Examenul național de bacalaureat 2026
Proba E. d)
Informatică
Limbajul C/C++

Varianta 3

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

- **Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.**
- **Timpul de lucru efectiv este de trei 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. Variabilele **x** și **y** sunt întregi. Indicați o expresie C/C++ care are valoarea **1** dacă și numai dacă numerele naturale memorate în variabilele **x** și **y** au aceeași paritate.

a. `(x*y)%2==0` **b.** `x%2==0 && y%2==0`

c. `(x+y)%2==0` **d.** `!(x%2==y%2)`

2. Subprogramul alăturat, **f**, este incomplet definit. Indicați expresia cu care pot fi înlocuite punctele de suspensie, astfel încât, în urma apelului de mai jos, să se afișeze **7**.

```
f(35,17);
```

```
void f(int n, int d)
{ if(n%d==0)
    cout<<d; | printf("%d",d);
  else f(........);
}
```

a. `n+1,d+1` **b.** `n-1,d-1` **c.** `n,d+1` **d.** `n,d-1`

3. Variabila **p**, declarată alăturat, memorează primul termen și rația unei progresii aritmetice. Știind că diferența dintre un termen al progresiei și termenul anterior este egală cu rația, indicați expresia a cărei valoare este egală cu cel de al **10**-lea termen al progresiei.

```
struct progresie
{  int prim;
   int ratie;
}p;
```

a. `prim.p+ratie.p*9` **b.** `p.prim+9*p.ratie`

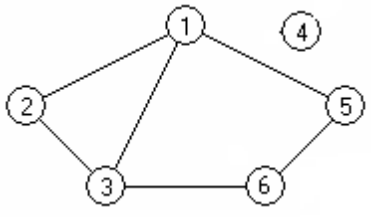
c. `progresie.prim.p+9*progresie.ratie.p` **d.** `p.progresie.prim+9*p.progresie.ratie`

4. Un graf orientat cu **5** vârfuri, numerotate de la **1** la **5**, este reprezentat prin matricea de adiacență alăturată. Indicați numărul de vârfuri ale grafului cu proprietatea că valoarea absolută a diferenței gradelor intern și extern este **1**.

```
0 1 0 0 1
1 0 0 0 0
1 1 0 1 0
0 0 1 0 1
0 0 1 1 0
```

a. `2` **b.** `3` **c.** `4` **d.** `5`

5. Un graf neorientat cu **6** noduri, numerotate de la **1** la **6**, este reprezentat alăturat. Indicați numărul maxim de muchii care pot fi adăugate, astfel încât graful obținut să fie eulerian.



a. `9` **b.** `6` **c.** `3` **d.** `2`

SUBIECTUL al II-lea (40 de puncte)

1. Se consideră algoritmul alăturat, reprezentat în pseudocod.

a. Scrieți valorea afișată dacă se citesc, în această ordine, numerele **8**, **3**, **1**, **5**, **2**, **4**, **5**, **5**, **7**. **(6p.)**

b. Dacă pentru **n** se citește numărul **3**, scrieți un set de date care pot fi citite în continuare, astfel încât, în urma executării algoritmului, să se afișeze **0**. **(6p.)**

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

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

```
citește n
   (număr natural nenul)
 s←0; x←0; p←1
┌repetă
 citește y (număr natural nenul)
┌dacă y>x atunci
 s←s+y; x←y
altfel
┌dacă x=y atunci s←s-y
└■
└■
 p←p+1
└până când p>n
 scrie s
```

2. Utilizând metoda backtracking se generează toate seturile de câte cel mult trei materiale de promovare turistică din mulţimea ordonată {**broșură**, **ghid**, **pliant**, **revistă**}, astfel încât din același set să nu facă parte simultan un pliant și o revistă. Două seturi sunt distincte dacă diferă prin cel puțin un material. Primele cinci soluții generate sunt, în această ordine, (**broșură**), (**broșură**, **ghid**), (**broșură**, **ghid**, **pliant**), (**broșură**, **ghid**, **revistă**), (**broșură**, **pliant**). Scrieți a șasea și a șaptea soluție generată. **(6p.)**

