

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

**Varianta 2**

- ♦ 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. Folosind modelul combinărilor, se generează cuvinte cu câte două litere distincte din mulțimea  $\{i, t, e, m\}$  obținându-se, în ordine: *it, ie, im, te, tm, em*. Dacă se utilizează exact aceeași tehnică pentru a genera cuvinte cu patru litere distincte din mulțimea  $\{i, t, e, m, a, x\}$ , atunci numărul de cuvinte generate care încep cu litera *t* este:
  - a. 24
  - b. 12
  - c. 16
  - d. 4
2. Se consideră un graf orientat cu 6 noduri numerotate cu 1, 2, ..., 6 și cu mulțimea arcelor formată doar din arcele:
  - de la fiecare nod numerotat cu un număr neprim *i* ( $i > 1$ ) la toate nodurile numerotate cu numere ce aparțin mulțimii divizorilor proprii ai lui *i* (divizori diferiți de 1 și de *i*);
  - de la nodul numerotat cu 1 la nodul numerotat cu 2;
  - de la fiecare nod numerotat cu un număr prim *i* la nodul numerotat cu  $i+1$ .
 Stabiliți care este numărul de circuite elementare distincte conținute de graful din enunț. (Două circuite sunt distincte dacă diferă prin cel puțin un arc).
  - a. 1
  - b. 2
  - c. 3
  - d. 0
3. Se consideră un graf neorientat cu nodurile: 1, 2, 3, 4, 5, 6, 7, 8 și muchiile: [1, 3], [1, 7], [2, 6], [3, 7], [5, 2], [5, 6], [8, 4]. Care este numărul minim de muchii ce pot fi adăugate astfel încât graful să devină conex?
  - a. 0
  - b. 2
  - c. 3
  - d. 4
4. În secvența de instrucțiuni alăturată, variabila *s* memorează un șir de caractere, iar variabila *i* este de tip *int*. Știind că în urma executării secvenței s-a afișat succesiunea de caractere *p\*r\*o\*b\*a\*\*\*e\** stabiliți care este șirul de caractere memorat de variabila *s*.
 

```
for(i=0; i<strlen(s); i++)
    printf("%c", s[i]);
    /cout<<s[i]<<' ';
```

  - a. *proba\*e*
  - b. *\*p\*r\*o\*b\*a\*\*\*e*
  - c. *probae*
  - d. *p\*r\*o\*b\*a\*e\**
5. Știind că variabilele *a* și *b* sunt utilizate pentru a memora două numere naturale cu cel puțin două cifre fiecare, stabiliți care dintre instrucțiunile de mai jos determină, în urma executării, inițializarea variabilei *m* cu cifra zecilor a numărului obținut prin adunarea numerelor memorate în *a* și *b*.
  - a. *m=a%100/10+b%100/10;*
  - b. *m=(a+b)%100;*
  - c. *m=(a+b)/10%100;*
  - d. *m=(a+b)%100/10;*
6. În secvența de instrucțiuni alăturată, *n* și *y* sunt variabile întregi. Valoarea variabilei *y* la finalul executării secvenței este:
 

```
n=156; y=770;
while(n*y>0)
    if (n>y) n=n*y;
    else y=y*n;
y=y+n;
```

  - a. 13
  - b. 0
  - c. 2
  - d. 4
7. Într-o listă simplu înălțuită, cu cel puțin patru elemente, fiecare element reține în câmpul *urm* adresa elementului următor din listă. Dacă *p*, *q* și *r* sunt adresele a trei elemente din listă astfel încât *q==p->urm->urm* și *r->urm==p->urm->urm* atunci ordinea logică a elementelor în listă (elementele fiind identificate prin adrese) este:
  - a. *q, r, p*
  - b. *r, q, p*
  - c. *p, r, q*
  - d. *p, q, r*

8. Pentru definiția alăturată a subprogramului `ex`, stabiliți ce se afișează la apelul `ex(120)?`
- ```
void ex(int x)
{
    printf("%d",x%10); /cout<<x%10;
    if(x!=0) ex(x/10);
}
```
- a. 0120                      b. 021                      c. 210                      d. 0210

### SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat:

1. Ce se va afișa pentru  $n=8$ ? (5p.)
2. Scrieți o valoare strict pozitivă pentru variabila  $n$  astfel încât să se afișeze valoarea 132? (4p.)
3. Scrieți programul pseudocod care să fie echivalent cu algoritmul dat și care să nu conțină nicio structură repetitivă. (3p.)
4. Scrieți programul C sau C++ corespunzător algoritmului dat. (8p.)

```
citește n
    {număr natural nenul}
s ← 0
x ← 2
pentru i ← 1, n execută
    s ← s + x
    x ← x + 2
scrie s
```

### SUBIECTUL III (30 de puncte)

1. Scrieți programul C sau C++ care construiește în memorie o matrice pătratică cu  $n$  linii și  $n$  coloane formată numai din valori 0, 1 și 2 astfel încât elementele de pe diagonala secundară și cea principală să fie egale cu 0, elementele situate între diagonalele matricei, în partea superioară și inferioară a acesteia, să fie egale cu 1, iar restul elementelor din matrice să fie egale cu 2. Valoarea lui  $n$  (număr natural,  $2 < n < 23$ ) se citește de la tastatură, iar matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu spații între elementele fiecărei linii (ca în exemplu).

De exemplu, pentru  $n=5$  se construiește în memorie și se afișează matricea:

```
0 1 1 1 0
2 0 1 0 2
2 2 0 2 2
2 0 1 0 2
0 1 1 1 0
```

(10p.)

2. Realizați următoarele cerințe utilizând limbajul C/C++:

a) Scrieți definiția completă a unei funcții `nr` cu doi parametri, funcție care:

- primește prin intermediul parametrilor `a` și `b` două numere reale;
- returnează numărul de numere întregi cuprinse între valorile parametrilor `a` și `b`, inclusiv.

De exemplu, pentru valorile 10.5 și 7 ale parametrilor `a` și `b`, funcția va returna valoarea 4 deoarece între valorile 7 și 10.5 sunt 4 numere întregi: 7, 8, 9 și 10. (4p.)

b) Scrieți programul în care se citesc de la tastatură numerele naturale nenule `a` și `k` ( $a \in [1000, 9999]$ ,  $4 \leq k \leq 9$ ) și în care se determină câte numere naturale de câte `k` cifre se divid cu `a`, folosind apeluri ale subprogramului `nr`. Programul va afișa pe ecran numărul de numere determinat.

De exemplu, dacă `a=2007` și `k=4`, atunci numărul afișat este 4, deoarece sunt 4 numere de 4 cifre care se divid cu 2007 (2007, 4014, 6021, 8028). (6p.)

Observații: Concepția prelucrărilor de la a) și b) trebuie să aibă în vedere criteriul de eficiență privind timpul de executare; dacă se rezolvă corect cerința b) fără a se utiliza apeluri ale funcției definite la a), se va acorda un punctaj parțial.

3. Scrieți programul C sau C++ care citește de la tastatură un șir de cel mult 30 de litere ale alfabetului englez și creează fișierul text **BAC.TXT** ce conține șirul de caractere dat și toate sufixele acestuia de lungime cel puțin 1, fiecare pe câte o linie, în ordinea crescătoare a lungimii sufixelor, aliniate la stânga. De exemplu, dacă se citește șirul `teste`, atunci **BAC.TXT** va conține:

```
e
te
ste
este
teste
```

(10p.)