

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 intervalul căruia îi aparține valoarea memorată în variabila `x`, dacă și numai dacă expresia C/C++ alăturată are valoarea 1.
`!(x<=2020) && !(x>2025)`
a. (2020, 2025] b. (2020, 2025) c. [2020, 2025] d. (2021, 2025)
- Elementele unui tablou unidimensional sunt, în această ordine, (2, 3, 5, 7, 10, 12, 16, 17, 20, 21, 27). Pentru a verifica dacă în tablou există elementul cu valoarea `x`, se aplică metoda căutării binare. Indicați trei dintre valorile pe care le poate avea `x`, astfel încât căutarea să se încheie după ce `x` a fost comparat cu trei sau cu patru elemente ale tabloului.
a. 2, 3, 5 b. 2, 7, 21 c. 7, 16, 20 d. 20, 21, 27
- Indicați valoarea expresiei C/C++ alăturate.
`abs(-25)-25`
a. -50 b. -25 c. 0 d. 50
- În secvența C/C++ alăturată toate variabilele sunt de tip întreg, iar `n ≥ 2`. Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei `x` să fie egală cu cel mai mare divizor al numărului memorat în variabila `n`, divizor strict mai mic decât `n`.
`x=2;`
`while(n%x!=0)`
`x=x+1;`
`x=.....;`
a. `x-1` b. `x*x` c. `n/2` d. `n/x`
- În secvențele notate cu `s1`, `s2` și `s3`, definite mai jos, toate variabilele sunt întregi. Indicați secvențele în urma executării cărora, independent, variabila `s` memorează suma `1·2+2·3+3·4+4·5+5·6+6·7+7·8+8·9+9·10`.
`//s1`
`s=0;`
`for(i=10; i>1; i--)`
`s=s+i*(i-1);`
`//s2`
`s=0; i=1;`
`do { p=i*(i+1); s=s+p; i=i+1;`
`}while(i<10);`
`//s3`
`s=0; i=1; p=1;`
`while(i<10)`
`{ p=p*(i+1); s=s+p; i=i+1;}`
a. `s1`, `s2`, dar nu și `s3` b. `s1`, `s3`, dar nu și `s2` c. `s2`, `s3`, dar nu și `s1` d. `s1`, `s2` și `s3`

SUBIECTUL al II-lea

(40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.
S-a notat cu `a` și `b` restul împărțirii numărului natural `a` la numărul natural nenul `b` și cu `[c]` partea întreagă a numărului real `c`.
a. Scrieți ce se afișează în urma executării algoritmului dacă se citește, în această ordine, numerele 9767 și 8204. (6p.)
b. Dacă primul număr citit este 2025, scrieți două valori distincte din intervalul [10, 99] care pot fi citite pentru `y` astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, mesajul afișat să fie DA, urmat de o valoare numerică. (6p.)
c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind prima structură `cât timp...execută` cu o structură repetitivă cu test final. (6p.)

```
citește x, y
(numere naturale distincte)
cx ← 0; cy ← 0
cât timp x ≥ 10 sau y ≥ 10 execută
  cât timp x + y ≠ 0 execută
    cx ← cx + x % 10; x ← [x / 10]
    cy ← cy + y % 10; y ← [y / 10]
  x ← cx; cx ← 0; y ← cy; cy ← 0
dacă x = y atunci scrie "DA ", x
altfel scrie "NU ", x, " ", y
```

