

EXAMENUL DE BACALAUREAT – 2008
Proba scrisă la INFORMATICĂ
PROBA E, limbajul C/C++
Specializarea Matematică-informatică intensiv informatică

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru 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).

Subiectul I (30 de puncte)

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

1. Care este valoarea expresiei **C/C++** alăturate? (4p.) $50 - (100 - 300/2 / (2+3))$
- a. -30 b. 70 c. -20 d. 60

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

2. Se consideră algoritmul alăturat descris în pseudocod.

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

- a) Scrieți numerele afișate dacă se citește valoarea **$x=168$** . (6p.)
- b) Scrieți o valoare care poate fi citită pentru variabila **x** astfel încât, în urma executării algoritmului, să se afișeze două valori egale. (4p.)
- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască fiecare structură **cât timp...execută** cu câte o structură repetitivă cu test final. (6p.)
- d) Scrieți programul **C/C++** corespunzător algoritmului dat. (10p.)

```
citește x
      (număr natural nenul)
 $d \leftarrow 2$ ;  $y \leftarrow 0$ ;  $z \leftarrow 0$ 
cât timp  $x \neq 1$  execută
     $p \leftarrow 0$ 
    cât timp  $x \% d = 0$  execută
         $p \leftarrow p + 1$ 
         $x \leftarrow [x/d]$ 
    ■
    dacă  $p < > 0$  atunci
        dacă  $y = 0$  atunci  $y \leftarrow d$ 
        ■
         $z \leftarrow d$ 
        ■
         $d \leftarrow d + 1$ 
    ■
scrie y
scrie z
```

Subiectul II (30 de puncte)

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

1. Se consideră vectorul de "tați" al unui arbore cu rădăcină $t = (3, 4, 0, 3, 3, 5)$ ale cărui noduri sunt numerotate de la 1 la 6. Alegeți afirmația corectă: **(4p.)**
- a. nodurile 4 și 6 sunt noduri de tip frunză b. nodul 3 are un singur descendent direct (fiu)
- c. nodul 6 este tatăl nodului 5 d. nodurile 1, 2, 6 sunt noduri de tip frunză
2. Se consideră o coadă, în care au fost introduse inițial, în această ordine, două numere: 2 și 1. Conținutul cozii este reprezentat în figura alăturată. Notăm cu **AD** x operația prin care se adaugă informația x în coadă și cu **EL** operația prin care se elimină un element din coadă. Asupra cozii se efectuează, exact în această ordine, operațiile **AD 10; AD 15; EL; AD 4; EL; AD 20; EL**. Care este conținutul cozii după executarea operațiilor de mai sus? **(4p.)**
- | | |
|---|---|
| 2 | 1 |
|---|---|
- a. 15 4 b. 15 4 20 c. 4 20 d. 20

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

3. Se consideră un graf neorientat cu 8 noduri, numerotate de la 1 la 8, și muchiile $[1, 5]$, $[1, 6]$, $[2, 6]$, $[3, 4]$, $[3, 6]$, $[3, 7]$, $[4, 6]$, $[6, 8]$, $[7, 8]$. Dacă se elimină nodul 6 și toate muchiile incidente cu acesta câte componente conexe va avea subgraful rezultat?**(6p.)**
4. Considerăm declarațiile:
`int i, j, a[10][10];`
Ce se va afișa după executarea secvenței de instrucțiuni alăturate? **(6p.)**
- ```
for(i=1; i<=3; i++)
 for(j=1; j<=3; j++)
 if(i<j) a[i][j]=i;
 else a[i][j]=j;
for(i=1; i<=3; i++) {
 for(j=1; j<=3; j++)
 cout<<a[i][j]; | printf("%d", a[i][j]);
 cout<<endl; | printf("\n");
}
```
5. Un șir cu maximum 255 de caractere conține cuvinte separate prin caracterul \*. Cuvintele sunt formate numai din litere mici ale alfabetului englez. Scrieți un program **C/C++** care citește un astfel de șir și afișează pe ecran șirul obținut prin eliminarea tuturor aparițiilor primului cuvânt, ca în exemplu.  
**Exemplu:** pentru șirul: **bine\*bine\*e\*foarte\*bine\*** se va afișa:  
**\*\*e\*foarte\*\*** **(10p.)**

**Subiectul III (30 de puncte)**

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

1. Utilizăm metoda backtracking pentru generarea tuturor modalităților de a scrie numărul **6** ca sumă a cel puțin două numere naturale nenule. Termenii descompunerii sunt în ordine crescătoare. Soluțiile se generează în ordinea: **1+1+1+1+1+1**, **1+1+1+1+2**, **1+1+1+3**, **1+1+4**, **1+5**, **2+2+2**, **2+4** și **3+3**. Se aplică exact aceeași metodă pentru scrierea lui **9**. Câte soluții de forma **2+...** vor fi generate? **(6p.)**
- a. 2                                      b. 3                                      c. 4                                      d. 5

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

2. Pentru definiția alăturată a subprogramului **f**, stabiliți ce valoare are **f(23461)**? **(4p.)**
- |                                                                                                                                                                              |  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|
| <pre>int f(int x) {     if (x&lt;10) {if (x%2!=0) return 0;               return x;             }     if ((x%2!=0)         return f(x/10);     return f(x/10)+x%10; } </pre> |  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|
3. Pe prima linie a fișierului **bac.in** se află un număr natural nenul **n ≤ 1000**, iar pe a doua linie a fișierului se află un șir format din **n** numere naturale, 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 mesajul **DA** dacă elementele pare în șir sunt în ordine crescătoare, iar cele impare sunt în ordine descrescătoare și mesajul **NU** în caz contrar.
- Exemplu:** dacă fișierul **bac.in** are conținutul **8**  
alăturat pe ecran se va afișa: **DA** **(10p.)**
- |  |                               |
|--|-------------------------------|
|  | 10 1133 12 331 42 1354 221 13 |
|--|-------------------------------|
4. Considerăm definite subprogramele:
- **pr** care primește prin intermediul parametrului **x** un număr natural cu cel mult **4** cifre și returnează **1** dacă numărul este prim și **0** în caz contrar;
  - **div** care primește prin intermediul parametrului **y** un număr natural cu cel mult **4** cifre și returnează suma tuturor divizorilor numărului **y**.
- a) Scrieți numai antetul subprogramelelor **pr** și **div**. **(4p.)**
- b) Scrieți un program **C/C++** care citește de la tastatură un număr natural **n (n < 1000)** și care afișează pe ecran toate numerele mai mici decât **n** cu proprietatea că suma divizorilor lor este număr prim, folosind apeluri utile la subprogramele **pr** și **div**.
- Exemplu:** dacă **n=20** atunci programul va afișa: **2 4 9 16** **(6p.)**