

Examenul de bacalaureat național 2020

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

Testul 4

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.

I. TÊTEL

(20 pont)

Îrja a vizsgalapra az 1 és 5 közti feladatokra adott helyes válaszoknak megfelelő betűt. Mindenik helyes válasz 4 pontot ér.

- Az x, y és z egész típusú változók. Határozzátok meg azt a C/C++ kifejezést melynek az értéke 1 akkor és csakis akkor, ha az x, y és z változók azonos értékűek.
 - $x==y \ \&\& \ x==z$
 - $x==y==z$
 - $!(x!=y \ \&\& \ x!=z)$
 - $!(x!=y!=z)$
- Mellékelten definiáltuk az f alprogramot.
Az $f(20,2020)$ hívásra a függvény végrehajtásának száma:

```
int f(int x, int y)
{
    if(x<=1 || y<=1) return 0;
    if(x>y) return 1+f(f(x/y,y),y);
    return 1+f(x,f(x,y/x));
}
```

 - 5
 - 9
 - 11
 - 20
- A backtracking módszert használva generáljuk az összes palacsintakészítés lehetőségét felhasználva egy megadott sorrendben minden hozzávalót a {făină, lapte, ouă} halmazból a tésztához, majd egy elemet a {ciocolată, dulceață, urdă} halmazból a töltelékhez, végül egy elemet a {cașcaval, mărar, frișcă} halmazból a díszítéshez. Figyelembe kell vegyük a következő szabályokat: a frișcă csak a ciocolată és dulceața elemmel együtt használható, míg a mărar és cașcaval csak az urda elemmel együtt. Az első 5 megoldás, a generálás sorrendjében: (făină, lapte, ouă, ciocolată, frișcă), (făină, lapte, ouă, dulceață, frișcă), (făină, lapte, ouă, urdă, cașcaval), (făină, lapte, ouă, urdă, mărar), (făină, ouă, lapte, ciocolată, frișcă). Határozzátok meg a hetedik generált megoldást.
 - (ouă, lapte, făină, urdă, mărar)
 - (lapte, făină, ouă, ciocolată, frișcă)
 - (făină, ouă, lapte, dulceață, frișcă)
 - (făină, ouă, lapte, urdă, cașcaval)
- Egy gyökeres fa magasságának nevezzük azon élek számát, amelyek a gyökértől a fa leveleihez vezető leghosszabb sétát alkotják.
Adott egy 8 csomópontot tartalmazó gyökeres fa, melyben a csomópontokat 1-től 8-ig sorszámozzuk. A fa ősvektora, az "apák" tömbje: (6, 6, 5, 3, 0, 5, 8, 4). A fa magassága:
 - 2
 - 3
 - 4
 - 5
- Adott egy 10 csomópontot tartalmazó irányítás nélküli gráf, csomópontjait 1-től 10-ig sorszámozzuk, élei: [1,2], [1,3], [1,10], [3,10], [4,5], [4,6], [4,8], [5,7], [5,9], [6,8], [6,9], [7,9], [8,9]. Határozzátok meg legkevesebb hány élt kell hozzáadni ahhoz, hogy a létrejött gráfban legyen legkevesebb egy euler-séta (olyan séta, amely áthalad a gráf összes élén).
 - 1
 - 2
 - 3
 - 4

II. TÉTEL

(40 pont)

- Adott a mellékelt algoritmus pszeudokódban.**
Az $a \% b$ jelöli az a természetes szám b nem nulla természetes számmal való osztási maradékát, és $[c]$ a c valós szám egész részét.

a. Írjátok le mit ír ki, ha a beolvasott szám 49335. (6p.)

b. Adjatok meg három olyan számot, amelyeket beolvashatunk, és ha elvégezzük az algoritmust, mindegyik esetében a 1100 érték lesz kiírva. (6p.)

c. Írjátok az adott algoritmusnak megfelelő C/C++ programot. (10p.)

