

Examenul național de bacalaureat 2021

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

Testul 3

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.

1. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul memorat în variabila întreagă x aparține reuniunii de intervale $[-21, -2] \cup [2, 21]$.

- a. $x > -21 \ \&\& \ x < -2 \ \&\& \ x > 2 \ \&\& \ x < 21$
b. $!(x < -21 \ || \ x > -2) \ || \ !(x < 2 \ || \ x > 21)$
c. $x > -21 \ || \ x < -2 \ || \ x > 2 \ || \ x < 21$
d. $!(x < -21 \ \&\& \ x > 21 \ \&\& \ x > -2 \ || \ x < 2)$

2. Subprogramul f este definit alăturat. Indicați ce se afișează în urma apelului de mai jos.
 $f(54321)$;

```
void f (int x)
{ cout<<"*"; | printf("*");
  if(x>0)
  { cout<<x; | printf("%d",x);
    f(x/100);
  }
  cout<<"/"; | printf("/");
}
```

- a. ****554354321
b. ****/5/543/54321/
c. *54321*543*5*///
d. /5/543/54321/

3. Utilizând metoda backtracking, se generează toate modalitățile de a selecta un grup de patru muzee de vizitat în București, dintre cele aflate în mulțimea {Muzeul de Artă Veche Apuseană (**MAVA**), Muzeul Colecțiilor de Artă (**MCA**), Muzeul Căilor Ferate Române (**MCFR**), Muzeul Național al Hârtiei și Cărții Vechi (**MNHCV**), Muzeul Național al Literaturii Române (**MNLR**), Muzeul Național Tehnic (**MNT**)}. Două grupuri diferă prin cel puțin un muzeu. Primele cinci soluții generate sunt, în această ordine: (**MAVA**, **MCA**, **MCFR**, **MNHCV**), (**MAVA**, **MCA**, **MCFR**, **MNLR**), (**MAVA**, **MCA**, **MCFR**, **MNT**), (**MAVA**, **MCA**, **MNHCV**, **MNLR**), (**MAVA**, **MCA**, **MNHCV**, **MNT**). Indicați o enumerare care este generată ca soluție, în această ordine.

- a. (**MNHCV**, **MNLR**, **MCFR**, **MNT**)
b. (**MCA**, **MNHCV**, **MNLR**, **MNT**)
c. (**MCA**, **MCFR**, **MNHCV**, **MNLR**, **MNT**)
d. (**MAVA**, **MCA**, **MNHCV**, **MCFR**)

4. Un graf neorientat cu 5 noduri este reprezentat prin matricea de adiacență alăturată. Indicați numărul de muchii care pot fi eliminate, astfel încât graful parțial obținut să fie arbore.

0	1	1	0	1
1	0	1	0	0
1	1	0	1	1
0	0	1	0	1
1	0	1	1	0

- a. 3
b. 4
c. 5
d. 6

5. Un graf neorientat cu 7 noduri, numerotate de la 1 la 7, are mulțimea muchiilor $\{[2, 3], [2, 5], [2, 6], [3, 6], [4, 7], [5, 1], [6, 5]\}$. Indicați numărul de subgrafuri ale sale, pentru fiecare dintre acestea mulțimea muchiilor fiind $\{[2, 3], [2, 6], [3, 6]\}$.

- a. 5
b. 6
c. 7
d. 8

SUBIECTUL al II-lea

(40 de puncte)

- 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 .

 - Scrieți ce se afișează dacă se citește numărul 250887. **(6p.)**
 - Scrieți două numere care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze valoarea citită. **(6p.)**
 - Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
 - Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat a doua structură repetitivă cu o structură repetitivă de alt tip. **(6p.)**
- ```

citește n
(număr natural nenul)
x ← 0; m ← 0; p ← 1
cât timp x < 10 execută
 cn ← n
 cât timp cn ≠ 0 execută
 c ← cn % 10; cn ← [cn / 10]
 dacă c = x atunci
 m ← c * p + m; p ← p * 10
 x ← x + 2
scrie m

```
- Variabila  $m$  memorează simultan numărul de exponate dintr-un muzeu (număr natural din intervalul  $[3, 10^2]$ ) și date despre fiecare exponat (numele și colecția din care face parte, șiruri de cel mult 20 de caractere). Expresiile C/C++ de mai jos au ca valori numărul de exponate, numele celui de al treilea exponat, respectiv colecția din care face parte acesta. Scrieți definiția unei structuri cu eticheta **muzeu**, care permite memorarea datelor despre un muzeu, și declarați corespunzător variabila  $m$ .

