

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

Testul 19

*Filieră teoretică, profil real, specializare științe ale naturii*

- 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.

**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.

- Variabilele  $x$ ,  $y$  și  $z$  sunt de tip întreg. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă  $x$ ,  $y$  și  $z$  au valori identice.
  - `!(x!=y) || !(y!=z)`
  - `!(x!=y!=z)`
  - `!(x!=y) || (y!=z)`
  - `!(x!=y || y!=z)`
- Tabloul unidimensional  $A$  are elementele:  $A=(2,20,27,36,50)$ , iar în urma interclasării lui descrescătoare cu tabloul unidimensional  $B$  se obține tabloul cu elementele  $(63,50,45,36,27,20,8,5,3,2)$ . Indicați elementele tabloului  $B$ , în ordinea apariției lor în acesta.
  - $(2,20,27,36,50,1,15,19,9,13)$
  - $(3,5,8,45,63)$
  - $(19,15,13,9,1)$
  - $(63,45,8,5,3,50,36,27,20,2)$
- Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea  $x=9$ , se aplică metoda căutării binare, iar succesiunea de elemente a căror valoare se compară cu  $x$  pe parcursul aplicării metodei este 14, 8, 9. Elementele tabloului pot fi:
  - $(2,9,7,8,12,14,48)$
  - $(14,8,9,9,8,14)$
  - $(4,8,9,14,16,24,48)$
  - $(14,14,8,8,9,9)$
- Variabilele  $x$  și  $y$  sunt de tip întreg. Indicați instrucțiunea care realizează o prelucrare echivalentă cu cea alăturată.

				$x=abs(y)$ ;
a. <code>if(x&lt;y &amp;&amp; y&lt;0)</code> <code>  x=-y;</code>	b. <code>if(x&gt;y &amp;&amp; y&gt;0)</code> <code>  y=-x;</code>	c. <code>if(y&lt;0) x=-y;</code> <code>  else x=y;</code>	d. <code>if(x&lt;0) y=-x;</code> <code>  else y=x;</code>	
- Variabilele  $x$ ,  $y$ ,  $z$ ,  $w$  și  $r$  sunt de tip întreg, iar  $r$  are inițial valoarea 0. Indicați o secvență echivalentă cu cea de mai jos.  
`if(x!=y) r=3; else if(z==w) r=1; else r=2;`
  - `if (x==y && z==w) r=1; else if(x==y && z!=w) r=2; else r=3;`
  - `if (x==y || z==w) r=1; else if(x==y || z!=w) r=2; else r=3;`
  - `if (x==y && z==w) r=1; else if(x==y && z!=w) r=2; else if(x!=y && z!=w) r=3;`
  - `if (x==y || z==w) r=1; else if(x==y || z!=w) r=2; else if(x!=y || z!=w) r=3;`

**SUBIECTUL al II-lea**

**(40 de puncte)**

1. Algoritmul alăturat este reprezentat în pseudocod.  
S-a notat cu  $a \% b$  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 numărul 60534. (6p.)

b. Scrieți trei numere din intervalul  $[0, 10^2]$  care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze 88. (6p.)

c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)

citește  $n$  (număr natural nenul)

cât timp  $n \neq 0$  execută

$c \leftarrow n \% 10; n \leftarrow [n/10]$

dacă  $c \% 2 \neq 0$  atunci

$c \leftarrow c + 1$

cât timp  $c > 0$  și  $c < 10$  execută

scrie  $c$

$c \leftarrow c * 2$

d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat a doua structură cât timp...execută cu o structură repetitivă de alt tip. (6p.)

2. Variabilele întregi  $cod1$  și  $nr1$  memorează codul unui proiect, respectiv numărul surselor de documentare consultate în cadrul acestuia, iar variabilele întregi  $cod2$  și  $nr2$  memorează codul unui alt proiect, respectiv numărul surselor de documentare consultate în cadrul acestuia. Știind că numărul de surse de documentare consultate este diferit pentru cele două proiecte, scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran, separate printr-un spațiu, codul proiectului cu număr maxim de surse documentare, precum și partea întreagă a numărului mediu de surse consultate în cadrul celor două proiecte. (6p.)

3. Variabila  $a$  memorează un caracter, iar variabilele  $i$  și  $k$  sunt de tip întreg. De la tastatură se citesc numai litere mici ale alfabetului englez.

Fără a utiliza alte variabile, scrieți secvența înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila  $k$  să memoreze numărul de vocale citite. Se consideră vocale literele  $a, e, i, o, u$ .

Exemplu: dacă se citesc literele i n f o r m a t i e în urma executării secvenței,  $k=5$ . (6p.)

.....

for ( $i=1; i \leq 10; i++$ )

{  $cin \gg a; | scanf("%c", \&a);$

.....

}

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Se citește un număr natural nenul  $n$  și se cere să se scrie numărul de divizori naturali ai lui  $n$  cu aceeași paritate ca  $n$ .

Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.