d. Írjátok meg azt az eredetivel egyenértékű algoritmust pszeudokódban, amelyikben az **amíg... végezd el** struktúrát helyettesítitek egy másik ismétlődő struktúrával. (6p.)
- A p változó egy szabályos poliéder adatait tárolja: a csúcsok számát (**NrVarfuri**), az él hosszát (**Muchie**) és két speciális szöveget (egy lap és egy él közötti szög illetve két lap közötti szög). Tudjuk, hogy az alábbi C/C++ kifejezések értékei: egy természetes szám, amely a poliéder csúcsainak számát, és valós számok, amelyek a poliéder éleinek a számát illetve a két speciális szög mértékét tárolják. Írjátok meg a **poliedru** címkéjű struktúra definícióját, úgyhogy, abban tárolhassuk a rögzített adatokat, és deklaráljátok helyesen a p változót. (6p.)

p.NrVarfuri p.Muchie p.Unghi.FataMuchie p.Unghi.FataFata
- Az i és j változó egész típusú, az s változó egy olyan karakterlánc, amely leg több 20 karaktert tárolhat. Írjátok le, milyen karakterlánc lesz az s változóban a következő programrész végrehajtása után. (6p.)

```
strcpy(s, "optsprezece"); i=0; j=strlen(s)-1;
while(i<j)
{ if(strchr("aeiou", s[i])==NULL && strchr("aeiou", s[j])!=NULL)
{ s[i]=s[i]+1; s[j]=s[j]-1; }
i=i+1; j=j-1;
}
```

III. TÉTEL

(30 pont)

- Az a és b számokat az n természetes szám **generátorának** nevezzük, ha $a \cdot b + [a/b] = n$, ahol $[c]$ a c valós szám egész részét jelöli.
A **generatoare** alprogramnak egyetlen paramétere van n , melyen keresztül egy természetes számot kap ($n \in [2, 10^9]$). Az alprogram kiírja a képernyőre az összes különböző számpárt, amelyek olyan természetes számokból állnak melyek az n szám generátorai lehetnek, és az első közülük páros. A számpárok tagjai mínusz (-) jellel vannak elválasztva, a párok pedig szóközzel vannak elválasztva egymástól. Ha nem létezik ilyen számpár, akkor a **nu exista** üzenetet írja ki. Írjátok meg az alprogram teljes definícióját.
Például: ha $n=2020$ a képernyőn megjelenik:
2-1010 4-505 10-202 20-101 96-21 200-10 606-3 808-2 1010-1 (10p.)
- Egy kétdimenziós tömb elemei csak a $\{0, 1\}$ halmazból vehetnek fel értékeket. „Komplementáris” oszlopoknak nevezünk két olyan oszlopot, amelyeknek bármely két eleme, amely azonos soron található különböző értékű.
Írjátok egy C/C++ programot, amely beolvas két m és n természetes számot a $[2, 20]$ intervallumból, majd egy m sorból és n oszlopból álló kétdimenziós tömb elemeit, melyek természetes számok a $\{0, 1\}$ intervallumból.
A program kiírja a tömb első oszlopával „komplementáris” oszlopok számát.
Például: ha $m=3$, $n=6$, a mellékelt tömb esetén a kiírt szám: 3. (10p.)

1	1	0	0	1	0
0	1	1	1	1	1
1	0	0	0	1	0
- A **bac.txt** állomány legkevesebb két és leg több 10^6 természetes számot tartalmaz. A számok értékei a $[0, 10^9]$ intervallumból vannak, **csökkenő sorban** egy-egy szóközzel elválasztva helyezkednek el. Jelenítse meg a képernyőn azokat a számokat, amelyek pontosan kétszer találhatók meg az állományban, szigorúan csökkenő sorban egy-egy szóközzel elválasztva. Ha nincs egyetlen ilyen szám sem, akkor a **nu exista** üzenet jelenjen meg a képernyőn. Tervezzetek a futási idő és a memóriahasználat szempontjából hatékony algoritmust.
Például: ha az állományban található számok 100 50 50 50 49 49 36 16 16 12 10 10 9 7 7
A képernyőn megjelennek a 49 16 10 7 számok ebben a sorrendben
a. Írjátok a tervezett algoritmusnak megfelelő C/C++ programot. (8p.)
b. Írjátok le saját szavaitokkal a használt algoritmust, és indokoljátok annak hatékonyságát. (2p.)

