

Examenul național de bacalaureat 2023

Proba E. d)
Informatică
Limbajul C/C++

Varianta 6

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 întregi **m** și **n** memorează câte un număr natural. Indicați expresia C/C++ care are valoarea 1 dacă și numai dacă fiecare dintre numerele memorate în variabilele **m** și **n** este par.

- a. **(m+n)%2==0** b. **(m-n)%2==0** c. **m%2+n%2==0** d. **m%2-n%2==0**

2. Subprogramul **f** este definit alăturat. Indicați valoarea **f(35, 3)**.
- ```
int f(int n, int k)
{ if (n==0 || k==0) return 0;
 return n%k+10*f(n/k, k);
}
```

- a. 1022      b. 355      c. 102      d. 33

3. Variabila **p**, declarată alăturat, memorează dimensiunile unui paralelipiped dreptunghic: lungimea, lățimea, respectiv înălțimea.
- ```
struct paralelipiped
{ float lg, lt, h;
} p;
```

Volumul paralelipipedului este egal cu produsul celor trei dimensiuni ale sale. Indicați o instrucțiune prin care i se atribuie variabilei reale **v** valoarea volumului paralelipipedului corespunzător variabilei **p**.

- a. **v=lg*lt*h;** b. **v=lg.p*lt.p*h.p;** c. **v=p(lg*lt*h);** d. **v=p.lg*p.lt*p.h;**

4. Utilizând metoda backtracking se generează toate șirurile de 5 litere distincte din mulțimea {**A, M, U, R, G**}, astfel încât în fiecare șir litera **R** precede literele **A** și **M**. Primele cinci soluții generate sunt, în această ordine: **URAMG, URAGM, URMAG, URMGA, URGAM**. Indicați soluția generată imediat după **GURMA**.

- a. **GARUM** b. **GRAMU** c. **GRAUM** d. **GRUAM**

5. Într-o zonă sunt 23 de obiective turistice, numerotate de la 1 la 23; zona este reprezentată ca un graf orientat în care vârfurile corespund acestor obiective, iar arcele sunt stabilite în funcție de modalitatea de deplasare între două obiective, astfel: de la fiecare obiectiv numerotat cu **i** ($i \in [1, 23]$) se poate merge direct doar la obiectivul numerotat cu **2*i**, dacă acesta există, sau doar la obiectivul numerotat cu **i-1**, în caz contrar. Indicați numărul maxim de vârfuri ale unui drum elementar în acest graf, cu extremitatea inițială 1.

- a. 5 b. 11 c. 16 d. 21

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

- a. Scrieți ce se afișează în urma executării algoritmului, dacă se citește numărul 3. (6p.)
- b. Scrieți două numere care pot fi citite, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, în total, numărul de cifre 2 afișate să fie în intervalul [20, 50]. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind una dintre structurile **pentru...execută** cu o structură de tip **cât timp...execută**. (6p.)

```
citește n
(n număr natural nenul)
pentru i ← 1, n execută
  pentru j ← 1, n execută
    dacă j ≤ i atunci
      scrie 2
    altfel
      scrie 3
```

