

Examenul de bacalaureat național 2019
Proba E. d)
Informatică
Limbajul C/C++

Varianta 2

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)

Az 1-től 5-ig számozott ítemek esetén, írja a vizsgalapra a helyes válasznak megfelelő betűt. Minden helyes válasz 4 pontot ér.

1. Egy C/C++ kifejezés, amelynek értéke 0:

a. 'm' < 'n'

b. 'm' == 'M'

c. 'M' == 'm' + 'N' - 'n'

d. 'N' == 'M' + 1

2. Az **f** alprogram meghatározása mellékelve van. Adja meg, mit jelenít meg az alábbi híváskor:
f(75, 30);

```
void f(int i, int j)
{ cout<<i<<" "; | printf("%d ",i);
  if(i!=j)
  { if(i<j){ i=i+j; j=i-j; i=i-j; }
    f(i-j,j);
  }
}
```

a. 75 30 45

b. 75 45 15

c. 75 45 15 15

d. 75 30 45 15

3. Használva a backtracking módszert, generálja le az összes tortatípust, melyek három különböző krémréteggel rendelkeznek a következő halmazból {**caramel**, **ciocolată**, **frișcă**, **nuci**, **vanilie**}. Az utolsó réteg vagy tejszín, vagy vanília, valamint ezek az első két rétegben nem lehetnek. Két torta, amelyekben a rétegek ugyanolyan típusú krémeket tartalmaznak különbözőek, ha ezek a krémelek más sorrendben találhatók. Az első négy legenerált eredmény a következő, a megadott sorrendben: (**caramel**, **ciocolată**, **frișcă**), (**caramel**, **ciocolată**, **vanilie**), (**caramel**, **nuci**, **frișcă**), (**caramel**, **nuci**, **vanilie**). Ekkor az ötödik megoldás:

a. (**ciocolată**, **caramel**, **frișcă**)

b. (**ciocolată**, **nuci**, **frișcă**)

c. (**nuci**, **caramel**, **vanilie**)

d. (**nuci**, **ciocolată**, **vanilie**)

4. Egy 4 élű fa csomóinak száma:

a. 2

b. 3

c. 4

d. 5

5. Egy 6 csomóból álló irányítatlan gráf esetén a csomók fokszámait ábrázoló értékek a következők lehetnek:

a. 1, 0, 0, 2, 2, 2

b. 2, 2, 3, 4, 0, 3

c. 2, 2, 5, 5, 0, 1

d. 6, 5, 4, 3, 2, 1

II. TÉTEL

(40 punct)

1. A mellékelt algoritmus pszeudokódban van ábrázolva.

Az $a \% b$ az a természetes számnak a b nullától különböző természetes számmal való osztási maradékát és $[c]$ a c valós szám egész részét jelöli.

- Írja le a kiírt értéket, ha a beolvasott szám 27102. (6p.)
- Írjon három különböző számot a $[10, 9999]$ intervallumból, amelyek beolvashatók, úgy, hogy az algoritmus végrehajtása után, mindegyik szám esetén a megjelenített érték 11 legyen. (6p.)
- Írjon az algoritmusnak megfelelő C/C++ programot. (10p.)
- Írjon pszeudokód algoritmust, amely egyenértékű a megadottal, úgy, hogy helyettesítse az első **amíg...végezd el** struktúrát egy **minden...végezd el** típusú struktúrára. (6p.)

```
beolvas n
    (természetes szám)
m ← 0; x ← 1
amíg x ≤ 9 végezd el
    cp ← n
    amíg cp ≠ 0 végezd el
        ha cp % 10 = x akkor
            m ← m * 10 + x
        cp ← [cp / 10]
    x ← x + 1
kiír m
```

- Adott két a és b karaktersorozat, az a -t a b **sufix**-ének nevezzük, ha a egyenlő b -vel, vagy, ha a b -t megkapjuk az a -ból, ha ennek a bal részére újabb karaktereket ragasztunk. Az a és b változók tárolhatnak egy - egy leg több 20 karakterből álló sorozatot. Írjon egy utasítássort, melynek a végrehajtása után az a változó a b egy **sufix**-ét tárolja, három karakterből alkotva, vagy a **nedeterminat** sorozatot, ha nem létezik ilyen sufix.

Példa: ha b a centaur sort tárolja, akkor az a az aur sort tartalmazza, vagy ha a b az au, sort, akkor az a a **nedeterminat** sort. (6p.)

- A mellékelt deklarálások esetén, a p változó az xOy koordináta rendszer egy pontjának koordinátáit tárolja, valamint a c változó egy kört jellemző adatokat tárolja: a sugár hosszát és a középpontjának a koordinátáit. Írjon egy kifejezést, amely értéke 1, akkor és csakis akkor, ha a p változóban tárolt pont a c változóban tárolt középpontnak felel meg. (6p.)

```
struct punct
{
    int x, y;
} p;
struct cerc
{
    struct punct centru;
    float raza;
} c;
```

III. TÉTEL

(30 punct)

- A **MaxImp** alprogramnak két paramétere van, a és b , amelyek által egy-egy természetes számot kap ($2 \leq a < b \leq 400$). Az alprogram visszatéríti a legnagyobb természetes számot az $[a, b]$ intervallumból, amelynek esetében a pozitív páratlan osztók szorzata szigorúan nagyobb nála, vagy 0 ha nincs egyetlen ilyen szám sem. Írja le a teljes alprogramot.

Példa: ha $a=14$ és $b=19$, akkor a visszatérített érték 18 ($1 \cdot 3 \cdot 9 = 27 > 18$). (10p.)

- Egy kétdimenziós tömb m méretű **négyzetének** nevezzük, azt a tömböt, amelyet az eredeti tömbből úgy kapunk, hogy csak az első m soron és m oszlopon található elemeket tartjuk meg. Írjon egy C/C++ programot, amely beolvas a billentyűzetről egy n ($n \in [2, 20]$) természetes számot, aztán az n soros és n oszlopos tömb elemeit, amelyek természetes számok a $[0, 10^4]$ intervallumból.

A program meghatározza a beolvasott tömb egy olyan maximális méretű négyzetét, amelynek az összes eleme egyenlő, és kiírja a képernyőre e dimenzió értékét.

Példa: $n=5$ és a mellékelt tömb esetében, a képernyőre kiírt érték 3. (10p.)

```
2 2 2 2 2
2 2 2 2 8
2 2 2 2 2
2 1 2 8 7
3 5 2 1 2
```

- A **bac.txt** állomány legkevesebb két és leg több 10^6 természetes számot tartalmaz a $[0, 10^3]$ intervallumból, a számok szóközzel vannak elválasztva egymástól. A sorozatnak legalább egy páros és legalább egy páratlan eleme van. Írassuk ki a képernyőre a sorozat tagjait, szóközzel elválasztva, úgy hogy a páratlan tagok a páros tagok előtt jelenjenek meg, és úgy a páratlan tagokból álló, mint a páros tagokból álló részsorozat elemei, növekvő sorrendben jelenjenek meg, ahogy a példa mutatja. Tervezzon a végrehajtási idő szempontjából hatékony algoritmust.

Példa: ha az állomány tartalma 12 2 3 1 2 5

a képernyőre kiírt értékek: 1 3 5 2 2 12

- Írja le saját szavaival a használt algoritmust és indokolja annak hatékonyságát. (2p.)
- Írja meg a leírt algoritmusnak megfelelő C/C++ programot. (8p.)