

Examenul național de bacalaureat 2024

Proba E. d)
INFORMATICĂ
Limbaajul C/C++

Varianta 3

Filieră teoretică, profil real, specializare științe ale naturii

- 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.

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++ care are valoarea 1 dacă și numai dacă numerele memorate în variabilele întregi **x** și **y** sunt pare.
 - x%2==0 && (y+1)%2!=0**
 - (x-y)%2==0**
 - (x+y)%2==0**
 - x%2==y%2**
- Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea **x=16**, se aplică metoda căutării binare, iar succesiunea de elemente a căror valoare se compară cu **x** pe parcursul aplicării metodei este **14, 24, 16**. Indicați succesiunea de valori care pot fi, în această ordine, elementele tabloului.
 - (2, 14, 7, 24, 12, 16, 48)**
 - (14, 24, 16, 14, 24, 16)**
 - (4, 8, 9, 14, 16, 24, 48)**
 - (14, 14, 24, 24, 16, 16)**
- Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul memorat în variabila întregă **x** aparține mulțimii **{-3, -1, 1, 3}**.
 - abs(x)>3 || x!=0**
 - abs(x-3)<=3 || abs(x%2)==0**
 - abs(x-3)>0**
 - abs(x)<=3 && abs(x%2)==1**
- În secvența alăturată, toate variabilele sunt de tip întreg, iar **x** memorează inițial un număr din intervalul **[10, 10⁵]**. Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila **u** să memoreze ultima cifră a produsului cifrelor lui **x**.

```
u=1;
while(x!=0)
{
    u=.....;
    x=x/10;
}
```

4	4	4	4	2	2	2	2	2
4	4	4	2	2	2	2	2	2
4	4	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	4
2	2	2	2	2	2	2	4	4
2	2	2	2	2	2	4	4	4
2	2	2	2	2	4	4	4	4

 - u*(x%10)**
 - (u*x)%10**
 - u%10*x**
 - (u%10)*(x%10)**
- În secvența de instrucțiuni de mai jos toate variabilele sunt de tip întreg.

```
for(i=0;i<9;i++)
{
    for(j=0;j<9;j++)
        if(.....)
            cout<<"4 "; | printf("4 ");
        else
            cout<<"2 "; | printf("2 ");
    cout<<endl; | printf("\n");
}
```

4	4	4	4	2	2	2	2	2
4	4	4	2	2	2	2	2	2
4	4	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	4
2	2	2	2	2	2	2	4	4
2	2	2	2	2	2	4	4	4
2	2	2	2	2	4	4	4	4

Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran valorile din figura de mai sus, în această ordine.

 - i-j>4 || i+j<12**
 - i-j>4 && i+j<12**
 - i+j<4 || i+j>12**
 - i+j>4 && i+j>12**

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

S-a notat cu **[c]** partea întreagă a numărului real **c**.

- Scriveți valoarea afișată dacă se citește, în această ordine, numerele 5, 15, 27, 10, 1, 17. **(6p.)**
- Dacă pentru **n** se citește valoarea 2, scrieți un set de numere distincte din intervalul $[0, 10^3]$ care pot fi citite în continuare, astfel încât, în urma executării algoritmului, să se afișeze valoarea 4. **(6p.)**
- Scriveți programul C/C++ corespunzător algoritmului dat. **(10p.)**
- Scriveți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **pentru...execută** cu o structură repetitivă de tip **cât timp...execută**. **(6p.)**

```

citește n
    (număr natural nenul)
p ← 1
pentru i ← 1, n execută
    citește x
    (număr natural)
    repetă
        x ← [x/3]
    până când x ≤ 3
    dacă x ≠ 0 atunci
        p ← p * x
scrie p
    
```

