

Examenul național de bacalaureat 2025

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

Variantă 1

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.

- Indicați expresia C/C++ cu aceeași valoare ca a expresiei alăturate. 2025%2019+6
a. 2025/2020+5 b. 2025/2021+8 c. 2025%2020+5 d. 2025%2021+8
- Subprogramul **f** este definit alăturat. Indicați ce se afișează în urma apelului de mai jos. **f(3);**

```
void f(int n)
{
    int i;
    for(i=1; i<=n; i++) if(i%2==0) { cout<<i; | printf("%d", i);
                                f(i-1);
                                }
                                else { f(i-1);
                                cout<<i; | printf("%d", i);
                                }
}
```


a. 1211213 b. 123121 c. 123 d. 01201012013
- Indicați o declarație a unui tablou bidimensional **m**, care poate memora maximum 100 de numere reale.
a. **float m[2,50];** b. **float m[4][25];**
c. **float m[10] x float m[10];** d. **int m[100];**
- Utilizând metoda backtracking, s-au generat toate codurile posibile pentru deblocarea unor telefoane, coduri de câte 6 cifre distincte, din mulțimea cifrelor, ordonată crescător. Fiecare cod are primele trei cifre impare și ultimele trei cifre pare. Primele patru coduri sunt 135024, 135026, 135028, 135042. Indicați penultimul cod generat.
a. 957862 b. 957846 c. 975862 d. 975846
- Un graf orientat fără circuite are 10 vârfuri. Indicați numărul maxim de arce ale grafului.
a. 10 b. 45 c. 50 d. 90

SUBIECTUL al II-lea

(40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.**
a. Scrieți ce se afișează în urma executării algoritmului dacă se citește, în această ordine, numerele 7 și 17. (6p.)
b. Dacă pentru variabila **n** se citește valoarea 25, scrieți două numere distincte care pot fi citite pentru variabila **m**, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze valoarea 25. (6p.)
c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind structura **cât timp...execută** cu o structură repetitivă cu test final. (6p.)
- ```
citește m, n
(numele naturale nenule, m ≤ n)
nr ← 0; i ← m
repetă
 x ← 1
 cât timp x * x < i execută
 x ← x + 1
 dacă x * x = i atunci nr ← i
 altfel i ← i + 1
până când i > n sau nr ≠ 0
scrie nr
```

2. Un graf neorientat cu 6 noduri, numerotate de la 1 la 6, este reprezentat prin listele de adiacență alăturate. Scrieți mulțimea nodurilor și mulțimea muchiilor unui subgraf al acestuia, fără noduri izolate, care să fie graf eulerian. (6p.)
- |    |            |
|----|------------|
| 1: | 2, 3, 4, 6 |
| 2: | 1, 3, 5    |
| 3: | 1, 2, 5    |
| 4: | 1, 5, 6    |
| 5: | 2, 3, 4, 6 |
| 6: | 1, 4, 5    |
3. Variabila **p** memorează simultan, pentru un tip de prăjitură, codul (un număr natural de două cifre), prețul (număr real) și un set de trei numere naturale din intervalul  $[1, 10^2]$ , reprezentând informații specifice, în această ordine: tipul glazurii, tipul cremei principale și numărul de blaturi. Știind că expresiile C/C++ de mai jos au ca valori codul, prețul, respectiv tipul glazurii pentru o prăjitură, scrieți definiția unei structuri cu eticheta **prajitura**, care permite memorarea datelor despre o prăjitură, și declarați corespunzător variabila **p**. (6p.)
- p.cod    p.pret    p.informatii[0]**

### SUBIECTUL al III-lea

(30 de puncte)

