



**SUBIECTUL al II-lea**

**(40 de puncte)**

**1. Algoritmul alăturat este reprezentat în pseudocod.**

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

a. Scrieți numărul afișat în urma executării algoritmului dacă se citesc, în această ordine, numerele 3845267 și 3. (6p.)

b. Dacă pentru variabila  $k$  se citește 2, scrieți trei numere din intervalul  $[10^3, 10^4]$  care pot fi citite pentru  $n$  astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze 20. (6p.)

c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)

d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat structura **repetă...până când** cu o structură repetitivă cu test inițial. (6p.)

citește  $n, k$   
(numere naturale)  
 $p \leftarrow 1$   
dacă  $k=0$  atunci  
     $nr \leftarrow -1$   
altfel  
     $nr \leftarrow 0$   
repetă  
     $c \leftarrow n \% 2$ ;  $n \leftarrow [n/10]$   
    dacă  $c \neq 0$  atunci  
         $nr \leftarrow nr + (n \% 10) * p$ ;  $p \leftarrow p * 10$   
    altfel  
         $k \leftarrow k - 1$   
    până când  $n=0$  sau  $k=0$   
scrie  $nr$

2. Utilizând metoda backtracking se generează toate numerele din intervalul  $[10^4, 10^5]$ , cu cifrele în ordine strict crescătoare. Primele trei numere generate sunt 12345, 12346, 12347. Scrieți, în ordinea obținerii, ultimele trei numere generate care au prima cifră 4. (6p.)

3. Variabila  $s$  memorează un șir cu cel mult 20 de caractere, iar celelalte variabile sunt de tip întreg. Scrieți ce se afișează pe ecran în urma executării secvenței alăturate. (6p.)

0AIAE

$k = 'a' - 'A';$   
 $strcpy(s, "A realizat tot");$   
 $for(i = strlen(s) - 1; i \geq 0; i--)$   
    if( $strchr("aeiou", s[i]) \neq NULL$ )  
        {  $s[i] = s[i] - k$ ;  
           $cout \ll s[i]$ ; }  
    printf("%c", s[i]);

**SUBIECTUL al III-lea**

**(30 de puncte)**

**1. Subprogramul prime are trei parametri:**

- $n$ , prin care primește un număr natural ( $n \in [4, 10^9]$ );
- $x$  și  $y$ , prin care furnizează cele mai mari două numere prime din intervalul  $[1, n]$ ,  $x < y$ .

Scrieți definiția completă a subprogramului.

**Exemplu:** dacă  $n=49$ , în urma apelului  $x=43$ ,  $y=47$ .

(10p.)

2. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul  $[2, 10^2]$ ,  $m$  și  $n$ , și construiește în memorie un tablou bidimensional cu  $m$  linii și  $n$  coloane, cu proprietatea că parcurgându-l linie cu linie de sus în jos și fiecare linie de la stânga la dreapta, se obține șirul primelor  $m \cdot n$  pătrate perfecte pare, ordonat strict descrescător, ca în exemplu.

Elementele tabloului obținut se afișează pe ecran, fiecare linie a tabloului pe câte o linie a ecranului, valorile de pe aceeași linie fiind separate prin câte un spațiu.

**Exemplu:** pentru  $m=2$ ,  $n=3$  se obține tabloul alăturat.

(10p.)

100 64 36  
16 4 0

3. Fișierul **cheltuieli.in** are cel mult  $10^6$  linii, fiecare linie conținând câte trei numere naturale din intervalul  $[1, 10^2]$ , reprezentând, în această ordine, date despre câte o achiziție: tipul produsului cumpărat, numărul de produse de acest tip cumpărate, respectiv prețul unui astfel de produs la acel moment. Numerele aflate pe aceeași linie sunt separate prin câte un spațiu.

Se cere să se afișeze pe ecran cea mai mare sumă cheltuită pentru toate produsele de același tip, precum și numărul de tipuri de produse pentru care s-a obținut această sumă. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

**Exemplu:** dacă fișierul **cheltuieli.in** are conținutul alăturat, se afișează pe ecran: 26 2 (s-a cheltuit suma maximă 26 pentru produsele de tipul 1 și 4:  $26 = 16 \cdot 1 + 5 \cdot 2 = 1 \cdot 10 + 2 \cdot 8$ )

4 1 10  
1 16 1  
4 2 8  
2 1 5  
1 5 2

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia.

(2p.)

b. Scrieți programul C/C++ corespunzător algoritmului proiectat.

(8p.)