- Tablourile unidimensionale **A** și **B** au elementele: **A = (8, 4, 2)** și **B = (3, 8, 10, 17)**. Scrieți elementele tabloului obținut prin interclasarea în ordine crescătoare a tablourilor **A** și **B**, în ordinea în care ele apar în acesta. **(6p.)**
- Variabila întreagă **pret** memorează prețul unui sortiment de ciocolată exprimat în lei (număr natural), iar variabila **tip** memorează litera corespunzătoare tipului acesteia: litera **N** pentru **ciocolata neagra** sau litera **L** pentru **ciocolata cu lapte**.
Declarați variabila **tip** și scrieți o secvență de instrucțiuni C/C++ în urma executării căreia se afișează pe ecran tipul ciocolatei și, pe linia următoare, mesajul **scumpa**, dacă prețul acesteia este strict mai mare decât 10 lei, sau mesajul **ieftina** în caz contrar.
Exemplu: dacă variabila **pret** memorează valoarea 16, iar variabila **tip** memorează litera **N**, se afișează pe ecran
ciocolata neagra
scumpa **(6p.)**

SUBIECTUL al III-lea

(30 de puncte)

- Un număr natural se numește **major impar** dacă suma divizorilor săi proprii impari este strict mai mare decât suma divizorilor săi proprii pari. Divizorii proprii ai unui număr sunt divizorii săi naturali diferiți de 1 și de el însuși.
Exemplu: 18 este număr major impar (divizorii săi proprii pari sunt 2, 6, cei impari 3, 9, iar $3+9 > 2+6$).
Se citește două numere naturale, **a** și **b** ($2 \leq a \leq b \leq 10^4$) și se cere să se scrie cel mai mic număr major impar din intervalul **[a, b]**, sau valoarea 0, dacă în interval nu există un astfel de număr. Scrieți în pseudocod algoritmul corespunzător.
Exemplu: dacă **a=16**, **b=30**, atunci se scrie 18. **(10p.)**
- Scriveți un program C/C++ care citește de la tastatură un număr natural **n** ($n \in [2, 50]$) și cele **n** elemente ale unui tablou unidimensional, numere întregi din intervalul $[10, 10^4]$. Programul afișează pe ecran mesajul **DA** și un număr natural **k**, separate printr-un spațiu, dacă toate valorile memorate în tablou au câte **k** cifre, sau mesajul **NU** în cazul în care nu toate valorile au același număr de cifre.
Exemplu: pentru **n=6** și tabloul (123, 241, 896, 908, 100, 536)
se afișează pe ecran **DA 3**
iar pentru **n=6** și tabloul (123, 21, 896, 908, 100, 1536)
se afișează pe ecran **NU** **(10p.)**
- De-a lungul unui traseu montan este utilizată o succesiune de marcaje turistice, care trebuie urmate în acea ordine. Pentru fiecare marcaj se cunoaște cota (înălțimea, măsurată în metri) la care este plasat. Numim **scară** într-un traseu o secvență de marcaje aflate pe poziții consecutive în cadrul traseului, care au drept cote numere consecutive, ordonate strict crescător. O scară este formată din cel puțin două marcaje, iar lungimea acesteia este egală cu numărul de marcaje care o compun.
Fișierul **bac.txt** conține un șir de cel mult 10^6 numere naturale din intervalul $[10, 10^4]$, separate prin câte un spațiu, reprezentând cotele marcajelor turistice din cadrul unui traseu, în ordinea în care se succed în acesta. Se cere să se afișeze pe ecran lungimea maximă a unei scări pe acest traseu. Dacă în cadrul traseului nu există nicio scară, pe ecran se afișează mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.
Exemplu: dacă fișierul conține numerele 500 600 601 405 569 570 700 701 625 626 627 520
atunci pe ecran se afișează 3
 - Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
 - Scriveți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**

Examenul național de bacalaureat 2024

Proba E. d)
INFORMATICĂ
Limbaajul Pascal

Varianta 3

Filieră teoretică, profil real, specializare științe ale naturii

- 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.