2. Tablourile unidimensionale $A = (2, 5, 7, 9)$ și $B = (x, y, z)$ sunt interclasate în ordine crescătoare, fiind parcurse de la stânga la dreapta. Scrieți un exemplu de valori naturale distincte pentru x , y și z , știind că cele două tablouri nu au nicio valoare comună și că x se compară cu o singură valoare din A , iar z nu se compară cu nicio valoare din A . (6p.)
3. O bijuterie este confecționată dintr-un singur metal prețios, iar variabilele întregi mb și gb memorează codul metalului din care este confecționată bijuteria, respectiv greutatea acesteia, exprimată în grame. Se cunosc date despre două metale prețioase, iar variabilele întregi $m1$ și $p1$ memorează codul primului metal prețios, respectiv prețul unui gram din acel metal, și variabilele întregi $m2$ și $p2$ memorează codul celui de al doilea metal prețios, respectiv prețul unui gram din acel metal. Scrieți o secvență de instrucțiuni C/C++ în urma executării căreia se afișează pe ecran prețul bijuteriei menționate, dacă ea este confecționată din unul dintre cele două metale prețioase despre care se cunosc date, sau mesajul **pret necunoscut**, în caz contrar. (6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Un fermier are un teren de formă dreptunghiulară, pe care vrea să îl împartă în parcele de arie egală, astfel încât numărul de parcele și aria să fie valori naturale, iar numărul de parcele să fie par și strict mai mic decât aria unei parcele. Se citesc două numere naturale din intervalul $[2, 100]$, x și y , reprezentând lungimea și lățimea terenului, măsurate în metri. Se cere să se scrie toate soluțiile posibile de împărțire a terenului, fiecare soluție pe câte un rând, sub forma a două numere, separate prin mesajul **parcele de arie** și spații adecvate, unde primul număr reprezintă numărul de parcele obținute, iar al doilea aria unei parcele corespunzătoare, ca în exemplu. Dacă nu există nicio astfel de soluție, se scrie mesajul **nu exista**. Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate. Operația de trecere la un rând nou se precizează în algoritm astfel: **scrie rând nou**.
Exemplu: pentru $x=6$ și $y=8$, se afișează valorile alăturate, nu neapărat în această ordine (terenul poate fi împărțit și în 3 parcele de arie 16 sau 8 parcele de arie 6, dar nu corespund cerinței). (10p.)
- | | |
|--|----------------------|
| | 2 parcele de arie 24 |
| | 4 parcele de arie 12 |
| | 6 parcele de arie 8 |
2. Scrieți un program C/C++ care citește de la tastatură un număr natural, n ($n \in [2, 10^2]$), și un șir de n numere naturale distincte din intervalul $[0, 10^9]$, elemente ale unui tablou unidimensional. Programul modifică tabloul în memorie, înlocuind valoarea minimă cu valoarea aflată pe ultima poziție. Elementele tabloului obținut sunt afișate pe ecran, separate prin câte un spațiu.
Exemplu: pentru $n=8$ și tabloul (17, 19, 10, 4, 23, 3, 11, 9) se obține tabloul (17, 19, 10, 4, 23, 9, 11, 9) (10p.)
3. La o centrală solară se monitorizează **stocul zilnic** de energie produsă și **stocul total** calculat pentru fiecare perioadă. Zilele de monitorizare sunt numerotate cu valori naturale consecutive, în ordine cronologică, începând cu ziua 1. O perioadă este formată din una sau mai multe zile de monitorizare, consecutive, iar stocul total calculat pentru ea este suma stocurilor zilnice corespunzătoare. O **zi este validată** dacă stocul zilnic este cel puțin egal cu limita zilnică, **minZ**; o **perioadă este validată** dacă stocul total calculat pentru ea este cel puțin egal cu limita stabilită pentru perioade, **minP**, fiecare zi a perioadei este validată, iar perioada este maximală în raport cu această proprietate (nu i se mai poate adăuga nicio zi validată). Fișierul text **bac.in** conține cel mult 10^6 numere naturale din intervalul $[1, 10^3]$: pe prima linie **minZ** și **minP**, reprezentând limitele precizate pentru validare, iar pe a doua linie stocurile zilnice de energie produse în zile consecutive, în ordinea monitorizării. Numerele aflate pe aceeași linie a fișierului sunt separate prin câte un spațiu. Se cere să se afișeze pe ecran, pentru fiecare perioadă de producție validată, corespunzătoare datelor din fișier, stocul total calculat pentru ea. Valorile se afișează pe câte o linie a ecranului, iar dacă nu există nicio astfel de perioadă, se afișează mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.
Exemplu: dacă fișierul conține valorile alăturate, se afișează pe ecran
- | | |
|----|---------------------------------------|
| 10 | 40 |
| 65 | 9 20 25 12 14 7 3 11 15 12 8 19 50 21 |
| 65 | |
| 71 | |
| 90 | |
- 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