1. Numărul natural **an** este **ascendent** al numărului natural **n**, dacă oricare dintre cifrele lui **an** este mai mare sau egală cu cifra unităților lui **n**.  
**Exemplu:** oricare dintre numerele 7, 9, 98 sau 7998 este ascendent al lui 827, dar numărul 857 nu este ascendent al lui 827.  
Subprogramul **ascendent** are trei parametri:
- **n**, prin care primește un număr natural ( $n \in [0, 10^3]$ );
  - **x** și **y**, prin care primește câte un număr natural din intervalul  $[0, 10^3]$  ( $x < y$ ).
- Subprogramul returnează suma ascendenților lui **n** din intervalul  $[x, y]$ , sau valoarea 0, dacă nu există niciun astfel de ascendent. Scrieți definiția completă a subprogramului C/C++.
- Exemplu:** dacă **n=827**, **x=9**, **y=800**, subprogramul returnează 7893 ( $9+77+78+79+87+88+89+97+98+99+777+778+779+787+788+789+797+798+799=7893$ ). (10p.)
2. Un **cuvânt semioglindit** se obține dintr-un cuvânt cu  $2 \cdot k$  ( $k \in [1, 10^2]$ ) litere, prin interschimbarea în acesta a secvenței formate din primele **k** litere cu secvența formată din ultimele **k** litere.  
**Exemplu:** din cuvântul platim se obține cuvântul semioglindit timpla.  
Într-un text de cel mult 200 de caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat, pe care îl transformă în memorie, prin înlocuirea fiecărui cuvânt cu număr par de litere, cu cel semioglindit obținut din acesta, ca în exemplu. Programul afișează pe ecran textul obținut, sau mesajul **nu exista**, dacă toate cuvintele au număr impar de litere.  
**Exemplu:** pentru textul am facut fotografii unei flori mari se afișează pe ecran textul ma facut rafiifotog eiun flori rima (10p.)
3. Un tânăr pasionat de călătorii are o listă cu muzee virtuale și, pentru fiecare, câte un singur interval orar, în care acesta poate fi vizitat online, gratuit. Tânărul dispune zilnic de același interval orar pentru vizite; un muzeu este **convenabil** dacă poate fi vizitat online gratuit în timpul disponibil și dacă pentru vizită îi poate aloca cel puțin o oră. Muzeele din listă sunt numerotate cu valori naturale consecutive, începând cu 1, și cel puțin unul este convenabil.  
Fișierul text **bac.in** conține cel mult  $10^5$  linii, iar pe fiecare linie câte o pereche de numere, reprezentând limitele câte unui interval orar: pe prima linie intervalul orar de care tânărul dispune zilnic, iar pe fiecare dintre următoarele linii, intervalul orar de vizitare gratuită pentru câte un muzeu, în ordinea din listă. Limitele intervalelor sunt ore fixe, numere naturale din intervalul  $[8, 22]$ , iar cele aflate pe aceeași linie a fișierului sunt în ordine strict crescătoare și sunt separate printr-un spațiu.  
Se cere să se afișeze pe ecran, separate printr-un spațiu, două valori, reprezentând numărul de muzee convenabile, respectiv numărul de ordine al ultimului astfel de muzeu din lista tânărului. Utilizați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.  
**Exemplu:** dacă fișierul conține valorile alăturate, atunci pe ecran se afișează numerele 3 4.  
(pot fi vizitate trei muzee cu numerele de ordine 1, 2 și 4, în intervalele 16–18, 17–19, respectiv 18–20). (2p.)
- |    |    |
|----|----|
| 16 | 19 |
| 15 | 18 |
| 17 | 21 |
| 19 | 21 |
| 18 | 20 |
| 12 | 13 |
- a.** Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)  
**b.** Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)

Examenul național de bacalaureat 2025  
Proba E. d)  
INFORMATICĂ  
Limbajul Pascal

Variantă 1

*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.

