

**Examenul național de bacalaureat 2021**

**Proba E. d)**  
**INFORMATICĂ**  
**Limbajul C/C++**

**Testul 9**

*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 muchie are extremități distincte și oricare două 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.**

- Variabila  $x$  este de tip întreg și memorează un număr natural de cel puțin două cifre. Indicați o instrucțiune C/C++ în urma executării căreia se memorează în variabila  $x$  numărul obținut din valoarea sa inițială, prin inserarea cifrei 4 între cifra zecilor și cifra unităților.
  - $x = (x/10*10+4) *10+x\%10 ;$
  - $x = x/10+4+x\%10 ;$
  - $x = (x\%10*10+4) *10+x/10 ;$
  - $x = (x/10+4) *10+x\%10 ;$
- Utilizând metoda backtracking se generează toate posibilitățile de a planta de-a lungul unei străzi cinci arbori din mulțimea {salcie, carpen, larice, fag, ulm}. Două soluții sunt diferite dacă ordinea arborilor diferă. Primele patru soluții obținute sunt, în această ordine: (salcie, carpen, larice, fag, ulm), (salcie, carpen, larice, ulm, fag), (salcie, carpen, fag, larice, ulm), (salcie, carpen, fag, ulm, larice). Indicați antepenultima soluție generată.
  - (ulm, fag, larice, salcie, carpen)
  - (ulm, salcie, larice, fag, carpen)
  - (ulm, fag, carpen, larice, salcie)
  - (ulm, fag, carpen, salcie, larice)
- Indicați o declarație corectă pentru o variabilă  $x$  care să memoreze simultan tensiunea electrică (număr întreg) și intensitatea curentului electric (număr real) într-un circuit.
  - ```
struct circuit
{ int x.tensiune;
  double x.intensitate;
};
```
  - ```
struct x
{ int tensiune;
  double intensitate;
};
```
  - ```
int x.tensiune;
double x.intensitate;
```
  - ```
struct
{ int tensiune;
  double intensitate;
} x;
```
- Într-un arbore cu rădăcină un nod se află pe nivelul  $x$  dacă lanțul elementar care are o extremitate în nodul respectiv și cealaltă extremitate în rădăcina arborelui are lungimea  $x$ . Pe nivelul 0 se află un singur nod (rădăcina).  
Un arbore are 8 noduri, numerotate de la 1 la 8, și muchiile [1,3], [1,7], [1,8], [2,4], [3,5], [3,6], [4,5]. Indicați două noduri care pot fi alese drept rădăcină astfel încât, pentru fiecare dintre arborii obținuți, pe nivelul 2 să fie un număr maxim de noduri.
  - 3,5
  - 4,7
  - 5,7
  - 6,8
- Un graf neorientat cu 5 noduri, numerotate de la 1 la 5, are muchiile [2,5], [3,1], [5,3], [5,4]. Indicați numărul minim de muchii care pot fi adăugate, astfel încât în graful obținut să existe cel puțin un ciclu elementar care să conțină toate nodurile acestuia.
  - 1
  - 2
  - 3
  - 4

**SUBIECTUL al II-lea**

**(40 de puncte)**

1. Algoritmul alăturat este reprezentat în pseudocod.  
S-a notat cu  $x\%y$  restul împărțirii numărului natural  $x$  la numărul natural, nenul,  $y$ .
- a. Scrieți ce se va afișa în urma executării algoritmului dacă se citește numărul 3. **(6p.)**
- b. Scrieți toate valorile care pot fi citite astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, numărul total de caractere + (plus) afișate să fie cuprins în intervalul închis  $[10, 20]$ . **(6p.)**
- c. Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat doar una dintre structurile **pentru...execută** cu o structură repetitivă de tip **cât timp...execută**. **(6p.)**
2. Subprogramul  $f$  este definit alăturat. Scrieți toate valorile naturale din intervalul  $[1, 10]$  pe care le poate avea  $x$ , astfel încât valoarea lui  $f(10, x)$  să fie un număr strict mai mare decât 20. **(6p.)**
3. Variabilele  $i$  și  $j$  sunt de tip întreg, iar variabila  $a$  memorează un tablou bidimensional cu 4 linii și 5 coloane, numerotate începând de la 0, cu elemente numere întregi, inițial toate nule. Fără a utiliza alte variabile decât cele menționate, scrieți secvența de mai jos astfel încât, în urma executării secvenței obținute, variabila  $a$  să memoreze tabloul alăturat.
- ```
for(i=0;i<4;i++)
    for(j=0;j<5;j++)
        .....
```
- |   |   |   |   |   |
|---|---|---|---|---|
| 0 | 1 | 2 | 0 | 1 |
| 1 | 2 | 0 | 1 | 2 |
| 2 | 0 | 1 | 2 | 0 |
| 0 | 1 | 2 | 0 | 1 |
- (6p.)**

