

Examenul național de bacalaureat 2022

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

Varianta 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 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. Indicați o expresie care are valoarea 1 dacă și numai dacă expresia C/C++ alăturată are valoarea 1.

a. **!(x<3 && x>=10)**

c. **x<3 && !(x>=10)**

b. **!(x<3 || x>=10)**

d. **!(x<3) || x>=10**
- Subprogramul **f** este definit incomplet, alăturat. Indicați o expresie cu care pot fi înlocuite punctele de suspensie pentru ca **f(n, 3)** să aibă valoarea 1 pentru orice număr prim **n** ($n \in [2, 10^4]$) și valoarea 0 în caz contrar.

a. **y+2**

b. **y-2**

c. **y*2**

d. **y/2**

```
int f(int x, int y)
{ if(x!=2 && x%2==0) return 0;
  if(y*y>x) return 1;
  if(x*y==0) return 0;
  return f(x, .....);
}
```
- Utilizând metoda backtracking, se generează toate posibilitățile de a forma platouri de câte trei fructe distincte din mulțimea {**măr, gutuie, prună, caisă, piersică**}, astfel încât pe același platou nu se află o gutuie și o piersică simultan. Două platouri sunt distincte dacă ele conțin cel puțin un fruct diferit. Primele patru soluții generate sunt, în această ordine, (**măr, gutuie, prună**), (**măr, gutuie, caisă**), (**măr, prună, caisă**), (**măr, prună, piersică**). Indicați soluția generată imediat după (**gutui**e, **prună**, **caisă**).

a. (**caisă, piersică, măr**)

c. (**mar, caisă, piersică**)

b. (**gutui**e, **prună, piersică**)

d. (**prună, caisă, piersică**)

SUBIECTUL al II-lea

(40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.**

a. Scrieți ce se afișează în urma executării algoritmului dacă se citesc, în această ordine, valorile **21** și **47**. (6p.)

b. Dacă pentru variabila **m** se citește numărul **5**, scrieți două valori care pot fi citite pentru variabila **n** astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, valoarea afișată să fie **10**. (6p.)

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

d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat prima structură **cât timp...execută** cu o structură repetitivă cu test final. (6p.)
- Un arbore cu **9** noduri, numerotate de la **1** la **9**, are muchiile **[1, 8]**, **[2, 3]**, **[2, 5]**, **[2, 7]**, **[4, 5]**, **[5, 6]**, **[5, 8]**, **[8, 9]**. Scrieți două noduri care pot fi alese drept rădăcină, astfel încât nodul **5** să aibă doi "frați". (6p.)
- Variabila **x**, declarată alăturat, memorează simultan un an calendaristic și temperatura medie a aerului, măsurată în grade Celsius, înregistrată în București în anul respectiv.

Scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran mesajul **CALDUROS** dacă temperatura menționată a fost strict mai mare decât **11°C**, mesajul **RACOROS** dacă temperatura menționată a fost strict mai mică decât **10°C**, sau mesajul **NORMAL** în orice alt caz. (6p.)

```

citește m,n
(numere naturale nenule, m<n)
x←1
cât timp x=1 execută
    x←m; y←n; n←n-1
    cât timp x≠y execută
        dacă x>y atunci x←x-y
        altfel y←y-x
    x
scrie n+1

```

```

struct meteo
{
    int an, temperatura;
}x;

```

SUBIECTUL al III-lea

(30 de puncte)

- Subprogramul **schimb** are trei parametri:

 - n** și **x**, prin care primește câte un număr natural ($n \in [0, 10^8]$, $x \in [1, 9]$);
 - p**, prin care primește un număr natural reprezentând poziția unei cifre a numărului **n** ($0 \leq p$). Pozițiile cifrelor sunt numerotate de la dreapta la stânga, astfel: cifra unităților este pe poziția **0**, cifra zecilor este pe poziția **1** ș.a.m.d.

Subprogramul transformă numărul **n**, înlocuind cifra de pe poziția **p** cu cifra **x**, și furnizează numărul obținut tot prin parametrul **n**. Scrieți definiția completă a subprogramului.

Exemplu: dacă **n=12587**, **x=6** și **p=3**, după apel, **n=16587**. (10p.)
- Se consideră o vocală oarecare a alfabetului englez, notată cu **v**, și o consoană oarecare a alfabetului englez, notată cu **c**. Litera **v** se numește **vocală prietenă** a lui **c** dacă în șirul literelor alfabetului englez, ordonat lexicografic, **v** îl precede pe **c**, iar între **v** și **c** nu există nicio vocală. Se consideră vocale literele **a, e, i, o, u**.

Exemplu: **e** este vocală prietenă pentru consoanele **f, g** și **h**, dar nu este vocală prietenă pentru consoanele **d** și **j**.

Un elev vrea să transmită unui prieten o parolă, codificată. Parola este formată dintr-un singur cuvânt de cel mult **50** de caractere, litere mici ale alfabetului englez, cel puțin una fiind consoană. Codificarea se face prin înlocuirea fiecărei consoane cu vocala sa prietenă, ca în exemplu.

Scrieți un program C/C++ care citește de la tastatură un cuvânt, reprezentând o parolă de tipul precizat și determină, în memorie, forma codificată a acesteia. Programul afișează pe ecran parola codificată obținută.

Exemplu: pentru parola rame se afișează oaie, iar pentru parola sport se afișează oooo (10p.)
- Numim **secvență paritară** a unui șir de numere naturale un subșir al acestuia, format din termeni cu aceeași paritate, aflați pe poziții consecutive în șirul dat. Lungimea secvenței este egală cu numărul de termeni ai acesteia.

Fișierul **bac.txt** conține un șir de cel puțin două și cel mult **10⁶** numere naturale din intervalul **[0, 10⁹]**. Numerele sunt separate prin câte un spațiu, iar în șir există cel puțin doi termeni cu aceeași paritate pe poziții consecutive.

Se cere să se afișeze pe ecran numărul secvențelor paritare de lungime maximă din șirul aflat în fișier, precum și această lungime maximă. Numerele afișate sunt separate printr-un spațiu. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.

Exemplu: dacă fișierul conține numerele
2 3 5 1 7 9 8 4 4 11 15 17 21 11 6 11 15 17 21 11 6 5 2 6 4 0 16
 atunci pe ecran se afișează valorile **4 5**

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

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