- Indicați expresia Pascal cu aceeași valoare ca a expresiei alăturate. 2025 mod 2019+6  
a. 2025 div 2020+5      b. 2025 div 2021+8      c. 2025 mod 2020+5      d. 2025 mod 2021+8
- Subprogramul **f** este definit alăturat. Indicați ce se afișează în urma apelului de mai jos. **f(3)**;  

```
procedure f(n:integer);
var i:integer;
begin
 for i:=1 to n do if i mod 2=0 then
 begin write(i); f(i-1) end
 else
 begin f(i-1); write(i) end
end;
```

  
a. 1211213      b. 123121      c. 123      d. 01201012013
- Indicați o declarație a unui tablou bidimensional **m**, care poate memora maximum 100 de numere reale.  
a. **m:array[2][1..50] of real;**      b. **m:array[1..4,1..25] of real;**  
c. **m:array[1..10] x [1..10] of real;**      d. **m:array[1..100] of integer;**
- Utilizând metoda backtracking, s-au generat toate codurile posibile pentru deblocarea unor telefoane, coduri de câte 6 cifre distincte, din mulțimea cifrelor, ordonată crescător. Fiecare cod are primele trei cifre impare și ultimele trei cifre pare. Primele patru coduri sunt 135024, 135026, 135028, 135042. Indicați penultimul cod generat.  
a. 957862      b. 957846      c. 975862      d. 975846
- Un graf orientat fără circuite are 10 vârfuri. Indicați numărul maxim de arce ale grafului.  
a. 10      b. 45      c. 50      d. 90

**SUBIECTUL al II-lea** (40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.  
a. Scrieți ce se afișează în urma executării algoritmului dacă se citesc, în această ordine, numerele 7 și 17. (6p.)  
b. Dacă pentru variabila **n** se citește valoarea 25, scrieți două numere distincte care pot fi citite pentru variabila **m**, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze valoarea 25. (6p.)  
c. Scrieți programul Pascal corespunzător algoritmului dat. (10p.)  
d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind structura **cât timp...execută** cu o structură repetitivă cu test final. (6p.)
- ```
citește m,n  
(numere naturale nenule, m≤n)  
nr←0; i←m  
repetă  
    x←1  
    cât timp x*x<i execută  
        x←x+1  
    ■  
    dacă x*x=i atunci nr←i  
    altfel i←i+1  
    ■  
    până când i>n sau nr≠0  
scrie nr
```

2. Un graf neorientat cu 6 noduri, numerotate de la 1 la 6, este reprezentat prin listele de adiacență alăturate. Scrieți mulțimea nodurilor și mulțimea muchiilor unui subgraf al acestuia, fără noduri izolate, care să fie graf eulerian. (6p.)
- | | |
|----|------------|
| 1: | 2, 3, 4, 6 |
| 2: | 1, 3, 5 |
| 3: | 1, 2, 5 |
| 4: | 1, 5, 6 |
| 5: | 2, 3, 4, 6 |
| 6: | 1, 4, 5 |
3. Variabila **p** memorează simultan, pentru un tip de prăjitură, codul (un număr natural de două cifre), prețul (număr real) și un set de trei numere naturale din intervalul $[1, 10^2]$, reprezentând informații specifice, în această ordine: tipul glazurii, tipul cremei principale și numărul de blaturi. Știind că expresiile Pascal de mai jos au ca valori codul, prețul, respectiv tipul glazurii pentru o prăjitură, scrieți definiția unui tip de date, cu numele **prajitura**, înregistrare care permite memorarea datelor despre o prăjitură, și declarați corespunzător variabila **p**. (6p.)
- p.cod p.pret p.informatii[0]**

SUBIECTUL al III-lea

(30 de puncte)