**Examenul național de bacalaureat 2021**

**Proba E. d)  
INFORMATICĂ**

**BAREM DE EVALUARE ȘI DE NOTARE  
(comun pentru limbajele C/C++ și Pascal)**

**Model**

*Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică*  
*Filieră vocațională, profil militar, specializare matematică-informatică*

- Se punctează orice formulare/modalitate de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit în barem. Nu se acordă fracțiuni de punct.
- Se acordă zece puncte din oficiu. Nota finală se calculează prin împărțirea punctajului total acordat pentru lucrare la zece.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

**SUBIECTUL I (20 de puncte)**

1c 2a 3d 4b 5c	5x4p.
----------------	-------

**SUBIECTUL al II - lea (40 de puncte)**

1.	<b>a. Răspuns corect: 46</b>	<b>6p.</b>	
	<b>b. Pentru răspuns corect</b>	<b>6p.</b>	Se acordă câte 2p. pentru fiecare dintre cele trei numere conform cerinței (numere de forma $2 \times 0y$ unde $x, y$ sunt cifre impare).
	<b>c. Pentru program corect</b> -declarare variabile -citire date -afișare date -instrucțiuni de decizie (*) -instrucțiune repetitivă -atribuiri -corectitudine globală a programului <sup>1)</sup>	<b>10p.</b> 1p. 1p. 1p. 3p. 2p. 1p. 1p.	(*) Se acordă numai 2p. dacă doar una dintre instrucțiunile de decizie este conform cerinței.
	<b>d. Pentru algoritm pseudocod corect</b> -echivalență a prelucrării realizate, conform cerinței (*) -corectitudine globală a algoritmului <sup>1)</sup>	<b>6p.</b> 5p. 1p.	(*) Se acordă numai 2p. dacă algoritmul are o structură repetitivă conform cerinței, principial corectă, dar nu este echivalent cu cel dat. Se va puncta orice formă corectă de structură repetitivă conform cerinței.
2.	<b>Răspuns corect:</b> <b>45689, 45789, 46789</b>	<b>6p.</b>	Se acordă câte 2p. pentru fiecare număr conform cerinței.
3.	<b>Răspuns corect:</b> <b>OAIAE</b>	<b>6p.</b>	Se acordă câte 2p. pentru fiecare aspect specific (litere mari, vocale, litere suport) conform cerinței.

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

1.	<b>Pentru subprogram corect</b> -antet subprogram (*) -determinare a valorilor cerute (**) -declarare a tuturor variabilelor locale, corectitudine globală a subprogramului <sup>1)</sup>	<b>10p.</b> 3p. 6p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect al antetului (structură, parametru de intrare, parametri de ieșire) conform cerinței. (**) Se acordă câte 2p. pentru fiecare aspect specific (algoritm principial corect de determinare a unui număr prim, determinare a celui mai mare număr prim din interval, determinare a celui de al doilea număr prim) conform cerinței.
2.	<b>Pentru program corect</b> -declarare a unei variabile care să memoreze un tablou bidimensional -citire a datelor -construire a tabloului cerut (*) -afișare a datelor în formatul cerut -declarare a variabilelor simple, corectitudine globală a programului <sup>1)</sup>	<b>10p.</b> 1p. 1p. 6p. 1p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect specific (numere pare, numere pătrate perfecte, ordonare valori pe linii, ordonare valori pe coloane, valori suport, construire în memorie) conform cerinței.

<b>3.</b>	<p><b>a. Pentru răspuns corect</b> -coerență a descrierii algoritmului (*) -justificare a elementelor de eficiență</p> <p><b>b. Pentru program corect</b> -operații cu fișiere: declarare, pregătire în vederea citirii, citire din fișier -determinare a valorilor cerute (*),(**) -utilizare a unui algoritm eficient (***) -declarare a variabilelor, afișare a datelor, corectitudine globală a programului<sup>1)</sup></p>	<p><b>2p.</b> 1p. 1p. <b>8p.</b> 1p. 5p. 1p. 1p.</p>	<p>(*) Se acordă punctajul chiar dacă algoritmul ales nu este eficient. (**) Se acordă numai 3p. dacă algoritmul este principial corect, dar nu oferă rezultatul cerut pentru toate seturile de date de intrare. (***) Se acordă punctajul numai pentru un algoritm liniar. O soluție posibilă utilizează un tablou, s, în care s[i]=suma cheltuită în total pentru produsele de tipul i. Pe măsura citirii datelor din fișier, pentru fiecare triplet x, y, z citit se actualizează s[x] prin adunare la valoarea sa actuală a produsului y*z. La final se determină valoarea maximă memorată în tabloul s, precum și numărul de apariții ale acestei valori.</p>
-----------	--	--	--

<sup>1)</sup> Corectitudinea globală vizează structura, sintaxa, alte aspecte neprecizate în barem.