

Varianta 4

- ◆ 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. Știind că variabila **x** este utilizată pentru a memora produsul a 12 numere reale (cu cel mult trei zecimale), stabiliți care este declararea corectă a variabilei **x**.
 a. `float x[12];` b. `float x[12][3];` c. `float x;` d. `float x[3];`
 2. 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. 2 b. 0 c. 3 d. 4
 3. 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 se termină cu litera **a** este:
 a. 4 b. 12 c. 24 d. 5
 4. Algoritmul alăturat determină în variabila **K** suma tuturor componentelor pozitive dintre cele 15 componente întregi ale vectorului **a**. Cu ce pot fi înlocuite punctele de suspensie?

```

K ← 0
pentru i ← 1, 15 execută
    dacă ai ≥ ... atunci K ← ... + K
    
```

 a. 0 și a_1 b. 0 și a_i c. K și a_i d. a_1 și a_i
 5. Într-o listă simplu înlănțuită (cu cel puțin patru elemente) fiecare element reține în câmpul **urm** adresa elementului următor din listă, iar **p** memorează adresa celui de-al treilea element din listă. Atunci **q** reține adresa primului element din listă dacă și numai dacă este satisfăcută condiția:
 a. `p->urm->urm==q->urm` b. `p->urm->urm==q`
 c. `q->urm->urm->urm==p->urm` d. `q->urm->urm==p->urm`
 6. Un graf orientat are 8 arce și fiecare nod al grafului are gradul interior un număr nenul. Doar **două** dintre noduri au gradul interior un număr par, restul nodurilor având gradele interioare numere impare. Care este numărul maxim de noduri pe care poate să le aibă graful?
 a. 7 b. 8 c. 5 d. 6
 7. Ce valori vor avea variabilele întregi **a** și **b** la finalul executării secvenței de instrucțiuni alăturate?

```

a=1; b=0;
do
    { b=b+1; a=a*b; }
while(a<125);
    
```

 a. a=126 și b=6 b. a=720 și b=6 c. a=125 și b=5 d. a=720 și b=125
 8. Pentru definiția alăturată a subprogramului **ex**, stabiliți ce se afișează la apelul `ex(2,10)`?

```

void ex(int i, int j)
{
    if(i <= j) {
        printf("%d", i); / cout << i;
        ex(i+1, j-1);
        if(i%2 != 0)
            printf("%d", 9-j); / cout << 9-j;
    }
}
    
```

 a. 2345620 b. 246357 c. 2345646 d. 234520

SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat:

S-a notat cu $x \div y$ restul împărțirii numărului întreg x la numărul întreg y și cu $[z]$ partea întreagă a numărului real z .

1. Ce se va afișa pentru $n=123321$? (5p.)
2. Scrieți două valori, numere naturale consecutive formate fiecare din câte cinci cifre, pentru variabila n astfel încât rezultatele afișate să fie diferite. (3p.)
3. Scrieți programul pseudocod care să fie echivalent cu algoritmul dat și care să conțină o structură repetitivă cu test inițial. (4p.)
4. Scrieți programul C sau C++ corespunzător algoritmului dat. (8p.)

citește n {număr natural}

$x \leftarrow 0$

```
repetă
     $x \leftarrow x * 10 + n \% 10$ 
     $n \leftarrow [n / 10]$ 
     $e \leftarrow n * (n - x) * ([n / 10] - x)$ 
până când  $e = 0$ 
```

```
dacă  $n > 0$ 
    atunci scrie 'A'
    altfel scrie 'B'
■
```

SUBIECTUL III (30 de puncte)

1. Scrieți programul C sau C++ care citește de la tastatură un număr natural impar n ($2 < n < 23$) și construiește în memorie o matrice pătratică cu n linii și n coloane formată numai din valori 1, 2 și 3 astfel încât elementele din matrice pe coloana mediană (situată în mijlocul matricei) și linia mediană să fie egale cu 1, elementul situat la intersecția liniei și coloanei mediane să fie egal cu 2, iar restul elementelor din matrice să fie egale cu 3. 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:

```
3 3 1 3 3
3 3 1 3 3
1 1 2 1 1
3 3 1 3 3
3 3 1 3 3
```

(10p.)

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

a) Scrieți definiția completă a unei funcții **produs** cu un parametru, funcție care:

- primește prin intermediul parametrului a un număr natural cu cel mult 8 cifre ($a > 1$);
- returnează produsul divizorilor primi ai valorii parametrului a .

De exemplu, pentru valoarea 300 a parametrului a , funcția va returna valoarea 30 deoarece $a = 2^2 * 3 * 5^2$ și $2 * 3 * 5 = 30$. (6p.)

b) Scrieți programul în care se citesc de la tastatură numerele naturale nenule a, b și c de cel mult 8 cifre fiecare ($a > 1, b > 1, c > 1$) și în care se verifică dacă numerele citite au exact aceiași factori în descompunerea lor în factori primi, folosind apeluri ale subprogramului **produs**. Dacă cele trei numere au exact aceiași factori (exponenții lor putând să difere), programul va afișa pe ecran mesajul **DA**, în caz contrar va afișa mesajul **NU**.

De exemplu, dacă $a=300, b=1500$ și $c=30$, atunci se va afișa mesajul **DA**, iar pentru numerele $a=300, b=700$ și $c=140$, se va afișa mesajul **NU**. (4p.)

3. Fișierul text **BAC.TXT** conține o singură linie pe care este scrisă o expresie aritmetică fără paranteze, în care operanzii sunt cifre, iar operatorii sunt $+$ sau $-$. La sfârșitul expresiei, în fișier, se află caracterul $=$. Expresia are cel mult 30 de operanzi și este corectă (primul și ultimul caracter din expresie sunt cifre, între două cifre există exact un operator, nu există două cifre una lângă alta și nu conține alte caractere). Scrieți programul C/C++ care citește din fișierul text **BAC.TXT** expresia aritmetică și determină în mod eficient, din punctul de vedere al spațiului de memorie folosit, rezultatul evaluării acesteia. Programul va afișa pe ecran atât expresia cât și rezultatul evaluării, pe o singură linie, sub forma: **expresie = rezultat** (ca în exemplu).

De exemplu, dacă fișierul **BAC.TXT** conține: $2+3-4+6+8-5=$ atunci rezultatul evaluării expresiei este 10, afișându-se pe ecran în forma: $2+3-4+6+8-5 = 10$ (10p.)