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 Pascal care are valoarea **true** dacă și numai dacă numerele memorate în variabilele întregi **x** și **y** sunt pare.
 - (x mod 2=0) and ((y+1) mod 2<>0)**
 - (x-y) mod 2=0**
 - (x+y) mod 2=0**
 - x mod 2=y mod 2**
- Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea **x=16**, se aplică metoda căutării binare, iar succesiunea de elemente a căror valoare se compară cu **x** pe parcursul aplicării metodei este **14, 24, 16**. Indicați succesiunea de valori care pot fi, în această ordine, elementele tabloului.
 - (2, 14, 7, 24, 12, 16, 48)**
 - (14, 24, 16, 14, 24, 16)**
 - (4, 8, 9, 14, 16, 24, 48)**
 - (14, 14, 24, 24, 16, 16)**
- Indicați o expresie Pascal care are valoarea **true** dacă și numai dacă numărul memorat în variabila întregă **x** aparține mulțimii **{-3, -1, 1, 3}**.
 - (abs(x)>3) or (x<>0)**
 - (abs(x-3)<=3) or (abs(x mod 2)=0)**
 - abs(x-3)>0**
 - (abs(x)<=3) and (abs(x mod 2)=1)**
- În secvența alăturată, toate variabilele sunt de tip întreg, iar **x** memorează inițial un număr din intervalul **[10, 10⁵]**. Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila **u** să memoreze ultima cifră a produsului cifrelor lui **x**.

```
u:=1;
while x<>0 do
begin u:=.....;
      x:=x div 10
end;
```

4	4	4	4	2	2	2	2	2
4	4	4	2	2	2	2	2	2
4	4	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	4
2	2	2	2	2	2	2	4	4
2	2	2	2	2	2	4	4	4
2	2	2	2	2	4	4	4	4

 - u*(x mod 10)**
 - (u*x) mod 10**
 - u mod 10*x**
 - (u mod 10)*(x mod 10)**
- În secvența de instrucțiuni de mai jos toate variabilele sunt de tip întreg.

```
for i:=0 to 8 do
begin for j:=0 to 8 do
    if ..... then
        write('4 ')
    else
        write('2 ');
    writeln
end;
```

4	4	4	4	2	2	2	2	2
4	4	4	2	2	2	2	2	2
4	4	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	4
2	2	2	2	2	2	2	4	4
2	2	2	2	2	2	4	4	4
2	2	2	2	2	4	4	4	4

Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran valorile din figura de mai sus, în această ordine.

 - (i-j>4) or (i+j<12)**
 - (i-j>4) and (i+j<12)**
 - (i+j<4) or (i+j>12)**
 - (i+j>4) and (i+j<12)**

SUBIECTUL al II-lea**(40 de puncte)****1. Algoritmul alăturat este reprezentat în pseudocod.**

S-a notat cu $[c]$ partea întreagă a numărului real c .

- a. Scrieți valoarea afișată dacă se citește, în această ordine, numerele 5, 15, 27, 10, 1, 17. (6p.)
- b. Dacă pentru n se citește valoarea 2, scrieți un set de numere distincte din intervalul $[0, 10^3]$ care pot fi citite în continuare, astfel încât, în urma executării algoritmului, să se afișeze valoarea 4. (6p.)
- c. Scrieți programul Pascal corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **pentru...execută** cu o structură repetitivă de tip **cât timp...execută**. (6p.)

```
citește n  
    (număr natural nenul)  
p ← 1  
pentru i ← 1, n execută  
    citește x  
    (număr natural)  
    repetă  
        x ← [x/3]  
    până când x ≤ 3  
    dacă x ≠ 0 atunci  
        p ← p * x  
scrie p
```

2. Tablourile unidimensionale **A** și **B** au elementele: **A** = (8, 4, 2) și **B** = (3, 8, 10, 17). Scrieți elementele tabloului obținut prin interclasarea în ordine crescătoare a tablourilor **A** și **B**, în ordinea în care ele apar în acesta. (6p.)
3. Variabila întreagă **pret** memorează prețul unui sortiment de ciocolată exprimat în lei (număr natural), iar variabila **tip** memorează litera corespunzătoare tipului acesteia: litera **N** pentru **ciocolata neagra** sau litera **L** pentru **ciocolata cu lapte**.
Declarați variabila **tip** și scrieți o secvență de instrucțiuni Pascal în urma executării căreia se afișează pe ecran tipul ciocolatei și, pe linia următoare, mesajul **scumpa**, dacă prețul acesteia este strict mai mare decât 10 lei, sau mesajul **ieftina** în caz contrar.
Exemplu: dacă variabila **pret** memorează valoarea 16, iar variabila **tip** memorează litera **N**, se afișează pe ecran
ciocolata neagra
scumpa (6p.)