1. Numărul natural **an** este **ascendent** al numărului natural **n**, dacă oricare dintre cifrele lui **an** este mai mare sau egală cu cifra unităților lui **n**.
Exemplu: oricare dintre numerele 7, 9, 98 sau 7998 este ascendent al lui 827, dar numărul 857 nu este ascendent al lui 827.
Subprogramul **ascendent** are trei parametri:
- **n**, prin care primește un număr natural ($n \in [0, 10^3]$);
 - **x** și **y**, prin care primește câte un număr natural din intervalul $[0, 10^3]$ ($x < y$).
- Subprogramul returnează suma ascendenților lui **n** din intervalul $[x, y]$, sau valoarea 0, dacă nu există niciun astfel de ascendent. Scrieți definiția completă a subprogramului Pascal.
Exemplu: dacă **n=827**, **x=9**, **y=800**, subprogramul returnează 7893 ($9+77+78+79+87+88+89+97+98+99+777+778+779+787+788+789+797+798+799=7893$). (10p.)
2. Un **cuvânt semioglindit** se obține dintr-un cuvânt cu $2 \cdot k$ ($k \in [1, 10^2]$) litere, prin interschimbarea în acesta a secvenței formate din primele **k** litere cu secvența formată din ultimele **k** litere.
Exemplu: din cuvântul platim se obține cuvântul semioglindit timpla.
Într-un text de cel mult 200 de caractere, cuvintele sunt formate din litere mici ale alfabetului englez și sunt separate prin câte un spațiu. Scrieți un program Pascal care citește de la tastatură un text de tipul precizat, pe care îl transformă în memorie, prin înlocuirea fiecărui cuvânt cu număr par de litere, cu cel semioglindit obținut din acesta, ca în exemplu. Programul afișează pe ecran textul obținut, sau mesajul **nu exista**, dacă toate cuvintele au număr impar de litere.
Exemplu: pentru textul am facut fotografii unei flori mari se afișează pe ecran textul ma facut rafiifotog eiun flori rima (10p.)
3. Un tânăr pasionat de călătorii are o listă cu muzee virtuale și, pentru fiecare, câte un singur interval orar, în care acesta poate fi vizitat online, gratuit. Tânărul dispune zilnic de același interval orar pentru vizite; un muzeu este **convenabil** dacă poate fi vizitat online gratuit în timpul disponibil și dacă pentru vizită îi poate aloca cel puțin o oră. Muzeele din listă sunt numerotate cu valori naturale consecutive, începând cu 1, și cel puțin unul este convenabil.
Fișierul text **bac.in** conține cel mult 10^5 linii, iar pe fiecare linie câte o pereche de numere, reprezentând limitele câte unui interval orar: pe prima linie intervalul orar de care tânărul dispune zilnic, iar pe fiecare dintre următoarele linii, intervalul orar de vizitare gratuită pentru câte un muzeu, în ordinea din listă. Limitele intervalelor sunt ore fixe, numere naturale din intervalul $[8, 22]$, iar cele aflate pe aceeași linie a fișierului sunt în ordine strict crescătoare și sunt separate printr-un spațiu.
Se cere să se afișeze pe ecran, separate printr-un spațiu, două valori, reprezentând numărul de muzee convenabile, respectiv numărul de ordine al ultimului astfel de muzeu din lista tânărului. Utilizați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.
Exemplu: dacă fișierul conține valorile alăturate, atunci pe ecran se afișează numerele 3 4.
(pot fi vizitate trei muzee cu numerele de ordine 1, 2 și 4, în intervalele 16–18, 17–19, respectiv 18–19). (2p.)
- | | |
|----|----|
| 16 | 19 |
| 15 | 18 |
| 17 | 21 |
| 19 | 21 |
| 18 | 20 |
| 12 | 13 |
- a.** Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul Pascal corespunzător algoritmului proiectat. (8p.)