2. Un arbore cu 8 noduri, numerotate de la 1 la 8, este reprezentat prin vectorul de „tați” (0, 1, 1, 2, 2, 4, 5, 1). Enumerați toți descendenții nodului numerotat cu 2. (6p.)
3. Un șir de caractere **a** este numit **prefix** al unui șir de caractere **b** dacă este identic cu **b** sau dacă **b** se obține din **a** prin adăugarea la dreapta a unor alte caractere.
Variabila **k** este de tip întreg, iar variabila **s** permite accesul la un șir de cel mult 20 de caractere. Scrieți secvența de instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran, în ordinea descrescătoare a lungimii, separate prin câte un spațiu, toate prefixele șirului accesat prin variabila **s**, fiecare încheindu-se cu prima literă a șirului **s**, ca în exemplu. Declarați eventualele alte variabile utilizate.
Exemplu: pentru șirul **elemente** se afișează: elemente eleme ele e
for (**k=strlen** (**s**) -1; **k>=0**; **k--**)
{ } (6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul **Putere** are trei parametri:
- n**, prin care primește un număr natural ($n \in [2, 10^9]$);
 - x** și **p**, prin care furnizează două numere naturale cu proprietatea că $n = x^p$, iar **x** este cel mai mic număr cu această proprietate.
- Scrieți definiția completă a subprogramului.
Exemplu: dacă **n=16** atunci, după apel, **x=2** și **p=4**, dacă **n=216** atunci, după apel, **x=6** și **p=3**, iar dacă **n=12** atunci, după apel, **x=12** și **p=1**. (10p.)
2. La o expoziție sunt prezentate mai multe tipuri de flori, fiecare având un anumit cod. Zona de expunere este organizată pe **n** rânduri, fiecare având câte **n** parcele de dimensiune egală. Pe primul rând sunt expuse **n** tipuri de flori distincte, fiecare parcelă conținând flori de un anumit tip. Celelalte parcele sunt ocupate astfel:
- toate parcelele aflate în dreapta diagonalei secundare a zonei de expunere sunt ocupate de gazon;
 - fiecare parcelă, aflată pe diagonala secundară a zonei de expunere sau în stânga acestei diagonale, conține flori cu un cod de cel mult două cifre, obținut din ultimele două cifre ale sumei codurilor florilor aflate în două parcele de pe rândul anterior, și anume cea de pe aceeași coloană, respectiv cea de pe coloana din dreapta sa; dacă prima dintre cifrele codului obținut este 0, ea este ignorată, ca în exemplu.
- Scrieți un program C/C++ care citește de la tastatură un număr natural, **n** ($n \in [2, 20]$), apoi **n** numere naturale distincte din intervalul $[0, 10^2]$, reprezentând, în această ordine, de la stânga la dreapta, codurile florilor din parcelele aflate pe primul rând al zonei de expunere; gazonul are codul -1. Programul construiește în memorie un tablou bidimensional cu **n** linii și **n** coloane, în care fiecare linie corespunde unui rând de parcele și fiecare element al liniei corespunde unei parcele, în ordinea acestora, și memorează codul plantelor (flori sau gazon) plasate în acea parcelă.
Programul afișează pe ecran tabloul obținut, fiecare linie a tabloului pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.
- | | | | | |
|----|----|----|----|----|
| 90 | 10 | 4 | 27 | 30 |
| 0 | 14 | 31 | 57 | -1 |
| 14 | 45 | 88 | -1 | -1 |
| 59 | 33 | -1 | -1 | -1 |
| 92 | -1 | -1 | -1 | -1 |
- Exemplu:** dacă se citesc numerele **n=5**, apoi 90, 10, 4, 27, 30 se obține tabloul alăturat. (10p.)
3. Intervalul **[x, y]** se numește **p-interval** pentru un șir de valori întregi, dacă oricare dintre primii **p** termeni ai șirului aparține intervalului, iar numărul de valori întregi distincte din interval este minim.
Exemplu: pentru șirul 2, 7, -1, 8, 3, 10 există [2, 2] ca 1-interval, [-1, 8] ca 4-interval și 5-interval etc.
Fișierul **bac.in** conține un șir de cel mult 10^6 numere întregi din intervalul $[-10^9, 10^9]$, separate prin câte un spațiu.
Se cere să se afișeze pe ecran cea mai mică și cea mai mare valoare a lui **p** ($p \geq 2$) cu proprietatea că **(p-1)-interval** este identic cu **p-interval** pentru șirul aflat în fișier. Valorile afișate pot fi egale, iar dacă nu există nicio astfel de valoare, pe ecran se afișează mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.
Exemplu: dacă fișierul conține numerele 2 7 1 8 3 10 6 -3 -2 13, se afișează pe ecran 5 9 (intervale conform cerinței se obțin pentru valorile 5, 7 și 9 ale lui **p**), iar dacă fișierul conține numerele 2 7 1 0 8 10 -3 13, se afișează pe ecran **nu exista** (2p.)
a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)

Examenul național de bacalaureat 2023
Proba E. d)
Informatică

BAREM DE EVALUARE ȘI DE NOTARE
(comun pentru limbajele C/C++ și Pascal)

Variantă 6

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

- Se punctează orice modalitate de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit în barem. Nu se acordă fracțiuni de punct.
- Se acordă zece puncte din oficiu. Nota finală se calculează prin împărțirea la zece a punctajului total acordat pentru lucrare.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț (de exemplu tipuri întregi cu semn pentru memorarea numerelor naturale, dimensiune a tablourilor) este acceptată din punctul de vedere al corectitudinii programului, dacă acest lucru nu afectează funcționarea sa.

