

Examenul de bacalaureat național 2020
Proba E. d)
Informatică
Limbajul C/C++

Testul 5

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

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 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 o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul natural memorat în variabila întreagă x are exact o cifră.
 - $x/10==0$
 - $x\%10==0$
 - $(x/10)/10==0$
 - $(x\%10)\%10==0$
- Subprogramul f este definit alăturat. Scrieți ce se afișează în urma apelului de mai jos.
 $f(3)$;

```
void f(int n)
{ int i;
  for(i=n;i>=1;i--)
  { f(n-1);
    cout<<i; | printf("%d",i);
  }
}
```

 - 121321
 - 1211312111212111
 - 322111
 - 321112211112111
- Utilizând metoda backtracking, se generează toate modalitățile de a pregăti o ținută, luând, într-o anumită ordine, articolele din mulțimea {cămașă, cravată, pantaloni, pantofi, sacou, șosete}, având în vedere următoarele restricții: cămașa va fi luată înaintea cravatei, cravata înaintea sacoului și atât șosetele, cât și pantalonii, înaintea pantofilor. Primele trei soluții generate sunt, în această ordine: (cămașă, cravată, pantaloni, sacou, șosete, pantofi), (cămașă, cravată, pantaloni, șosete, sacou, pantofi), (cămașă, cravată, pantaloni, șosete, sacou, pantofi). Indicați cea de a șasea soluție generată.
 - (cămașă, cravată, sacou, șosete, pantaloni, pantofi)
 - (cămașă, cravată, șosete, pantaloni, sacou, pantofi)
 - (cămașă, cravată, șosete, pantaloni, pantofi, sacou)
 - (cămașă, cravată, șosete, sacou, pantaloni, pantofi)
- Un arbore cu 9 noduri, numerotate de la 1 la 9, este reprezentat prin vectorul de „tați” (2, 7, 0, 8, 1, 5, 3, 9, 2). Rădăcina arborelui este:
 - 1
 - 3
 - 4
 - 6
- Matricea de adiacență a unui graf neorientat cu 2020 de noduri are 200 de elemente nenule. Numărul maxim de componente conexe ale grafului este:
 - 2006
 - 2000
 - 1820
 - 400

SUBIECTUL al II-lea

(40 de puncte)

1. **Algoritmul alăturat este reprezentat în pseudocod.**
S-a notat cu $a \div b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .
- a. Scrieți ce se afișează dacă se citește numărul 100. **(6p.)**
- b. Scrieți toate numerele din intervalul $[1, 9]$ care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze N . **(6p.)**
- c. Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, care să nu cuprindă nicio structură repetitivă. **(6p.)**
- ```

citește n
(număr natural nenul)
x ← 1; y ← n; d ← 2
cât timp x < y execută
 dacă n % d = 0 atunci
 x ← d
 y ← [n/d]
 d ← d + 1
dacă x = y atunci
 scrie 'D', x
altfel scrie 'N'

```
2. Variabila  $s$  memorează simultan numărul de soluții complexe ale unei ecuații (număr natural din intervalul  $[2, 10^2)$ ) și soluțiile propriu-zise (partea reală și partea imaginară, numere reale). Știind că expresiile C/C++ de mai jos au ca valori numărul de soluții ale unui ecuații, partea reală, respectiv partea imaginară a primei sale soluții, scrieți definiția unei structuri cu eticheta `ecuatie`, care permite memorarea datelor despre soluțiile unei ecuații, și declarați corespunzător variabila  $s$ .
- ```

s.numar      s.solutie[0].pre      s.solutie[0].pim
  
```
- (6p.)**
3. Variabilele i și j sunt de tip întreg, iar variabila a memorează un tablou bidimensional cu 6 linii și 6 coloane, numerotate de la 0 la 5, având inițial toate elementele egale cu caracterul `@`. Fără a utiliza alte variabile, scrieți secvența de instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila a să memoreze tabloul alăturat.
- ```

for (i=0; i<6; i++)
 for (j=0; j<6; j++)


```
- ```

( ( ( ) ) )
* ( ( ) ) *
* * ( ) * *
* * ( ) * *
* ( ( ) ) *
( ( ( ) ) )
  
```
- (6p.)**

SUBIECTUL al III-lea

(30 de puncte)