3. Variabilele **i** şi **j** sunt de tip întreg, iar variabila **a** memorează un tablou bidimensional cu **5** linii şi **5** coloane, numerotate de la **0** la **4**, având iniţial toate elementele nule.
Fără a utiliza alte variabile, scrieți secvența C/C++ de mai jos, înlocuind punctele de suspensie cu instrucțiuni adecvate, dintre care cel mult patru atribuiri, astfel încât, în urma executării secvenței obținute, tabloul memorat în variabila **a** să aibă elementele alăturate.

```
for(i=0;i<5;i++)
  for(j=0;j<5;j++)
    ..................
```
(6p.)

```
0 9 10 19 20
1 8 11 18 21
2 7 12 17 22
3 6 13 16 23
4 5 14 15 24
```

SUBIECTUL al III-lea (30 de puncte)

1. Numim **număr uniform** asociat unei valori naturale numărul obținut din aceasta prin eliminarea fie a tuturor cifrelor sale pare, fie a tuturor cifrelor sale impare.
Exemplu: lui **19472** i se asociază numerele uniforme **197** și **42**.
Subprogramul **ImparPar** are un singur parametru, **n**, prin care primește un număr natural cu toate cifrele nenule (**$n \in [11, 10^9)$**), având cel puțin o cifră pară și cel puțin o cifră impară. Subprogramul returnează un număr obținut din cifrele numărului uniform impar asociat lui **n**, urmate de cifrele numărului uniform par asociat lui **n**, ca în exemplu. Scrieți definiția completă a subprogramului.
Exemplu: dacă **n=19472**, subprogramul returnează numărul **19742**. **(10p.)**

2. Scrieți un program C/C++ care citește de la tastatură un cuvânt format din cel puțin două și cel mult **100** de caractere, numai litere mici ale alfabetului englez, și afișează pe ecran mesajul **alterneaza**, dacă în text nu există două vocale sau două consoane pe poziții consecutive, sau mesajul **nu alterneaza**, în caz contrar.
Exemplu: pentru cuvintele **imunopolizaharide** sau **pas** se afișează mesajul **alterneaza**
iar pentru cuvântul **aer** se afișează mesajul **nu alterneaza** **(10p.)**

3. De-a lungul timpului, o persoană a participat la mai multe concursuri de șah și de go. **Un rezultat motivant** corespunde unei poziții mai bune (cu un număr mai mic) în clasamentul unui concurs, comparativ cu poziția obținută la concursul anterior de același tip.
Fișierul **bac.txt** conține un șir de cel mult **10^4** numere naturale, reprezentând rezultatele obținute la concursurile la care a participat, în ordine cronologică: pentru fiecare dintre rezultate, câte o pereche formată din poziția ocupată în clasamentul concursului corespunzător (un număr din intervalul **[1,300]**), urmată de valoarea **1**, dacă acesta a fost concurs de șah, sau de valoarea **2**, dacă acesta a fost concurs de go. Fiecare pereche se află pe câte o linie a fișierului, cu valorile separate printr-un spațiu.
Se cere să se afișeze pe ecran, separate printr-un spațiu, două valori naturale, reprezentând, în această ordine, numărul de rezultate motivante obținute la concursurile de șah, respectiv la cele de go. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.
Exemplu: dacă fișierul conține numerele alăturate, pe ecran se afișează **3 2**
pentru că la șah a obținut trei rezultate motivante (poziția **3** după poziția **7**, poziția **1** după poziția **5** și poziția **1** după poziția **2**), iar la go două rezultate motivante (poziția **5** după poziția **8**, poziția **1** după poziția **5**).

```
6 1
8 2
7 1
5 2
3 1
4 1
5 1
1 2
4 2
1 1
2 1
1 1
```

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