SUBIECTUL I (20 de puncte)

1c 2a 3d 4b 5c	5x4p.
----------------	-------

SUBIECTUL al II - lea (40 de puncte)

1.	a. Răspuns corect: 233223222	6p.	Se acordă numai 1p. dacă s-a scris doar prima cifră corect, numai 2p. dacă s-au scris doar primele trei cifre corect, numai 4p. dacă s-au scris doar primele șase cifre corect sau dacă, în plus față de cifrele cerute, s-au scris și alte caractere.
	b. Pentru răspuns corect	6p.	Se acordă câte 3p. pentru fiecare dintre cele două numere conform cerinței (oricare dintre numerele 6,7,8,9).
	c. Pentru program corect - variabile declarate, conform cerinței - date citite, conform cerinței - date afișate, conform cerinței - instrucțiuni repetitive, conform cerinței (*) - instrucțiune de decizie, conform cerinței - corectitudine globală a programului ¹⁾	10p. 1p. 1p. 1p. 4p. 2p. 1p.	(*) Se acordă numai 2p. dacă doar una dintre instrucțiunile repetitive este conform cerinței.
	d. Pentru algoritm pseudocod corect - structură repetitivă de tipul cerut utilizată (*) - aspecte specifice ale secvenței obținute prin înlocuire, conform cerinței (**) - algoritm complet, corectitudine globală a algoritmului ¹⁾	6p. 2p. 3p. 1p.	(*) Se acordă punctajul chiar dacă algoritmul obținut nu este echivalent cu cel dat. Se va puncta orice formă de structură repetitivă conform cerinței (cât timp...execută, while...do etc.). (**) Se acordă câte 1p. pentru fiecare aspect specific (inițializare a contorului, expresie logică pentru test inițial, actualizare a contorului) conform cerinței.
2.	Răspuns corect: 4, 5, 6, 7	6p.	Se acordă numai 3p. dacă s-au precizat conform cerinței doar descendenții direcți (fiii) nodului indicat.
3.	Pentru rezolvare corectă - prefixe determinate, conform cerinței (*) - afișare a unor șiruri - corectitudine globală a secvenței ¹⁾	6p. 4p. 1p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect specific (identificare a unui prefix, acces la prima literă a unui prefix, acces la ultima literă a unui prefix, prefixe suport determinate) conform cerinței.

SUBIECTUL al III - lea (30 de puncte)

1.	Pentru subprogram corect - antet al subprogramului, conform cerinței (*) - numere determinate, conform cerinței (**) - variabile locale declarate conform cerinței, corectitudine globală a subprogramului ¹⁾	10p. 3p. 6p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect al antetului (structură, parametru de intrare, parametri de ieșire) conform cerinței. (**) Se acordă numai 3p. dacă s-a determinat o pereche (x,p) cu proprietatea cerută, dar x nu este cel mai mic număr cu această proprietate.
----	--	----------------------------------	--

2.	Pentru program corect - variabilă de tip tablou bidimensional, declarată conform cerinței - date citite, conform cerinței - memorare a valorilor elementelor, conform cerinței (*) - date afișate în formatul indicat, conform cerinței - variabile simple, declarate conform cerinței, corectitudine globală a programului ¹⁾	10p. 1p. 1p. 6p. 1p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect specific (acces la cel puțin un element vecin pe linia anterioară, acces la cel puțin un element vecin pe coloana din dreapta, încadrare a elementelor în dimensiunile tablei, valori și elemente suport pe prima linie, valori și elemente suport aflate pe diagonala secundară sau în stânga acesteia, valori și elemente suport aflate în dreapta diagonalei secundare), conform cerinței.
3.	a. Pentru răspuns corect - descriere coerentă a algoritmului, conform cerinței (*) - elemente de eficiență justificate, conform cerinței b. Pentru program corect - operații cu fișiere: declarare, pregătire în vederea citirii, citire din fișier - valori determinate, conform cerinței (*),(**) - eficiență a algoritmului, conform cerinței (***) - variabile declarate, afișare a datelor conform cerinței, corectitudine globală a programului ¹⁾	2p. 1p. 1p. 8p. 1p. 5p. 1p. 1p.	(*) Se acordă punctajul chiar dacă algoritmul ales nu este eficient. (**) Se acordă numai 3p. dacă algoritmul este principal corect, dar nu oferă rezultatul cerut pentru toate seturile de date de intrare. (***) Se acordă punctajul numai pentru un algoritm liniar, care utilizează eficient memoria. 1p. O soluție posibilă determină, pe măsura citirii datelor, valorile minime și maxime curente (minc, maxc, inițializate cu 10^9+1 , respectiv cu -10^9-1) dintre cele citite, poziția curentă în șir (p), precum și valorile cerute pmin și pmax. Pentru fiecare valoare citită (x), p este incrementat; dacă $x \in [\text{minc}, \text{maxc}]$, atunci se actualizează, după caz, pmin (la prima apariție a unui astfel de caz) sau pmax (la fiecare apariție a unui astfel de caz), care primesc valoarea curentă a lui p; dacă $x \notin [\text{minc}, \text{maxc}]$, se actualizează, după caz, minc (dacă $x < \text{minc}$) sau maxc (dacă $x > \text{maxc}$). Valorile cerute sunt pmin și pmax.

¹⁾ Corectitudinea globală vizează structura, sintaxa, alte aspecte neprecizate în barem.