

EXAMENUL DE BACALAUREAT – 2007
Proba scrisă la INFORMATICĂ
PROBA E, limbajul C/C++

Varianta 38

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru este de 3 ore.
- ♦ În programele cerute la subiectele II și III, 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 (40 de puncte)

Pentru fiecare din itemii de la 1 la 8, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 5 puncte.

1. Dacă un graf neorientat are n noduri și p componente conexe atunci numărul minim de muchii care trebuie adăugate astfel încât graful să devină conex este:
 - a. p
 - b. $p-1$
 - c. $n-1$
 - d. n
2. Se consideră o lista circulară cu 8 elemente numerotate cu $1, 2, 3, 4, \dots, 8$. Mai întâi se elimină elementul numerotat cu 3, apoi se elimină fiecare al treilea element al parcurgerii, numărarea continuându-se începând cu succesorul elementului eliminat, până când lista va mai conține un singur element. Care va fi numărul de ordine al elementului rămas?
 - a. 2
 - b. 7
 - c. 3
 - d. 4
3. Se consideră mulțimile $A=\{1, 2, 3\}$, $B=\{1\}$, $C=\{2, 3, 4\}$. Elementele produsului cartezian $A \times B \times C$ se generează, folosind metoda backtracking, în ordinea $(1, 1, 2), (1, 1, 3), (1, 1, 4), (2, 1, 2), (2, 1, 3), (2, 1, 4), (3, 1, 2), (3, 1, 3), (3, 1, 4)$. Dacă prin același algoritm se generează produsul cartezian al mulțimilor $A \times B \times C$ unde $A=\{x, y\}$, $B=\{x\}$, $C=\{x, y, z\}$, atunci cel de-al treilea element generat este :
 - a. (x, x, y)
 - b. (x, y, x)
 - c. (x, x, z)
 - d. (x, y, z)
4. Într-un graf orientat cu n noduri, gradul extern al unui nod poate fi maximum:
 - a. $n-1$
 - b. 1
 - c. $n+1$
 - d. 2
5. Fie k o variabilă globală de tip întreg, n și m două numere naturale și subprogramul p alăturat. Știind că variabilei k i s-a atribuit valoarea 0 exact înaintea apelului $p(n, m)$, stabiliți care este valoarea variabilei k după încheierea executării subprogramului.

- a. m
 - b. $m-1$
 - c. $m+1$
 - d. n

```

int p(int n, int m )
{
    k++;
    if (m==0) return 0;
    else return n + p(n,m-1);
}
          
```
6. Care dintre următoarele variante inițializează variabila întreagă n cu valoarea 2? Se presupune că inițializarea unei variabile este instrucțiunea prin care se atribuie o valoare în momentul declarării.
 - a. `int n:=2;`
 - b. `int n=2;`
 - c. `int n==2;`
 - d. `int n;`
7. Cu ce expresie trebuie înlocuite punctele de suspensie astfel încât în urma executării secvenței alăturate să se deplaseze elementele v_q, v_{q+1}, \dots, v_k ale unui tablou unidimensional v cu $p-1$ poziții spre dreapta?

- a. $p-1-j$
 - b. $j-p+1$
 - c. $p-1+j$
 - d. $p-j+1$

```

for (j=k; j>=q; j--)
    v[.....]=v[j]
          
```
8. Se consideră declararea `struct fractie{unsigned int x,y;}s,f1,f2;` în care câmpurile x și y reprezintă numărătorul, respectiv numitorul unei fracții. Care dintre următoarele variante construiește în variabila s o fracție echivalentă cu suma fracțiilor $f1$ și $f2$?
 - a. `s.x=f1.x+f2.x;`
`s.y=f1.y+f2.y;`
 - b. `s.x=f1.x*f2.y+f1.y*f2.x;`
`s.y=f1.y*f2.y;`
 - c. `s=f1+f2;`
 - d. `s.x=f1.x*f1.y+f2.x*f2.y;`
`s.y=f1.y*f2.y;`

SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat, unde subprogramul `suma(n)` returnează suma cifrelor numărului natural n transmis ca parametru.

1. Care este valoarea afișată pentru $n=1999$? (5p.)
2. Dați exemplu de o valoare pentru n astfel încât valoarea afișată să fie 1. (3p.)
3. Care este cea mai mare valoare de patru cifre ce trebuie citită pentru variabila n astfel încât să se afișeze 3? (2p.)
4. Scrieți programul C/C++ corespunzător algoritmului dat, inclusiv definiția completă a subprogramului necesar. (10p.)

```

citește n ( $n \in \mathbb{N}$ )
nr ← 0
cât timp n > 9 execută
    n ← suma(n)
    nr ← nr + 1
scrie nr

```

SUBIECTUL III (30 de puncte)

1. Scrieți un program C/C++ care citește n , m și un tablou bidimensional cu n linii ($1 < n < 10$) și m coloane ($1 < m < 20$) cu elemente numere întregi, un număr natural k ($1 \leq k \leq n$) și afișează cel mai mare element de pe linia k , numerotarea liniilor începând de sus în jos, de la 1 la n . (10p.)
2. Subprogramul f primește prin intermediul parametrului n un număr natural cuprins între 2 și 10000 și returnează cel mai mare termen din șirul lui Fibonacci strict mai mic decât n .
 - a) Scrieți definiția completă a subprogramului f . (3p.)
 - b) Scrieți programul C/C++ care citește elementele unui tablou unidimensional de numere naturale mai mari decât 1 și mai mici decât 10000, apoi înlocuiește fiecare element al tabloului cu cel mai mare termen din șirul lui Fibonacci strict mai mic decât el și afișează elementele tabloului astfel obținut pe o singură linie a ecranului, separate prin spațiu. Se vor folosi apeluri utile ale subprogramului f definit la punctul a).
De exemplu, dacă se citește tabloul (20, 2, 87, 3, 120) atunci se va afișa: 13 1 55 2 89. (7p.)
3. Se consideră două fișiere text **F1.TXT** și **F2.TXT** ce conțin numere reale dispuse pe mai multe linii și separate prin spații. Fișierul **F1.TXT** conține numai numere distincte, iar fișierul **F2.TXT** conține numai numere din fișierul **F1.TXT**, dar acestea se pot repeta.
Scrieți programul C/C++ care afișează pentru fiecare valoare din fișierul **F1.TXT** numărul său de apariții în fișierul **F2.TXT**. Pe linii diferite ale ecranului vor fi afișate câte două valori, separate prin spațiu, prima fiind numărul din fișierul **F1.TXT**, iar a doua numărul său de apariții din **F2.TXT**. (10p.)