

Examenul de bacalaureat național 2019
Proba E. d)
Informatică
Limbajul C/C++

Simulare

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.

- Indicați expresia C/C++ cu valoarea 1.
 - $21/3*19==3/19*21$
 - $21/19*3==19/3*21$
 - $19/3*21==3/21*19$
 - $19/21*3==3/19*21$
- Utilizând metoda backtracking, se generează toate drapelurile formate din câte 3 culori distincte din mulțimea {alb, galben, negru, roșu, verde}. Două drapeluri sunt distincte dacă diferă prin cel puțin o culoare sau prin ordinea culorilor. Primele patru soluții obținute sunt, în această ordine: (alb, galben, negru), (alb, galben, roșu), (alb, galben, verde) și (alb, negru, galben). Indicați soluția generată imediat înainte de (galben, verde, alb).
 - (negru, roșu, verde)
 - (negru, alb, galben)
 - (galben, verde, roșu)
 - (galben, roșu, verde)
- Subprogramul `f` alăturat este incomplet definit. Indicați expresia cu care pot fi înlocuite punctele de suspensie, astfel încât valoarea lui `f(2019, 1)` să fie egală cu numărul divizorilor pozitivi ai lui 2019.

```
int f(int n, int d)
{ if(.....) return 0;
  if(d*d==n) return 1;
  if(n%d==0) return 2+f(n,d+1);
  return f(n,d+1);
}
```

 - $d*d>n$
 - $d>n/2$
 - $d>n$
 - $d>2*n$
- Un graf neorientat cu 8 noduri, numerotate de la 1 la 8, are muchiile [1,2], [1,6], [4,6], [3,6], [6,5], [5,3], [3,4], [7,8], [8,2]. Trei noduri care nu aparțin niciunui ciclu în acest graf pot fi:
 - 1, 3, 4
 - 2, 7, 8
 - 3, 5, 6
 - 5, 6, 8
- Un arbore cu 19 noduri, numerotate de la 1 la 19, are ca rădăcină nodul numerotat cu 1; nodul 1 are un singur fiu, și anume nodul 2, iar fiecare nod i ($i \geq 2$) fie este frunză, fie are drept fii (descendenți direcți) noduri numerotate cu valori din intervalul $[i+1, 2 \cdot i - 1]$. Numărul maxim de frunze ale arborelui este:
 - 9
 - 12
 - 13
 - 16

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

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

- a) Scrieți valoarea afișată dacă se citește, în această ordine, numerele 5, 19 și 4. **(6p.)**
- b) Dacă pentru variabila b se citește numărul 2019, iar pentru variabila k se citește numărul 5, scrieți cea mai mică și cea mai mare valoare care pot fi citite pentru variabila a astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, valoarea afișată să fie 0. **(6p.)**
- c) Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

```

citește a,b,k
(numere naturale, 1 ≤ a ≤ b, k > 1)
p ← 0; y ← 0; i ← b
cât timp i ≥ a execută
    x ← i; p ← 0
    cât timp x % k = 0 execută
        x ← [x/k]; p ← p + 1
    ■
    dacă p ≠ 0 și (p < p_m sau p_m = 0) atunci
        p_m ← p; y ← i
    ■
    i ← i - 1
    ■
scrie y
    
```

- d) Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind prima structură `cât timp...execută` cu o structură repetitivă de tip `pentru...execută`. **(6p.)**

2. Expresiile de mai jos au ca valori numere naturale, reprezentând următoarele informații memorate pentru un eveniment din anul 2019: data desfășurării sale (ziua și luna) și un identificator. Scrieți definiția unei structuri cu eticheta **eveniment**, care să permită memorarea informațiilor menționate pentru un eveniment, și declarați corespunzător variabila e , de acest tip.

`e.data.zi` `e.data.luna` `e.id` **(6p.)**

3. Variabilele i și j sunt de tip întreg, iar variabila a memorează un tablou bidimensional cu 5 linii și 5 coloane, numerotate de la 0 la 4, având inițial toate elementele nule.

Fără a utiliza alte variabile decât cele menționate, 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<5; i++)
    for (j=0; j<5; j++)
        .....
    
```

0	1	2	3	4
2	3	4	5	6
4	5	6	7	8
6	7	8	9	10
8	9	10	11	12

(6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul **inserare** are un singur parametru, n , prin care primește un număr natural ($n \in [10, 10^5]$). Subprogramul furnizează prin același parametru numărul obținut din n prin inserarea, între oricare două cifre alăturate ale sale, a valorii absolute a diferenței acestora. Scrieți definiția completă a subprogramului.

Exemplu: dacă $n=7255$, atunci, după apel, $n=7\underline{5}2\underline{3}5\underline{0}5$. **(10p.)**

2. Într-un text de cel mult 50 de caractere cuvintele sunt separate prin câte un spațiu și sunt formate din litere mari ale alfabetului englez, urmate eventual de caracterul . (punct), dacă sunt scrise prescurtat. Textul reprezintă numele unei instituții de învățământ și doar cuvintele din mulțimea {**COLEGIUL, LICEUL, NATIONAL, TEORETIC**} pot fi prescurtate, eliminându-se ultimele lor litere. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și construiește în memorie, apoi afișează pe ecran, numele instituției scris fără prescurtări.

Exemplu: dacă se citește textul **COLEG. NATIONAL DE INFORMATICA** sau textul **COLEG. NAT. DE INFORMATICA** se obține **COLEGIUL NATIONAL DE INFORMATICA** **(10p.)**

3. Un șir format din $2 \cdot n$ numere naturale se numește **paritar** dacă fiecare dintre primii săi n termeni fie are aceeași paritate cu oricare dintre ultimii săi n termeni, fie este strict mai mic decât oricare număr de paritate diferită aflat printre aceștia.

Fișierul **bac.txt** conține numere naturale din intervalul $[0, 10^6]$: pe prima linie un număr nenul, n , iar pe a doua linie un șir de $2 \cdot n$ numere, separate prin câte un spațiu. Se cere să se afișeze pe ecran mesajul **DA**, în cazul în care șirul aflat în fișier este paritar, sau mesajul **NU**, în caz contrar. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.

Exemplu: dacă fișierul are unul dintre conținuturile de mai jos, se afișează pe ecran mesajul **DA**.

5	5
20 3 11 4 15 25 49 18 53 16	20 3 11 4 15 25 49 81 53 61

- a) Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- b) Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**