

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

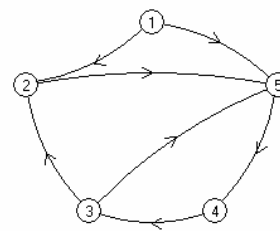
Varianta 83

- ◆ 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. Se consideră graful orientat din figura alăturată. Câte circuite elementare disticte are graful?



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

2. Se generează toate numerele de 5 cifre, cu cifre distincte, care pe poziții pare au cifre pare, iar pe poziții impare au cifre impare. Primele șase numere generate sunt: 10325, 10327, 10329, 10345, 10347, 10349. Care este următorul număr generat după numărul 96785?

- a. 96587 b. 98123 c. 96783 d. 98103

3. Fie algoritmul pseudocod alăturat.

Ce afișează algoritmul dacă pentru a se introduc pe rând numerele $2, -2.5, 4, 8, 0$?

```
p ← 1
citește a (număr real)
cât timp a ≠ 0 execută
    p ← p*a
    citește a (număr real)
scrie p
```

- a. -160 b. 160 c. 0 d. 1**

4. De câte ori se execută instrucțiunea de afișare în următoarea secvență de instrucțiuni, unde i este o variabilă de tip întreg?

```
i=3;
while (i<=9)
    i++;
cout<<i; / printf ("%d", i);
```

- a. 6 b. 3 c. 1 d. 7**

5. Fie functia recursivă:

```
void f(int n)
{ if (n)
  { cout<<n%10<<" "; / printf("%d ", n%10);
    f(n/10);}
}
```

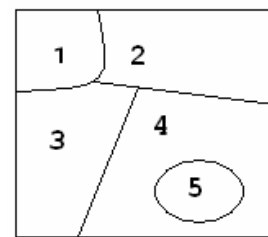
Ce afișează funcția la apelul $f(247)$?

- a.** 7 4 2 **b.** 2 4 7 **c.** 2 7 4 **d.** 4 2 7

6. Se consideră arborele cu 8 noduri, numerotate de la 1 la 8, dat prin lista de muchii: $(1,2)$, $(1,3)$, $(3,4)$, $(3,5)$, $(3,6)$, $(4,8)$, $(4,7)$. Dacă alegem ca rădăcină a arborelui nodul 3, atunci vectorul de tați corespunzător arborelui este:

- a. $(0, 1, 1, 3, 3, 3, 4, 4)$ b. $(2, 3, 0, 3, 4, 5, 6, 7)$
c. $(2, 3, 0, 7, 3, 3, 4, 1)$ d. $(3, 1, 0, 3, 3, 3, 4, 4)$

7. Desenul alăturat reprezintă o hartă cu 5 țări numerotate de la 1 la 5. Se generează toate variantele de colorare a acestei hărți având la dispoziție 4 culori notate cu **A**, **B**, **C**, **D**, astfel încât oricare două țări vecine să nu fie colorate la fel. Prima soluție este (**A**, **B**, **C**, **A**, **B**) având următoarea semnificație: țara 1 e colorată cu **A**, țara 2 e colorată cu **B**, țara 3 e colorată cu **C**, țara 4 e colorată cu **A**, țara 5 e colorată cu **B**. Care din următoarele variante poate reprezenta o soluție de colorare?



- a. (**C**,**D**,**B**,**A**,**A**) b. (**D**,**B**,**D**,**A**,**C**) c. (**D**,**C**,**B**,**D**,**C**) d. (**C**,**B**,**D**,**B**,**A**)
8. Se consideră o stivă **s** inițial vidă. În stivă se introduc în această ordine elementele: 3, 5, 6, 7, 10, 13. Se fac apoi următoarele operații: se scoate un element din stivă, apoi se adaugă elementul cu valoarea 8, se scot apoi două elemente din stivă. Ce element se va găsi în vârful stivei?
- a. 8 b. 7 c. 6 d. 10

SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat:

Se consideră funcția **max(x,y)** care calculează maximumul dintre numere întregi **x** și **y**.

- Ce se va afișa pentru $n=5$ și $v_1=4$, $v_2=-8$, $v_3=5$, $v_4=10$, $v_5=2$? (5p.)
- Rescrieți algoritmul modificând un număr minim de linii din algoritmul inițial astfel încât acesta să afișeze valoarea maximă din vectorul **v**. (5p.)
- Scrieți în limbajul C/C++ definiția completă a funcției **max(x,y)**. (3p.)
- Scrieți programul C/C++ corespunzător algoritmului dat. (7p.)

```

citește n (număr natural
nenul)
pentru i=1,n execută
    citește  $v_i$ 
m ←  $v_1$ 
i ← 3
cât timp i ≤ n execută
    m ← max(m,  $v_i$ )
    i ← i+2
scrie m

```

SUBIECTUL III (30 de puncte)

- Scrieți un program care construiește în memorie un vector **a** cu toate numerele de două cifre, cu cifre distincte, pare, aflate în ordine descrescătoare. Elementele vectorului vor fi în ordine strict descrescătoare. Afișați elementele vectorului **a** pe o linie a ecranului despărțite printr-un spațiu. Exemplu: vectorul va avea elementele: 86, 84, 82, 80, 64, ..., 20. (10p.)
- Scrieți un subprogram **f** cu doi parametri: un șir de caractere **s** cu maxim 50 de caractere și un număr natural **k**, $1 \leq k \leq 50$. Subprogramul va afișa pe ecran primele **k** caractere din șirul **s**. Dacă lungimea șirului **s** este mai mică decât **k**, subprogramul va afișa mesajul **nu este posibil**. De exemplu: **f("abracadabra", 5)** va afișa **abrac**.
f("abracadabra", 15) va afișa **nu este posibil**. (6p.)
 - Scrieți un program care citește de la tastatură **n** șiruri de maxim 50 de caractere ($n \leq 50$) și le afișează pe ecran utilizând funcția **f** astfel: pe primul rând, primul caracter din primul șir, pe al doilea rând, primele două caractere din al doilea șir, ..., pe al **n**-lea rând, primele **n** caractere din al **n**-lea șir; acolo unde șirul e mai scurt decât indicele liniei pe care se află, se va afișa mesajul **nu este posibil**. Exemplu: Dacă $n=5$ și se citesc șirurile de caractere **ionel**, **aurel**, **turcu**, **pop**, **trasca** se va afișa:
i
au
tur
nu este posibil
trasc (4p.)
- În fișierul **bac.txt** pe aceeași linie, despărțite printr-un spațiu, se găsesc maximum 1000 de numere întregi. Scrieți un program care să determine maximumul și minimumul numerelor din fișier folosind o metodă eficientă din punct de vedere al memoriei și al numărului de operații efectuate și afișează pe ecran valoarea maximumului și a minimumului. Exemplu: Dacă în fișierul **bac.txt** se găsesc numerele 100 34 1 78 90 123 0 67 8 -90 -899 -9 88 se va afișa **max=123** și **min=-899**. (10p.)