Exemplu: dacă  $n=20$  se scrie 4 (divizorii lui 20 sunt 1, 2, 4, 5, 10, 20).

(10p.)

2. Scrieți un program C/C++ care citește de la tastatură un număr natural,  $n$  ( $n \in [2, 20]$ ), și cele  $n$  elemente ale unui tablou unidimensional, numere naturale din mulțimea  $\{0, 1\}$ . Programul afișează pe ecran mesajul DA dacă există cel puțin o pereche de elemente complementare egal depărtate de extremitățile tabloului, sau mesajul NU în caz contrar. Două elemente sunt complementare dacă au valori diferite.

Exemplu: pentru  $n=10$  și tabloul (1, 1, 0, 0, 1, 1, 0, 1, 1, 0), se afișează pe ecran mesajul DA. (10p.)

3. Fișierul bac.txt conține, în ordine crescătoare, cel mult  $10^6$  numere naturale din intervalul  $[0, 10^9]$ , separate prin câte un spațiu. Se cere să se afișeze pe ecran, în ordine strict crescătoare, separate prin câte un spațiu, numerele distincte care apar în fișier. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fișierul conține numerele 5 5 6 7 8 8 8 8 10 15 16 16 25 25 49

pe ecran se afișează, în această ordine, numerele 5 6 7 8 10 15 16 25 49

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)

b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)

(2p.)

(8p.)

**Examenul de bacalaureat național 2020**  
**Proba E. d)**  
**Informatică**

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

**Testul 19**

*Filieră teoretică, profil real, specializare științe ale naturii*

- 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)

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

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

1.	a) Răspuns corect: 484866	6p.	Se acordă câte 1p. pentru fiecare cifră conform cerinței.
	b) Pentru răspuns corect	6p.	Se acordă câte 2p. pentru fiecare dintre cele trei numere conform cerinței (oricare dintre numerele 77, 78, 87, 88)
	c) Pentru program corect -declarare variabile -citire date -afișare date -instrucțiune de decizie -instrucțiuni repetitive (*) -atribuiri -corectitudine globală a programului <sup>1)</sup>	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 -echivalență a prelucrării realizate, conform cerinței (*) -corectitudine globală a algoritmului <sup>1)</sup>	6p. 5p. 1p.	(*) Se acordă numai 2p. dacă algoritmul are o structură repetitivă conform cerinței, principal corectă, dar nu este echivalent cu cel dat. Se va puncta orice formă corectă de structură repetitivă conform cerinței.
2.	Pentru rezolvare corectă -determinare a valorilor cerute (*) -afișare a datelor în formatul cerut -corectitudine globală a secvenței <sup>1)</sup>	6p. 4p. 1p. 1p.	(*) Se acordă câte 2p. pentru fiecare valoare (număr maxim de surse, număr mediu de surse) conform cerinței.
3.	Pentru rezolvare corectă -inițializare a variabilei k -determinare a valorii cerute (*) -corectitudine globală a secvenței <sup>1)</sup>	6p. 1p. 4p. 1p.	(*) Se acordă câte 2p. pentru fiecare aspect specific (identificare a unei vocale/consoane, algoritm de numărare principal corect) conform cerinței.

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

1.	Pentru algoritm corect -citire a datelor -determinare a numărului cerut (*) -scriere a datelor -scriere principal corectă a structurilor de control, corectitudine globală a algoritmului <sup>1)</sup> (**)	10p. 1p. 6p. 1p. 2p.	(*) Se acordă câte 2p. pentru fiecare aspect specific (identificare a unui divizor, algoritm principal corect de numărare, divizori suport) conform cerinței. (**) Se va puncta orice formă corectă de structură repetitivă sau decizională.
2.	Pentru program corect -declarare a unei variabile care să memoreze un tablou unidimensional -citire a tabloului -verificare a proprietății cerute (*) -afișare a mesajului -declarare a variabilelor simple, citire a datelor simple, corectitudine globală a programului <sup>1)</sup>	10p. 1p. 1p. 6p. 1p. 1p.	(*) Se acordă câte 2p. pentru fiecare aspect specific (algoritm principal corect de verificare a unei proprietăți, perechi de elemente complementare, perechi suport) conform cerinței.

<b>3.</b>	<b>a) Pentru răspuns corect</b> -coerență a descrierii algoritmului (*) -justificare a elementelor de eficiență	<b>2p.</b> 1p. 1p.	(*) 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.
	<b>b) Pentru program corect</b> -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 programului <sup>1)</sup>	<b>8p.</b> 1p. 5p. 1p. 1p.	(***) Se acordă punctajul numai pentru un algoritm liniar care utilizează eficient memoria. O soluție posibilă parcurge șirul din fișier, memorând valoarea curentă (crt) și valoarea anterioară acesteia (ant); la întâlnirea unei valori crt diferită de ant, se afișează crt și se actualizează corespunzător variabila ant.

<sup>1)</sup> Corectitudinea globală vizează structura, sintaxa, alte aspecte neprecizate în barem.