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Subprogramul `divizor` are patru parametri:
- $a, b$  și  $k$ , prin care primește câte un număr natural ( $a \in [0, 10^9], b \in [a, 10^9], k \in [1, 9]$ );
  - $nr$ , prin care furnizează numărul de valori naturale din intervalul  $[a, b]$  care sunt divizibile cu  $k$  și au ultima cifră egală cu  $k$ . Scrieți definiția completă a subprogramului.
- Exemplu:** dacă  $a=3, b=50$  și  $k=4$ , în urma apelului,  $nr=3$  (pentru numerele 4, 24, 44). **(10p.)**
2. Într-un text cu cel mult  $10^2$  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 și afișează pe ecran, pe linii separate, toate cuvintele sale care conțin o singură vocală distinctă, ca în exemplu. Dacă nu există niciun astfel de cuvânt, se afișează pe ecran mesajul **nu exista**. Se consideră vocale literele din mulțimea  $a, e, i, o, u$ .
- Exemplu:** pentru textul **a plantat cinci lalele visinii sau rosii** se afișează pe ecran, nu neapărat în această ordine, cuvintele alăturate. **(10p.)**
- |       |         |
|-------|---------|
| a     | plantat |
| cinci | visinii |
3. Fișierul `numere.txt` conține cel mult  $10^5$  numere naturale din intervalul  $[1, 10^9]$ , câte unul pe fiecare linie. Se cere să se afișeze pe ecran cel mai mare număr care se poate forma cu toate cifrele care apar în numerele din fișier, ca în exemplu.
- Proiectați un algoritm eficient din punctul de vedere al timpului de executare.
- Exemplu:** dacă fișierul are conținutul alăturat, se afișează
- ```
9988887766333220
```
- |     |        |
|-----|--------|
| 263 | 39628  |
| 79  | 887308 |
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**

Examenul național de bacalaureat 2021

Proba E. d)  
INFORMATICĂ

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

Testul 9

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 este acceptată dacă acest lucru nu afectează corectitudinea în funcționarea programului.

SUBIECTUL I

(20 de puncte)

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

SUBIECTUL al II - lea

(40 de puncte)

1. a) Răspuns corect: +++@+++@	6p.	Se acordă numai 2p. dacă doar primele trei caractere sunt conform cerinței, numai 3p. dacă doar primele patru caractere sunt conform cerinței, numai 4p. dacă doar primele 7 caractere sunt conform cerinței sau dacă, în continuarea caracterelor cerute, s-au precizat și alte caractere.
b) Răspuns corect: 4 5	6p.	Se acordă câte 3p. pentru fiecare dintre cele două numere conform cerinței.
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. Răspuns corect: 1 și 9	6p.	Se acordă câte 3p. pentru fiecare valoare conform cerinței.
3. Pentru rezolvare corectă -acces la un element al tabloului -determinare a valorilor conform cerinței (*) -corectitudine globală a secvenței <sup>1)</sup>	6p. 1p. 4p. 1p.	(*) Se acordă câte 2p. pentru fiecare aspect specific (mulțime a valorilor suport utilizate, corespondență valori- poziții) conform cerinței.

SUBIECTUL al III - lea

(30 de puncte)

1. Pentru subprogram corect -antet subprogram (*) -determinare a valorii cerute (**) -declarare a tuturor variabilelor locale, corectitudine globală a subprogramului <sup>1)</sup>	10p. 3p. 6p. 1p.	(*) Se acordă câte 1p. pentru fiecare aspect al antetului (structură, parametri de intrare, parametru de ieșire) conform cerinței. (**) Se acordă câte 1p. pentru fiecare aspect (acces la o cifră a unui număr, identificare a unei valori care are ultima cifră egală cu k, identificare a unei valori care este divizibilă cu k, identificare a unei valori din intervalul cerut, algoritm de bază pentru numărarea unei serii de valori, valori suport numărate) conform cerinței.
--	---------------------------	---

2.	<p><b>Pentru program corect</b> -declarare a unei variabile care să memoreze un șir de caractere -citire a datelor -determinare a cuvintelor cerute (*) -afișare a datelor în formatul cerut și tratare a cazului nu exista -declarare a variabilelor simple, corectitudine globală a programului<sup>1)</sup></p>	<p><b>10p.</b> 1p. 1p. 6p. 1p. 1p.</p>	<p>(*) Se acordă câte 1p. pentru fiecare aspect specific (identificare a unui cuvânt, identificare a unei vocale/consoane, algoritm de bază pentru verificarea unei proprietăți, identificare a unui cuvânt care conține cel puțin o vocală, identificare a unui cuvânt care conține o singură vocală distinctă, cuvinte suport cu proprietatea cerută determinate) conform cerinței.</p>
3.	<p><b>a) Pentru răspuns corect</b> -coerență a descrierii algoritmului (*) -justificare a elementelor de eficiență</p> <p><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></p>	<p><b>2p.</b> 1p. 1p.</p> <p><b>8p.</b> 1p. 5p. 1p. 1p.</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. O soluție posibilă folosește un vector de frecvență <math>ap</math>, inițial cu toate valorile nule, unde <math>ap_x</math> este egal cu numărul de apariții ale cifrei <math>x</math> în toate numerele din fișier. Pe măsura citirii datelor din fișier se actualizează vectorul <math>ap</math>, parcurgând cifrele fiecărui număr citit, apoi se afișează fiecare cifră <math>x</math>, cu <math>x</math> de la 9 la 0, de câte <math>ap_x</math> ori.</p>

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