

Examenul de bacalaureat național 2020
Proba E. d)
Informatică
Limbajul C/C++

Testul 20

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filieră vocațională, profil militar, specializare matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- Identificatorii utilizați în rezolvări 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). Datele de intrare se consideră corecte, validarea lor nefiind necesară.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

SUBIECTUL I (20 de puncte)

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

1. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul natural memorat în variabila întreagă **x** are exact două cifre.

- | | |
|--|--|
| a. <code>x/100!=0 x/10==0</code> | b. <code>x/100==0 && x/10!=0</code> |
| c. <code>x%100!=0 x%10==0</code> | d. <code>x%100==0 && x%10!=0</code> |

2. Subprogramul **f** este definit alăturat. Scrieți valoarea lui **f(3)**.

```
int f(int n)
{
    int r, i;
    r=0;
    for(i=1;i<=n;i++) r=r+i+f(n-i);
    return r;
}
```

- | | | | |
|------|------|------|-------|
| a. 3 | b. 6 | c. 9 | d. 11 |
|------|------|------|-------|

3. Utilizând metoda backtracking, se generează toate modalitățile de a pregăti o masă festivă, servind, într-o anumită ordine, preparatele din mulțimea {**ciuperci cu spanac**, **legume la cuptor**, **pere umplute cu nucă**, **panna cotta de cocos cu vanilie**, **salată cu smochine**, **tort de lămâie**}, având în vedere următoarele restricții: ciupercile vor fi servite înaintea legumelor, legumele înaintea salatei și atât tortul, cât și perele umplute, înainte de panna cotta. Două soluții sunt distincte dacă ordinea servirii preparatelor este diferită. Primele trei soluții generate sunt, în această ordine: (**ciuperci cu spanac**, **legume la cuptor**, **pere umplute cu nucă**, **salată cu smochine**, **tort de lămâie**, **panna cotta de cocos cu vanilie**), (**ciuperci cu spanac**, **legume la cuptor**, **pere umplute cu nucă**, **tort de lămâie**, **panna cotta de cocos cu vanilie**, **salată cu smochine**), (**ciuperci cu spanac**, **legume la cuptor**, **pere umplute cu nucă**, **tort de lămâie**, **salată cu smochine**, **panna cotta de cocos cu vanilie**). Indicați cea de a șasea soluție generată.

- | |
|---|
| a. (ciuperci cu spanac , legume la cuptor , salată cu smochine , tort de lămâie , pere umplute cu nucă , panna cotta de cocos cu vanilie) |
| b. (ciuperci cu spanac , legume la cuptor , tort de lămâie , pere umplute cu nucă , salată cu smochine , panna cotta de cocos cu vanilie) |
| c. (ciuperci cu spanac , legume la cuptor , tort de lămâie , pere umplute cu nucă , panna cotta de cocos cu vanilie , salată cu smochine) |
| d. (ciuperci cu spanac , legume la cuptor , tort de lămâie , salată cu smochine , pere umplute cu nucă , panna cotta de cocos cu vanilie) |

4. Un arbore cu 9 noduri, numerotate de la 1 la 9, este reprezentat prin vectorul de „tați” (8, 7, 0, 6, 2, 8, 3, 3, 2). Indicați rădăcina arborelui.

- | | | | |
|------|------|------|------|
| a. 1 | b. 3 | c. 5 | d. 9 |
|------|------|------|------|

5. Matricea de adiacență a unui graf neorientat cu 100 de noduri are 9900 de elemente nule. Indicați numărul maxim de componente conexe ale grafului.

- | | | | |
|-------|-------|---------|---------|
| a. 50 | b. 90 | c. 1000 | d. 9800 |
|-------|-------|---------|---------|

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

S-a notat cu $a \div b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .

a. Scrieți ce se afișează dacă se citește numărul 12. (6p.)

b. Scrieți două numere din intervalul $[2, 10^2]$, unul par și unul impar, care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze 9. (6p.)

c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)

d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat structura `cât timp...execută` cu o structură repetitivă de alt tip. (6p.)

```

citește n
    (număr natural nenul)
d ← 1; s ← 0
cât timp d * d < n execută
    dacă n % d = 0 și d % 2 ≠ [n/d] % 2 atunci
        s ← s + d + [n/d]
    d ← d + 1
dacă d * d = n atunci s ← s + d
scrie s
    
```

2. Variabila p memorează simultan date specifice pentru prepararea unei prăjituri: numărul de ingrediente necesare și, pentru fiecare ingredient, codul acestuia și cantitatea necesară. Știind că expresiile C/C++ de mai jos au ca valori numere naturale din intervalul $[2, 20]$, reprezentând numărul de ingrediente necesare pentru prepararea unei prăjituri, codul primului ingredient și cantitatea necesară din acesta, scrieți definiția unei structuri cu eticheta `prajitura`, care permite memorarea datelor specifice pentru prepararea unei prăjituri, și declarați corespunzător variabila p .

`p.numar` `p.ingredient[0].cod` `p.ingredient[0].cantitate` **(6p.)**

3. Variabilele i și j sunt de tip întreg, iar variabila a memorează un tablou bidimensional cu 7 linii și 7 coloane, numerotate de la 0 la 6, având inițial toate elementele egale cu caracterul `*`. Fără a utiliza alte variabile, scrieți secvența de instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila a să memoreze tabloul alăturat.

```

for (i=0; i<7; i++)
    for (j=0; j<7; j++)
        .....
    
```

```

b a a a a a b
b b a a a b b
b b b a b b b
b b b b b b b
b b b a b b b
b b a a a b b
b a a a a a b
    
```

(6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul `transformareBaza10` are doi parametri, b și n , prin care primește câte un număr natural ($b \in [2, 10]$, $n \in [0, 10^9]$). Subprogramul returnează suma tuturor produselor de forma $c \cdot b^k$, unde c este cifra de pe poziția k în scrierea numărului n ; pozițiile sunt numerotate de la dreapta la stânga, iar cifra unităților este pe poziția 0. Scrieți definiția completă a subprogramului.

Exemplu: dacă $b=2$ și $n=10010$, subprogramul returnează 18 ($18=1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$). **(10p.)**

2. Un text cu cel mult 100 de caractere conține cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere mici ale alfabetului englez, iar numerele sunt reale, pozitive, cu partea întreagă și partea zecimală separate prin simbolul virgulă, sau numai cu partea întreagă, ca în exemplu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și îl transformă în memorie, înlocuind fiecare număr real cu partea întreagă a acestuia.

Exemplu: pentru textul

partea întreaga a lui 5,75 este egala cu a lui 5,25 si cu a lui 5 si este 5
se afișează pe ecran

partea întreaga a lui 5 este egala cu a lui 5 si cu a lui 5 si este 5 **(10p.)**

3. Fișierul `bac.txt` conține un șir de cel mult 10^6 numere naturale din intervalul $[0, 10^3]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran suma maximă obținută adunând numere cu aceeași paritate, aflate pe poziții consecutive în șirul aflat în fișier. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fișierul `bac.txt` conține valorile 10 115 1 5 2 2 2 4 7 3 100 20 2 7 se afișează pe ecran numărul 122

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)

b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)