

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

Testul 14

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 care are valoarea 1 dacă și numai dacă numărul memorat în variabila întregă x aparține intervalului $[-2002, 2020]$.
 - $!(x < -2002 \ || \ x > 2020)$
 - $x > -2002 \ || \ x < 2020$
 - $!(x + 2020 < 0 \ \&\& \ x - 2002 > 0)$
 - $x + 2020 >= 0 \ \&\& \ x - 2002 <= 0$
- Subprogramul f este definit alăturat. Valoarea lui $f(2020, 20)$ este:

```
int f(int x, int y)
{ if(x<=0) return y;
  if(y<=0) return x;
  return f(x/y, x-y);
}
```

 - 0
 - 1
 - 1899
 - 2000
- Utilizând metoda backtracking se generează toate grupele de accesorii pentru înot din mulțimea {cască, costum de înot, cleme, dopuri de urechi, plută, mască de snorkeling, ochelari, snorkel}. Accesoriile au prețurile următoare, exprimate în lei: cască - 40, costum de înot - 400, cleme - 30, dopuri de urechi - 30, plută - 10, mască de snorkeling - 400, ochelari - 70, snorkel - 60. Într-o grupă accesoriiile sunt distincte, nu contează ordinea lor și costă, în total, exact 500 de lei. Primele trei soluții generate sunt, în această ordine: (cască, costum de înot, cleme, dopuri de urechi), (cască, costum de înot, snorkel), (cască, cleme, dopuri de urechi, mască de snorkeling). A cincea soluție generată este:
 - (cască, mască de snorkeling, snorkel)
 - (costum de înot, cleme, plută, snorkel)
 - (cleme, mască de snorkeling, ochelari)
 - (dopuri de urechi, plută, mască de snorkeling, snorkel)
- Într-un arbore cu rădăcină cu 6 noduri, numerotate de la 1 la 6, sunt 3 noduri de tip „frunză”. Indicați un vector de „tați” care poate corespunde arborelui.
 - (0, 3, 1, 6, 6, 2)
 - (2, 0, 1, 1, 2, 1)
 - (3, 5, 5, 6, 0, 5)
 - (5, 5, 1, 1, 0, 1)
- Un graf orientat cu 6 vârfuri, numerotate de la 1 la 6, este reprezentat prin matricea de adiacență alăturată. Indicați numărul minim de arce care pot fi adăugate astfel încât graful obținut să aibă cel puțin două circuite. Un circuit este format numai din arce distincte, iar două circuite sunt distincte dacă diferă prin cel puțin un arc.

```
0 0 0 0 0 1
1 0 1 0 1 0
0 0 0 1 1 0
0 0 0 0 0 1
0 0 0 0 0 0
0 0 0 0 0 0
```

 - 1
 - 2
 - 3
 - 4

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 valoarea afișată dacă se citește numărul 1527894. (6p.)
- b. Scrieți trei numere care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului să se afișeze 1. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat structura **repetă...până când** cu o structură repetitivă cu test inițial. (6p.)
- ```

citește n (număr natural)
m ← 0
repetă
 c ← n % 10; n ← [n/10]
 dacă c = 9 atunci c ← 1
 altfel
 dacă c % 2 = 1 atunci
 c ← c + 1
 ■
 m ← m * 10 + c
până când n = 0
scrie m

```
2. Variabila  $r$  memorează simultan următoarele date despre fiecare dintre cele 20 de rase de porumbei dintr-o volieră: un cod, reprezentând rasa, numărul de exemplare din rasa respectivă și vârstele acestora. În volieră sunt maximum 10 exemplare din fiecare rasă. Știind că expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul și numărul de exemplare din prima rasă, respectiv vârsta celui de al 4-lea exemplar din această rasă, scrieți definiția unei structuri cu eticheta **rasa**, care permite memorarea datelor despre o rasă de porumbei, și declarați corespunzător variabila  $r$ .
- ```

r[0].cod      r[0].nrExemplare      r[0].varsta[3]
    
```
- (6p.)
3. Variabilele i și j sunt de tip întreg, iar variabila a memorează un tablou bidimensional cu 5 linii și 7 coloane, numerotate începând de la 0, cu elemente numere întregi, inițial toate egale cu 0.
- Fără a utiliza alte variabile decât cele menționate, scrieți o secvență de instrucțiuni prin care se transformă în memorie tabloul, astfel încât orice element aflat pe prima linie sau pe ultima coloană să aibă valoarea 1 și oricare alt element din tablou să fie egal cu ultima cifră a sumei celor două elemente alăturate lui, aflate pe aceeași linie dar pe coloana din dreapta, respectiv pe aceeași coloană, dar pe linia anterioară. (6p.)
- ```

1 1 1 1 1 1 1
7 6 5 4 3 2 1
8 1 5 0 6 3 1
4 6 5 0 0 4 1
0 6 0 5 5 5 1

```

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Două numere  $a$  și  $b$  ( $a < b$ ) sunt numite **divizori pereche** ai unui număr natural  $n$  dacă  $a \cdot b = n$ . Subprogramul **perechi** are un singur parametru,  $n$ , prin care primește un număr natural ( $n \in [2, 10^9]$ ). Subprogramul afișează pe ecran toate perechile distincte de numere naturale cu paritate diferită cu proprietatea că sunt divizori pereche ai lui  $n$ . Fiecare pereche este afișată încadrată între paranteze drepte, numerele din pereche fiind afișate în ordine strict crescătoare, separate printr-un spațiu, iar dacă nu există astfel de perechi, se afișează pe ecran mesajul **nu exista**. Scrieți definiția completă a subprogramului. **Exemplu:** dacă  $n=12$  se afișează pe ecran, nu neapărat în această ordine, [1 12] [3 4] iar dacă  $n=9$  se afișează pe ecran **nu exista** (10p.)
2. Un text are cel mult 100 de caractere și este format din cuvinte și numere, separate prin câte un spațiu. Cuvintele sunt formate numai din litere ale alfabetului englez. Toate numerele sunt reale și sunt formate numai din parte întreagă sau din parte întreagă și parte fracționară, separate prin virgulă (,), numerele negative fiind precedate de semnul minus (-). Cel puțin unul dintre numerele reale este negativ. Scrieți un program C/C++ care citește de la tastatură textul, pe care îl transformă în memorie, înlocuind fiecare număr negativ cu valoarea sa absolută. Programul afișează apoi pe ecran textul obținut. **Exemplu:** pentru textul **Modul de -3,24 este 3,24 si modul de -15 este 15** se va afișa pe ecran textul: **Modul de 3,24 este 3,24 si modul de 15 este 15** (10p.)
3. Fișierul **bac.in** conține un șir de numere naturale distincte, din intervalul  $[1, 10^9]$ . Numerele din șir sunt separate prin câte un spațiu și cel puțin trei dintre ele au penultima cifră 2 și ultima cifră 0. Se cere să se afișeze pe ecran cele mai mici trei numere din șir cu proprietatea că au penultima cifră 2 și ultima cifră 0. Numerele determinate sunt afișate în ordine crescătoare, separate prin câte un spațiu. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare. **Exemplu:** dacă fișierul conține numerele 9731 50 112 20 8 16 8520 3 2520 1520 pe ecran se vor afișa, în această ordine, numerele: 20 1520 2520
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)