Examenul național de bacalaureat 2025

Proba E. d)

Informatică

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

Varianta 1

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)

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

SUBIECTUL al II - lea (40 de puncte)

1.	a. Răspuns corect: 9	6p.	
	b. Pentru răspuns corect	6p.	Se acordă câte 3p. pentru fiecare dintre cele două numere conform cerinței (oricare două numere naturale din intervalul [17, 25]).
	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 cu test final, înlocuind-o pe cea indicată (*) -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. 2p. Se punctează orice formă de structură repetitivă explicită, conform cerinței (repetă...până când, execută... până când, execută...cât timp, do...while etc.). 3p. (***) Se acordă numai 2p. dacă doar un aspect (echivalență a algoritmului pentru cazul inițial $x \geq i$, expresie logică pentru test final) este conform cerinței.
2.	Pentru răspuns corect	6p.	Se acordă câte 2p. pentru fiecare aspect (mulțime a nodurilor unui subgraf fără noduri izolate, mulțime a muchiilor unui subgraf corespunzător mulțimii nodurilor alese, graf eulerian) conform cerinței.
3.	Răspuns corect -definire a structurii/tipului cerut (*) -declaraire a variabilei conform cerinței -corectitudine globală a secvenței ¹⁾	6p. 4p. 1p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect (definire de bază a unei structuri/înregistrări, etichetă/nume, câmpuri de tip simplu – cod, pret, câmp de tip structurat –informatii) conform cerinței.

SUBIECTUL al III - lea

(30 de puncte)

1. Pentru subprogram corect -antet al subprogramului (*) -determinare a valorii cerute (**) -instrucțiune de returnare a rezultatului -declarare a tuturor variabilelor locale, corectitudine globală a subprogramului¹⁾	10p. (*) Se acordă câte 1p. pentru fiecare aspect al antetului (structură, parametri de intrare) conform cerinței. 2p. 6p. 1p. (**) Se acordă câte 1p. pentru fiecare aspect (identificare a cifrei unităților unui număr, identificare a unei cifre mai mici/mai mari sau egale cu o altă cifră, algoritm de bază pentru verificarea unei proprietăți, cifre suport verificate pentru determinarea unui ascendent, algoritm de bază pentru determinarea sumei unei serii de valori, numere ascendent suport însumate) conform cerinței. 1p.
2. Pentru program corect -declarare a unei variabile care să memoreze un șir de caractere, conform cerinței -citire a datelor, conform cerinței -transformare a șirului conform cerinței (*) -afișare a datelor, conform cerinței -declarare a variabilelor simple, corectitudine globală a programului¹⁾	10p. (*) Se acordă câte 1p. pentru fiecare aspect specific (identificare a unui cuvânt cu număr par/impar de litere, caractere suport obținute în prima jumătate a unui cuvânt, caractere suport obținute în a doua jumătate a unui cuvânt, cuvinte suport înlocuite/păstrate, transformare în memorie, tratare a cazului nu exista) conform cerinței. 1p. 1p. 6p. 1p.
3. a. Pentru răspuns corect -descriere coerentă a algoritmului (*) -justificare a elementelor de eficiență, conform cerinței b. Pentru program corect -operații cu fișiere: declarare, pregătire în vederea citirii, citire din fișier conform cerinței -determinare a valorilor cerute (*),(**) -utilizare a unui algoritm eficient (***) -declarare a variabilelor, afișare a datelor, corectitudine globală a programului¹⁾	2p. (*) Se acordă punctajul chiar dacă algoritmul ales nu este eficient. 1p. (**) Se acordă numai 3p. dacă algoritmul parcurge pașii necesari rezolvării, dar cu detalii care conduc la o rezolvare parțială. 1p. (***) Se acordă punctajul numai pentru un algoritm liniar, care utilizează eficient memoria. 8p. 1p. O soluție posibilă citește prima pereche din fișier (h1, h2), apoi, pe măsura citirii celorlalte perechi, se incrementează numărul de ordine al muzeului (nrCrt) și se calculează intersecția [v1,v2] dintre intervalul [h1,h2] și intervalul de timp determinat de perechea curentă [crt1,crt2], unde $v1 = \max(h1, crt1)$ și $v2 = \min(h2, crt2)$; dacă intersecția permite cel puțin o oră de vizită ($v1 < v2$) se incrementează contorul cerut (nrVizitate) și se actualizează numărul de ordine al muzeului cerut ($ultim = nrCrt$). 5p. 1p. 1p.

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