SUBIECTUL al III-lea**(30 de puncte)**

1. Un număr natural se numește **major impar** dacă suma divizorilor săi proprii impari este strict mai mare decât suma divizorilor săi proprii pari. Divizorii proprii ai unui număr sunt divizorii săi naturali diferiți de 1 și de el însuși.
Exemplu: 18 este număr major impar (divizorii săi proprii pari sunt 2, 6, cei impari 3, 9, iar $3+9 > 2+6$).
Se citește două numere naturale, **a** și **b** ($2 \leq a \leq b \leq 10^4$) și se cere să se scrie cel mai mic număr major impar din intervalul $[a, b]$, sau valoarea 0, dacă în interval nu există un astfel de număr. Scrieți în pseudocod algoritmul corespunzător.
Exemplu: dacă **a**=16, **b**=30, atunci se scrie 18. (10p.)
2. Scrieți un program Pascal care citește de la tastatură un număr natural **n** ($n \in [2, 50]$) și cele **n** elemente ale unui tablou unidimensional, numere întregi din intervalul $[10, 10^4]$. Programul afișează pe ecran mesajul **DA** și un număr natural **k**, separate printr-un spațiu, dacă toate valorile memorate în tablou au câte **k** cifre, sau mesajul **NU** în cazul în care nu toate valorile au același număr de cifre.
Exemplu: pentru **n**=6 și tabloul (123, 241, 896, 908, 100, 536)
se afișează pe ecran **DA 3**
iar pentru **n**=6 și tabloul (123, 21, 896, 908, 100, 1536)
se afișează pe ecran **NU** (10p.)
3. De-a lungul unui traseu montan este utilizată o succesiune de marcaje turistice, care trebuie urmate în acea ordine. Pentru fiecare marcaj se cunoaște cota (înălțimea, măsurată în metri) la care este plasat. Numim **scară** într-un traseu o secvență de marcaje aflate pe poziții consecutive în cadrul traseului, care au drept cote numere consecutive, ordonate strict crescător. O scară este formată din cel puțin două marcaje, iar lungimea acesteia este egală cu numărul de marcaje care o compun.
Fișierul **bac.txt** conține un șir de cel mult 10^6 numere naturale din intervalul $[10, 10^4]$, separate prin câte un spațiu, reprezentând cotele marcajelor turistice din cadrul unui traseu, în ordinea în care se succed în acesta. Se cere să se afișeze pe ecran lungimea maximă a unei scări pe acest traseu. Dacă în cadrul traseului nu există nicio scară, pe ecran se afișează mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.
Exemplu: dacă fișierul conține numerele 500 600 601 405 569 570 700 701 625 626 627 520
atunci pe ecran se afișează 3
a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul Pascal corespunzător algoritmului proiectat. (8p.)

Examenul național de bacalaureat 2024

**Proba E. d)
INFORMATICĂ**

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

Varianța 3

Filieră teoretică, profil real, specializare științe ale naturii

- 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)

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

SUBIECTUL al II - lea (40 de puncte)

