

Proba scrisă la INFORMATICĂ
Proba E, Limbajul C/C++

Bacalaureat iunie-iulie 2004

Varianta 1

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul efectiv de lucru este de 3 ore.
- În programele cerute, identificatorii utilizați trebuie să respecte precizările din enunț (cursive), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).
- **Atenție:** La întrebările care admit forme diferite pentru limbajul C respectiv C++, aceste forme sunt separate cu ajutorul simbolului /.

I. Pentru fiecare din itemii de mai jos (1–10), scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

1. Pentru subprogramul următor, variabila `x` și variabila reală `r=4.25`, stabiliți câte dintre secvențele alăturate afișează valoarea 4.

<pre>int z1(float& a) { if (a<0) a=-a; a*=10; return (int)a % 10; }</pre>	<pre>x=z1(r); printf("%d",x+z1(r)); / cout<<x+z1(r); ----- x=z1(r); printf("%d",z1(r)+x); / cout<<z1(r)+x; ----- printf("%d",z1(r)+z1(r));/cout<<z1(r)+z1(r); ----- x=z1(r); printf("%d",x+x); / cout<<x+x;</pre>
--	---

a. 4 b. 3 c. 2 d. 1
2. Dacă `x` este o variabilă reală, stabiliți care dintre următoarele variante este o instrucțiune corectă de atribuire în limbajul C/C++?

a. <code>if(x<2) then printf('*');/if(x<2) then cout<<'*';</code>	b. <code>x=4;</code>
c. <code>if(x>4.2) printf('*');/if(x>4.2) cout<<'*';</code>	d. <code>x:=3.4;</code>
3. Numărul de componente conexe formate de un graf neorientat cu 5 noduri și 4 muchii care conține cel puțin un ciclu este:

a. 3	b. 2	c. 1	d. 4
------	------	------	------
4. Variabila `x` este utilizată într-un program pentru a memora numele și prenumele unui elev. Care dintre declarațiile următoare **nu** este corectă:

a. <code>struct{char nume[15]; char prenume[20];}x;</code>
b. <code>char x[60];</code>
c. <code>float x;</code>
d. <code>char x[2][30];</code>
5. Determinarea celui mai mic element dintr-un vector cu numere reale distincte se realizează eficient cu ajutorul:

a. unei parcurgeri a vectorului	b. sortării vectorului
c. metodei backtracking	d. unei formule de calcul
6. Pentru `n` un număr natural de două cifre și `i` o variabilă întregă, suma elementelor vectorului construit prin metoda descrisă de algoritmul alăturat este:

<pre>pentru i=1,n execută dacă i≥n-3 atunci V_i←1 altfel V_i←0</pre>	<pre>■</pre>
--	--------------

a. `n-4` b. `n-3` c. 3 d. 4
7. Dacă `a` este o variabilă globală și la începutul subprogramului `sub` este definită o variabilă locală `a`, atunci în instrucțiunea `a=a+1` din subprogramul `sub`, `a` se referă la:

a. variabila globală <code>a</code>
b. nu se poate defini <code>a</code> ca variabilă globală și variabilă locală
c. variabila locală <code>a</code>
d. unul la variabila locală, iar celălalt la variabila globală
8. Care dintre următoarele secvențe de noduri reprezintă un lanț în grafurile neorientate dat prin matricea de adiacență alăturată, știind că nodurile sunt numerotate de la 1 la 5 corespunzător liniilor tabloului.

<pre>0 1 0 1 0 1 0 0 1 0 0 0 0 1 1 1 1 1 0 1 0 0 1 1 0</pre>	<p>a. 2 1 3 4 b. 1 5 3 4 2 4 c. 4 1 2 4 5 d. 3 4 1 2 5</p>
--	---

9. Într-o listă circulară simplu înlănțuită cu cel puțin 4 elemente, fiecare element memorează în câmpul `adr` adresa elementului următor din listă. Dacă se cunoaște adresa `p` a unui element din listă, atunci este accesibilă adresa elementului din listă precedent celui aflat la adresa `p`?
- Nu.
 - Da, în orice situație.
 - Da, numai dacă `p` este adresa primului element al listei.
 - Da, numai dacă `p` este adresa ultimului element al listei.
10. Pentru a determina toate modalitățile de a scrie pe 9 ca sumă de numere naturale nenule distincte (abstracție făcând de ordinea termenilor), un elev folosește metoda backtracking generând, în această ordine, toate soluțiile: $1+2+6$, $1+3+5$, $1+8$, $2+3+4$, $2+7$, $3+6$ și $4+5$. Aplicând exact aceeași metodă, el determină soluțiile pentru scrierea lui 12. Câte soluții de forma $3+\dots$ există?
- 7
 - 2
 - 1
 - 4

II. Se consideră programul pseudocod alăturat în care s-a folosit notația $x\%y$ pentru restul împărțirii întregi a lui x la y :

4p. 1. Ce se va afișa pentru $n=6$?

4p. 2. Determinați două numere naturale diferite, astfel încât, oricare dintre cele două valori s-ar citi pentru variabila n , valoarea afișată să fie aceeași.

8p. 3. Scrieți programul C/C++ corespunzător algoritmului dat.

4p. 4. Fără a schimba, adăuga sau șterge cuvinte cheie (bold), modificați un număr minim de linii din algoritm astfel încât el să afișeze 1 în cazul în care n este număr prim și 0 în caz contrar. Scrieți, pentru fiecare modificare, numărul liniei și modificarea efectuată.

```

1 citește n (număr natural, n>1)
2 i←1
3 s←1
4 cât timp i<n execută
5     i←i+1
6     dacă n%i=0 atunci
7         s←s+i
8 scrie s

```

III.

5p. 1. Scrieți un program care construiește cel mai mare număr natural format din k cifre distincte, k fiind un număr natural citit ($0 < k < 10$) și afișează rădăcina pătrată a acestuia. De exemplu, pentru $k=2$ se construiește numărul 98 și se afișează $9.8994\dots$ adică rădăcina pătrată a lui 98.

2. Funcția `sum` primește prin intermediul parametrului `v` un vector de numere reale cu 50 de componente și prin intermediul parametrului `k` un număr natural nenul ($1 < k < 50$). El returnează suma tuturor elementelor v_i ale vectorului cu proprietatea că $i \leq k$.

5p. a) Scrieți definiția completă a subprogramului `sum`.

5p. b) Scrieți programul care citește de la tastatură un șir `s` de 50 de numere reale și apoi două numere naturale m și n ($1 < m < n < 50$) și afișează suma elementelor din șir cu indicii cuprinși între m și n (inclusiv m și inclusiv n) folosind apeluri ale funcției `sum`.

5p. 3. Scrieți programul C/C++ care creează fișierul text `BAC.TXT` ce conține pe prima sa linie, în ordine, toate literele mari ale alfabetului englez aflate în alfabet după o literă mare dată de la tastatură. De exemplu, dacă se citește litera `R`, atunci `BAC.TXT` va conține: `STUVWXYZ`

4. Pentru un număr natural k citit de la tastatură ($0 < k < 2000000000$), scrieți un program care determină în mod eficient **câte** numere naturale mai mici sau egale cu k au exact 3 divizori. De exemplu, pentru $k=10$ se afișează valoarea 2 (deoarece există două numere mai mici sau egale cu 10 care au exact 3 divizori și acestea sunt: 4 și 9).

2p. a) Explicați în limbaj natural (4–6 rânduri) metoda utilizată.

1p. b) Explicați în ce constă eficiența metodei alese (2-3 rânduri).

7p. c) Scrieți programul C/C++ corespunzător.