booleene “ $A > B$ ” și “ $B > C$ ” sunt independente și că, în medie,  $A > B$  în 75% din cazuri și  $B > C$  în 25% din cazuri. Se dă codul (fragmentul) de program de mai jos, unde  $V$  este un vector și  $f$ ,  $g$  sunt funcții:

```
if (A>B) V[i] = f(i)
else if (B>C) V[i] = g(i);
```

De câte ori se execută  $f$  și  $g$ ?

- a) De 2500 ori  $f$  și de 1875 ori  $g$ .
- b) De 7500 ori  $f$  și de 625 ori  $g$ .
- c) De 7500 ori  $f$  și de 1875 ori  $g$ .
- d) De 7500 ori  $f$  și de 2500 ori  $g$ .

**3. (5p.)** Următoarea secvență de program:

```
int bc=188;
printf("bc=b%x\n", bc);
```

va tipări:

- a)  $bc=188$
- b)  $bc=b188$
- c)  $bc=bbc$
- d) altceva

**4. (5p.)** Ce număr va fi afișat pe ecran după execuția următoarelor instrucțiuni?

```
char ch;
int i;
ch='G';
i = ch - 'A';
printf("Number: %d\n", i);
```

- a) 5
- b) 6
- c) 7
- d) 8

**5. (5p.)** Se consideră declarațiile:

```
int a[5], i;
```

După execuția instrucțiunilor de mai jos:

```
for(i=0; i<5; i++) a[i]=i;  
for(i=0; i<4; i++) a[i]=a[i%5+1];
```

ce valori vor avea elementele lui a?

- a) 0,1,2,3,4
- b) 1,2,3,4,0
- c) 0,1,2,4,4
- d) 1,2,3,4,4

6. (5p.) Se dau următoarele declarații, asemănătoare din punct de vedere sintactic, dar și formal:

I. `char s[20];`  
`s="Universitate";`

II. `char s[20]=" Universitate ";`

Sunt acestea pe deplin corecte?

- a) doar secvența I este corectă
- b) doar secvența II este corectă
- c) ambele secvențe sunt corecte
- d) nici una dintre secvențe nu este corectă

### SUBIECTUL al II-lea (30 puncte)

1. (30p.) Se consideră două șiruri de numere reale, de lungimi  $n$ , respectiv  $m$ :  $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m$ . Să se scrie un program C / C++, care să realizeze următoarele operații

- a) Citește cele două șiruri de la terminalul standard de intrare.
- b) Determină dacă șirul al doilea este un subșir al primului șir (este format din elemente consecutive ale primului șir).
- c) Ordonează crescător elementele celor două șiruri.
- d) Crează un al treilea șir,  $z_1, z_2, \dots, z_k$ , ordonat crescător, care conține elementele din primele două șiruri, cu condiția ca valorile distincte să apară o singură dată.
- e) Afișează șirurile inițiale, cele sortate, preum și șirul creat la punctul d).

### SUBIECTUL al III-lea (30 puncte)

(30p.) Se consideră două polinoame cu coeficienți reali,  $P$  și  $Q$ , având gradele  $n$  și  $m$ , precum și coeficienții acestora date de șirurile  $a_n, a_{n-1}, \dots, a_1, a_0$  respectiv  $b_m, b_{m-1}, \dots, b_1, b_0$ . Să se scrie un program C / C++, care să realizeze următoarele operații:

- 1. (4p.) Declară variabilele asociate celor două polinoame și citește de la tastatură valori pentru cele două polinoame (gradele lor și vectorii coeficienților). În plus, declară variabilele reale  $x, y$  și  $a$ .
- 2. (7p.) Citește valori pentru  $x$  și  $y$  și determină valorile  $P(x)$  și  $Q(y)$ .
- 3. (7p.) Citește valori pentru  $x$  și  $y$ , să determină dacă polinomul  $P$  își schimbă semnul între  $x$  și  $y$ .
- 4. (4p.) Citește o valoare pentru  $a$ , și determină restul împărțirii lui  $P$  la  $x-a$ .
- 5. (8p.) Determină polinomul produs  $R=P \times Q$ .