1.	a. Răspuns corect: 9	6p.	
	b. Pentru răspuns corect	6p.	Se acordă câte 2p. pentru fiecare aspect specific (număr de valori scrise, prima valoare, a doua valoare) conform cerinței. Se punctează valori naturale din reuniunea intervalelor de forma $[2 \cdot 3^k, 3^{k+1})$, cu k număr natural, $k \in \{1, 2, 3, 4, 5\}$.
	c. Pentru program corect -declaraire a variabilelor -citire a datelor -afișare a datelor -instrucțiune de decizie -instrucțiuni repetitive (*) -atribuiri -corectitudine globală a programului ¹⁾	10p. 1p. 1p. 1p. 2p. 3p. 1p. 1p.	(*) Se acordă numai 2p. dacă doar una dintre instrucțiunile repetitive este conform cerinței.
	d. Pentru algoritm pseudocod corect -utilizare a unei structuri repetitive de tipul indicat (*) -aspecte specifice ale secvenței obținute prin înlocuire, conform cerinței (**) -algoritm complet, corectitudine globală a algoritmului ¹⁾	6p. 2p. 3p. 1p.	(*) Se acordă punctajul pentru orice formă de structură repetitivă de tipul cerut (cât timp ... execută, while ... do etc.). (**) Se acordă câte 1p. pentru fiecare aspect specific (inițializare a contorului, expresie de continuare, actualizare a contorului) conform cerinței.
2.	Pentru rezolvare corectă	6p.	Se acordă câte 2p. pentru fiecare aspect specific (număr de elemente, ordine, valori suport) conform cerinței.
3.	Pentru rezolvare corectă -declaraire a variabilei -afișare a datelor în formatul cerut (*) -corectitudine globală a secvenței ¹⁾	6p. 1p. 4p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect specific (corelare preț-mesaj, identificare a tipului utilizând constante de tip caracter, ordine a valorilor afișate, linii separate) conform cerinței.

SUBIECTUL al III - lea (30 de puncte)

1.	Pentru algoritm corect -citire a datelor -determinare a valorii cerute (*) -scriere a datelor și tratare a cazului 0 -scriere principal corectă a structurilor de control, corectitudine globală a algoritmului ¹⁾ (**)	10p. 1p. 6p. 1p. 2p.	(*) Se acordă câte 1p. pentru fiecare aspect specific (identificare a unui divizor par/impar, algoritm de bază pentru suma unei serii de valori, divizori suport pentru fiecare sumă, identificare a unui număr major impar, algoritm de bază pentru identificarea primului/celui mai mic număr cu o anumită proprietate dintr-o serie, numere suport verificate pentru determinarea numărului major impar cerut) conform cerinței. (**) Se va puncta orice formă principal corectă de structură repetitivă sau decizională.
----	---	---	---

Probă scrisă la INFORMATICĂ

Varianța 3

Filieră teoretică, profil real, specializare științe ale naturii

Barem de evaluare și de notare

2. Pentru program corect -declarare a unei variabile care să permită memorarea unui tablou unidimensional, conform cerinței -citire a datelor -verificare a proprietății cerute (*) -afișare a datelor în formatul cerut -declarare a variabilelor simple, corectitudine globală a programului¹⁾	10p. 1p. 1p. 6p. 1p. 1p.	(*) Se acordă câte 2p. pentru fiecare aspect specific (algoritm principal corect de verificare a unei proprietăți, determinare a numărului de cifre ale unui element, elemente suport verificate) conform cerinței.
3. a. Pentru răspuns corect -coerență a descrierii algoritmului (*) -justificare a elementelor de eficiență	2p. 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.
b. Pentru program corect -operații cu fișiere: declarare, pregătire în vederea citirii, citire din fișier -determinare a valorii cerute (*),(**) -utilizare a unui algoritm eficient (***) -declarare a variabilelor, afișare a datelor și tratare a cazului nu exista, corectitudine globală a programului¹⁾	8p. 1p. 5p. 1p. 1p.	(***) Se acordă punctajul numai pentru un algoritm liniar (de complexitate $O(n)$), care utilizează eficient memoria. O soluție posibilă parcurge o dată fișierul, memorând pentru scara curentă și pentru cea căutată lungimea ($lgCrt$, respectiv $lgMax$, inițial egale cu 0) și ultimul termen ($ultCrt$, inițializat cu 0) al scării curente; dacă numărul x citit din fișier la pasul curent face parte din scara curentă ($x=ultCrt+1$) se actualizează $lgCrt$, altfel se actualizează datele privind o nouă scară, deci $lgCrt=1$. La fiecare pas se actualizează $ultCrt=x$ și, după caz, $lgMax$. Se afișează $lgMax$, iar mesajul se afișează dacă $lgMax<2$.

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