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

Varianta 4

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

SUBIECTUL al II - lea

(40 de puncte)

1.	a. Răspuns corect: NU 2 5	6p.	Se acordă câte 2p. pentru fiecare valoare afișată (mesaj, respectiv cele două numere) conform cerinței.
	b. Pentru răspuns corect	6p.	Se acordă câte 3p. pentru fiecare dintre cele două valori conform cerinței (orice număr de două cifre, cu suma cifrelor egală cu 9 sau cu 18) sau numai câte 2p. pentru fiecare dintre cele două valori care conduc la afișarea datelor cerute, dar nu aparțin intervalului indicat.
	c. Pentru program corect -declarare 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 (*) -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 conform cerinței (repetă...până când, repetă... cât timp, execută...cât timp, do...while etc.). (**) Se acordă numai 2p. dacă doar un aspect (expresie logică pentru test final, algoritm echivalent pentru cazul inițial în care $x < 10$ și $y < 10$) este conform cerinței.
2.	Pentru rezolvare corectă	6p.	Se acordă câte 2p. pentru fiecare dintre cele trei valori conform cerinței (oricare valori cu proprietatea că $x < 2$; $9 < y < z$)
3.	Pentru răspuns corect -determinare a prețului cerut (*) -afișare a datelor și tratare a cazului preț necunoscut, conform cerinței -corectitudine globală a secvenței ¹⁾	6p. 4p. 1p. 1p.	(*) Se acordă câte 2p. pentru fiecare aspect (calcul al prețului în funcție de greutate și prețul unui gram de metal, corespondență preț-metal) conform cerinței.

SUBIECTUL al III - lea

(30 de puncte)

1.	Pentru algoritm corect -citire a datelor, conform cerinței -determinare a valorilor cerute (**) -scriere a datelor și tratare a cazului nu exista, conform cerinței -scriere principal corectă a structurilor de control, corectitudine globală a algoritmului ¹⁾ (**)	10p. 1p. 6p. 1p. 2p.	(*) Se acordă câte 1p. pentru fiecare aspect (identificare a ariei unei parcele/a unui divizor al ariei totale, identificare a numărului de parcele corespunzător ariei identificate, număr par de parcele, număr de parcele mai mic decât aria unei parcele, identificare a unei perechi corespunzătoare număr de parcele-arie a unei parcele, perechi suport număr de parcele-arie a unei parcele identificate) conform cerinței. (**) Se punctează orice formă explicită de structură repetitivă sau decizională.
2.	Pentru program corect -declarare a unei variabile care să memoreze un tablou unidimensional, conform cerinței -citire a datelor, conform cerinței -modificare a tabloului conform cerinței (*) -afișare a datelor, conform cerinței -declarare a variabilelor simple, corectitudine globală a programului ¹⁾	10p. 1p. 1p. 6p. 1p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect (algoritm de bază pentru determinarea unei valori minime, valori suport verificate pentru determinarea minimului, poziție suport a minimului, identificare a elementului aflat pe ultima poziție, element suport înlocuit/elemente suport nemodificate, modificare în memorie) conform cerinței.
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, tratare a cazului nu exista, 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 parcurge pașii necesari rezolvării, dar cu detalii care conduc la o rezolvare parțială. (***) Se acordă punctajul numai pentru un algoritm liniar, care utilizează eficient memoria. 1p. O soluție posibilă citește primele două valori din fișier 5p. (minZ și minP), apoi, pe măsura citirii stocurilor zilnice, dacă stocul curent citit (x) corespunde unei zile validate ($x \geq \min Z$), se actualizează stocul total ($\text{stocCrt} = \text{stocCrt} + x$) al perioadei curente. 1p. Dacă ziua nu este validată ($x < \min Z$) sau dacă s-a încheiat citirea, se afișează stocul total determinat, dacă acesta corespunde unei perioade validate ($\text{stocCrt} \geq \min P$), apoi se reinițializează această valoare cu 0.

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