1. Un număr este scris în baza de numerație b ($b \leq 10$) dacă cifrele sale aparțin intervalului $[0, b-1]$. Subprogramul `baza` are un singur parametru, n , prin care primește un număr natural ($n \in [0, 10^9]$). Subprogramul returnează cea mai mică bază din intervalul $[2, 10]$ căreia i-ar putea corespunde scrierea lui n . Scrieți definiția completă a subprogramului.
Exemplu: dacă $n=50731$, subprogramul returnează numărul 8. **(10p.)**
2. Un text cu cel mult 100 de caractere conține cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere mici ale alfabetului englez, iar numerele sunt reale, pozitive, cu partea zecimală și partea întreagă separate prin simbolul virgulă, sau numai cu partea întreagă, ca în exemplu. Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și afișează pe ecran numărul de valori întregi din text.
Exemplu: pentru textul
`grus leucoggeranus are 1,40 m inaltime si traieste intre 30 si 40 de ani`
se afișează pe ecran 2 **(10p.)**
3. Fișierul `bac.txt` conține un șir de cel mult 10^6 numere întregi din intervalul $[-10^3, 10^3]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran suma maximă obținută adunând numere de pe poziții consecutive în șirul aflat în fișier. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.
Exemplu: dacă fișierul `bac.txt` conține valorile 4 -6 7 2 -1 4 -10 -3 9 2 -2 se afișează pe ecran numărul 12
- a. Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**
b. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**

Examenul de bacalaureat național 2020

Proba E. d)

Informatică

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

Testul 5

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

- Se punctează oricare alte modalități de rezolvare corectă a cerințelor.
- Nu se acordă punctaje intermediare, altele decât cele precizate explicit prin barem. Nu se acordă fracțiuni de punct. Se acordă 10 puncte din oficiu. Nota finală se calculează prin împărțirea punctajului total acordat pentru lucrare la 10.
- Utilizarea unui tip de date care depășește domeniul de valori precizat în enunț este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

SUBIECTUL I

(20 de puncte)

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

SUBIECTUL al II - lea

(40 de puncte)

1.	a) Răspuns corect: D10	6p.	Se acordă câte 3p. pentru fiecare dintre cele două valori conform cerinței.
	b) Răspuns corect: 2 3 5 6 7 8	6p.	Se acordă câte 1p. pentru fiecare dintre numerele conform cerinței
	c) Pentru program corect -declarare variabile -citire date -afișare date -instrucțiuni de decizie (*) -instrucțiune repetitivă -atribuiri -corectitudine globală a programului ¹⁾	10p. 1p. 1p. 1p. 3p. 2p. 1p. 1p.	(*) Se acordă numai 2p. dacă doar una dintre instrucțiunile de decizie este conform cerinței.
	d) Pentru algoritm pseudocod corect -echivalență a prelucrării realizate, conform cerinței (*) -corectitudine globală a algoritmului ¹⁾	6p. 5p. 1p.	(*) Se acordă numai 2p. dacă algoritmul are structuri conform cerinței, principial corecte, dar nu este echivalent cu cel dat.
	2. Pentru rezolvare corectă -definire a structurii/înregistrării (*) -declarare a variabilei conform cerinței -corectitudine globală a secvenței ¹⁾	6p. 3p. 2p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect (definire principial corectă a unei structurii/înregistrării, câmpuri, etichetă/nume) conform cerinței.
3. Pentru rezolvare corectă -acces la un element al tabloului -atribuire a valorilor indicate elementelor tabloului (*) -corectitudine globală a secvenței ¹⁾	6p. 1p. 4p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect specific (atribuire valori (, atribuire valori), atribuire valori *, elemente suport) conform cerinței.	

SUBIECTUL al III - lea

(30 de puncte)

1.	Pentru subprogram corect -antet subprogram (*) -determinare a valorii cerute (**) -instrucțiune/instrucțiuni de returnare a rezultatului -declarare a tuturor variabilelor locale, corectitudine globală a subprogramului ¹⁾	10p.	(*) Se acordă câte 1p. pentru fiecare aspect al antetului (structură, parametru de intrare) conform cerinței.
		2p.	
		6p.	(**) Se acordă câte 2p. pentru fiecare aspect specific (identificare a unei cifre, algoritm principial corect pentru determinarea unei valori minime/maxime, stabilire bază) conform cerinței.
		1p.	(***) Se acordă câte 1p. pentru fiecare aspect specific (simbol -, spațiere) conform cerinței.

2.	<p>Pentru program corect -declarare a unei variabile care să memoreze un șir de caractere -citire a datelor -determinare a valorii cerute (*) -afișare a datelor -declarare a variabilelor simple, corectitudine globală a programului¹⁾</p>	<p>10p. 1p. 1p. 6p. 1p. 1p.</p>	<p>(*) Se acordă câte 2p. pentru fiecare aspect specific (identificare a unei entități/cuvânt, identificare a unui număr întreg/real, algoritm de numărare principal corect) 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, citire a datelor, tratare a cazului nu exista, 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. (**) 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. O soluție posibilă parcurge șirul din fișier, memorând valoarea curentă (crt), suma secvenței curente, s, și suma maximă; pentru fiecare valoare curentă se decide dacă includerea sa în secvența curentă este preferabilă începerii unei secvențe noi, deci se alege cea mai mare dintre valorile $s=s+crt$ sau $s=crt$. Suma maximă se actualizează, după caz, la fiecare pas.</p>

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