

Examenul național de bacalaureat 2026 – Simulare județeană
Proba E. d) - INFORMATICĂ – Limbajul C/C++
BAREM DE EVALUARE ȘI DE NOTARE

Varianta 2

Filieră teoretică, profil real, specializare matematică-informatică/matematică-informatică intensiv 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)

1. a	2. b	3. a	4. b	5. d	5 x 4p.
------	------	------	------	------	---------

SUBIECTUL al II-lea

(40 de puncte)

1.	a. Răspuns corect: 18	6p.	
	b. Răspuns corect: 106 și 998	6p.	Se acordă câte 3p. pentru fiecare răspuns corect
	c. Pentru program corect -declarare a variabilelor -citire a datelor și afișare a datelor -instrucțiuni de decizie (*) -instrucțiuni repetitive (**) -atribuiri -corectitudine globală a programului ¹⁾	10p 1p. 1p. 2p. 4p. 1p. 1p.	*) Se acordă numai 1p. dacă doar una dintre instrucțiunile de decizie este conform cerinței. **) 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 (pentru ... execută, for ... do etc.). **) Se acordă câte 1p. pentru fiecare aspect specific (expresie de inițializare a contorului, expresie pentru valoarea finală a contorului, doar actualizare implicită a contorului) conform cerinței.
2.	Răspuns corect: (10, 5, 3, 2), (9, 7, 3, 1)	6p.	Se acordă 3p. dacă doar unul din răspunsuri este corect.
3.	Pentru rezolvare corectă -expresie de accesare a unui element al tabloului -valori ale elementelor tabloului atribuite conform cerinței (*) -corectitudine globală a secvenței ¹⁾	6p. 1p. 4p. 1p.	(*) Se acordă câte 2p. pentru fiecare aspect specific (identificare a cel puțin unei relații între valoarea elementului și poziția acestuia în tablou, valori suport în corelare cu pozițiile elementelor utilizând numărul indicat de instrucțiuni de atribuire) conform cerinței. O soluție posibilă <code>if(i%2==1) a[i][j]=a[i-1][j]; //2p. else a[i][j]=i/2+j/2+1;</code>

SUBIECTUL al III-lea

(30 de puncte)

1.	<p>Pentru subprogram corect -antet al subprogramului (*) -determinare a valorii cerute (**) -instrucțiune/instrucțiuni de returnare a rezultatului și tratare a cazului 0 -declarare a tuturor variabilelor locale, corectitudine globală a subprogramului¹⁾</p>	<p>10p. 2p. 6p. 1p. 1p.</p>	<p>(*) Se acordă câte 1p. pentru fiecare aspect al antetului (structură, parametri de intrare) conform cerinței. (**) Se acordă câte 2p. pentru fiecare aspect specific (identificare a două numere din intervalul [1, N], verificarea dacă produsul numerelor este mai mare sau egal cu M, actualizarea valorii minime cerute) conform cerinței. Soluții posibile <pre>int prodFact(int N, int M) { int ans = 1e9; for(int i=1; i<=N; i++) { for(int j=i; j<=N; j++) { if(i*j>=M) ans=min(ans, i*j); } } if(ans==1e9) return 0; return ans; }</pre></p>
2.	<p>Pentru program corect -declarare a unei variabile care să memoreze un șir de caractere -citire a datelor -determinare a șirului cerut (*) -afișarea datelor -declarare a variabilelor simple, corectitudine globală a programului¹⁾</p>	<p>10p. 1p. 1p. 6p. 1p. 1p.</p>	<p>(*) Se acordă câte 1p. pentru fiecare aspect (identificare a unui cuvânt, identificare a unui cuvânt care trebuie prescurtat, identificarea primei litere a cuvântului, identificarea ultimei litere a cuvântului, înlocuirea unui subșir cu un alt subșir, cuvinte suport înlocuite/păstrate, construire în memorie ca șir de caractere) conform cerinței.</p>
3.	<p>a. Pentru răspuns corect -coerență a descrierii algoritmului (*) -justificare a elementelor de eficiență</p> <p>b. Pentru program corect -operații cu fișiere: declarare, pregătire în vederea citirii, citire din fișier -determinare a valorilor cerute (*),(**) -utilizare a unui algoritm eficient (***) -declarare a variabilelor, afișare a datelor, corectitudine globală a programului¹⁾</p>	<p>2p. 1p. 1p.</p> <p>8p. 1p. 5p. 1p. 1p.</p>	<p>(*) Se acordă punctajul chiar dacă algoritmul ales nu este eficient.</p> <p>(*) Se acordă punctajul chiar dacă algoritmul ales nu este eficient. (**) Se acordă numai 3p. dacă algoritmul este principial 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. O soluție posibilă parcurge o dată fișierul, memorând bomboanele rămase în cutia actuală și în cutia precedentă (b respectiv a). Dacă a+b>x atunci din cutia actuală se iau atâtea bomboane încât în cele două cutii să rămână x bomboane (ans←ans+(a+b-x)), în cutia actuală rămânând b←x-a bomboane. După fiecare cutie a←b. <pre>fin >> n >> x; a = 0; for(int i=1; i <= n; i++) { cin >> b; if(a + b > x) { ans = ans + (a + b - x); b = x - a; } a = b; } cout << ans;</pre></p>

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