$m.numar$        $m.exponat[2].nume$        $m.exponat[2].colectie$  **(6p.)**
  - Variabila  $s$  permite memorarea unui șir de cel mult 20 de caractere.

Scrieți ce se afișează în urma executării secvenței alăturate. **(6p.)**

```

strcpy(s, "muzeu");
s[0] = s[0] + 1;
cout << s[1] << s[0] << endl; | printf("%c%c\n", s[1], s[0]);
strcpy(s, "muzeu" + 2);
cout << s; | printf("%s", s);

```

**SUBIECTUL al III-lea**

**(30 de puncte)**

- Subprogramul **suma** are un singur parametru,  $n$ , prin care primește un număr natural ( $n \in [1, 10^6]$ ). Subprogramul returnează suma divizorilor pozitivi ai lui  $n$  care nu sunt primi. Scrieți definiția completă a subprogramului.

**Exemplu:** pentru  $n=12$  subprogramul returnează 23 ( $23=1+4+6+12$ ). **(10p.)**
- O sală de spectacole are mai multe rânduri de scaune, toate rândurile având același număr de locuri. O persoană poate rezerva mai multe locuri, dar cel mult câte unul pe un rând. Configurația sălii este memorată ca un tablou bidimensional, în care prima **coloană** corespunde primului rând de scaune, iar ultima **coloană** corespunde ultimului rând de scaune. Locurile libere sunt notate cu 0, iar cele rezervate sunt notate cu numere naturale nenule, astfel încât toate locurile rezervate de o persoană sunt notate cu același număr, iar cele rezervate de persoane distincte sunt notate cu numere distincte.

Scrieți un program C/C++ care citește de la tastatură numerele naturale din intervalul  $[2, 20]$ ,  $m$  și  $n$ , apoi  $m \cdot n$  numere naturale din intervalul  $[0, 400]$ , elemente ale tabloului bidimensional cu  $m$  linii și  $n$  coloane, reprezentând configurația sălii. Programul verifică dacă există persoane care au rezervat locuri atât pe primul, cât și pe ultimul rând al sălii, și afișează pe ecran numerele cu care sunt notate aceste locuri, ca în exemplu.

Numerele sunt afișate într-o ordine oarecare, separate prin câte un spațiu, iar dacă nu există astfel de locuri se afișează pe ecran mesajul **nu exista**.

**Exemplu:** pentru  $m=5$ ,  $n=4$  și tabloul alăturat se afișează pe ecran, nu neapărat în această ordine, numerele 6 8 **(10p.)**

|   |   |   |   |
|---|---|---|---|
| 3 | 7 | 0 | 0 |
| 0 | 0 | 5 | 8 |
| 6 | 2 | 0 | 1 |
| 0 | 3 | 0 | 6 |
| 8 | 0 | 6 | 4 |
- Două numere naturale sunt numite **z-prietene** dacă au aceeași cifră a zecilor.

Fișierul **bac.in** conține un șir de cel mult  $10^6$  numere naturale din intervalul  $[10, 10^9]$ , separate prin câte un spațiu. Se cere să se afișeze pe ecran pozițiile din șir pe care se află termeni precedați de un număr maxim de valori  $z$ -prietene cu ei. Numerele afișate sunt separate prin câte un spațiu. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

**Exemplu:** dacă fișierul conține numerele 726 358 98 157 20 49 128 879 659 271 pe ecran se afișează numerele 7 9 (termenii 128, respectiv 659 respectă proprietatea cerută).

  - Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
  - Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**