

**Examenul de bacalaureat național 2017**  
**Proba E. d)**  
**Informatică**  
**Limbajul C/C++**

**Varianța 5**

**Filiera teoretică, profilul real, specializările: matematică-informatică**

**matematică-informatică intensiv informatică**

**Filiera vocațională, profilul militar, specializarea matematică-informatică**

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.

**SUBIECTUL I** (30 de puncte)

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

1. Variabila **x** este de tip întreg. Numărul de **valori întregi distincte** ale lui **x** pentru care expresia C/C++ alăturată are valoarea 0 este: (4p.) **x/2017**

**a. 2016                      b. 2017                      c. 2·2016+1                      d. 2·2017+1**

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.

S-a notat cu **a%b** restul împărțirii numărului natural **a** la numărul natural nenul **b**, cu **[a]** partea întreagă a numărului real **a**.

- a) Scrieți ce se afișează dacă se citesc, în această ordine, numerele 11 și 16. (6p.)**
- b) Scrieți un set de date care pot fi citite astfel încât, în urma executării algoritmului, să se afișeze valoarea 10. (4p.)**

```
citește a,b
( numere naturale nenule, a≤b )
s←0
pentru x←a,b execută
| pentru j←1,[x/2] execută
| | dacă x%j=0 și [√j]*[√j]=j atunci
| | | s←s+j
| |
|
scrie s
```

- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat prima structură pentru...execută cu o structură repetitivă de alt tip. (6p.)**
- d) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)**

**SUBIECTUL al II-lea**

**(30 de puncte)**

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

1. Variabila `s` permite accesarea unui șir de maximum 49 de caractere. Indicați expresia egală cu 1 dacă și numai dacă șirul `2017` coincide cu subșirul format din primele patru caractere ale șirului accesat prin variabila `s` și **NU** mai apare pe alte poziții în acesta. **(4p.)**

- a. `strstr(s,"2017")==0 && strstr(s+4,"2017")==s+4`  
b. `strstr(s,"2017")!=s || strstr(s+4,"2017")!=s+4`  
c. `strstr(s,"2017")!=0 || strstr(s+4,"2017")!=0`  
d. `strstr(s,"2017")==s && strstr(s+4,"2017")==0`

2. Într-un arbore cu rădăcină considerăm că un nod se află pe nivelul  $x$  dacă lanțul elementar care are o extremitate în nodul respectiv și cealaltă extremitate în rădăcina arborelui are lungimea  $x$ . Pe nivelul 0 se află un singur nod (rădăcina).  
Pe fiecare nivel nenul al unui arbore cu rădăcină există cel puțin o frunză. Dacă ultimul nivel este 3, atunci numărul minim de noduri din arbore este: **(4p.)**

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

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

3. Se consideră declararea alăturată, în care variabila `t` memorează, pentru fiecare dintre cele 10 triunghiuri, lungimile laturilor.  
Scrieți o expresie C/C++ care are valoarea 1 dacă și numai dacă primul triunghi dintre cele menționate este echilateral. **(6p.)**
- ```
struct triunghi {  
    int x,y,z;  
} t[10];
```

4. Scrieți matricea de adiacență a unui graf orientat cu 5 vârfuri și 3 arce, știind că el are un număr maxim de vârfuri care au gradul intern egal cu gradul extern. **(6p.)**

5. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul  $[2, 50]$ ,  $m$  și  $n$ , și elementele unui tablou bidimensional cu  $m$  linii și  $n$  coloane, numere naturale distincte, din intervalul  $[0, 10^4]$ . Programul interschimbă valoarea minimă din ultima coloană a tabloului cu valoarea minimă din prima coloană a tabloului, apoi afișează pe ecran tabloul modificat, câte o linie a tabloului pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** dacă  $m=4$ ,  $n=3$  și tabloul este

7    5    19    atunci se obține tabloul următor:

**3**    8    4  
23    6    **1**  
10    2    9

7    5    19

**1**    8    4  
23    6    **3**  
10    2    9

**(10p.)**

**SUBIECTUL al III-lea**

**(30 de puncte)**

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

1. Utilizând metoda backtracking se generează toate modalitățile de a scrie numărul 6 ca sumă de numere naturale impare. Termenii fiecărei sume sunt în ordine crescătoare. Cele patru soluții sunt obținute în această ordine:  $1+1+1+1+1+1$ ;  $1+1+1+3$ ;  $1+5$ ;  $3+3$ .  
Aplicând același algoritm, numărul soluțiilor obținute pentru scrierea lui 8 este: **(4p.)**

a. 5                                      b. 6                                      c. 8                                      d. 9

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

2. Se consideră subprogramul `f`, definit alăturat. Scrieți ce se afișează în urma executării instrucțiunii:  
`cout<<f(10)<<' '<<f(11);` |  
`printf("%d %d",f(10),f(11));` **(6p.)**
- ```
int f (int x)
{ if (x%2==0)
    return x;
  return f(x/2);
}
```

3. Subprogramul `produs` are doi parametri:
- `a`, prin care primește un număr natural ( $a \in [1, 10^6]$ );
  - `k`, prin care furnizează cea mai mare valoare naturală impară cu proprietatea că produsul tuturor numerelor impare din intervalul  $[1, k]$  este mai mic sau egal cu `a`.
- Scrieți definiția completă a subprogramului.  
**Exemplu:** dacă `a=200`, atunci `k=7` ( $1 \cdot 3 \cdot 5 \cdot 7 \leq 200 < 1 \cdot 3 \cdot 5 \cdot 7 \cdot 9$ ). **(10p.)**

4. Fișierul `bac.txt` conține un șir de cel mult  $10^6$  numere naturale din intervalul  $[1, 10^9]$ , ordonate crescător. Numerele din șir sunt separate prin câte un spațiu.  
Se cere să se afișeze pe ecran cel mai mic număr din șir care apare în fișier de un număr impar de ori.  
Dacă în fișier nu se află o astfel de valoare, pe ecran se afișează mesajul **nu exista**.  
Proiectați un algoritm eficient din punctul de vedere al spațiului de memorie utilizat și al timpului de executare.  
**Exemplu:** dacă fișierul `bac.txt` conține numerele  
5 5 8 8 8 8 20 20 20 34 54 54 65 65 65  
atunci pe ecran se afișează 20.
- a) Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**  
b) Scrieți programul C/C++ corespunzător algoritmului descris. **(8p.)**