

II.TÉTEL

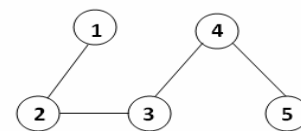
(30 pont)

Az 1-es és 2-es item esetén írja a vizsgalapra a helyes válasznak megfelelő betűt.

1. Adott egy 7 csúcsot tartalmazó irányított gráf, amelyben a csúcsok 1-től 7-ig vannak számozva, az $(1,2)$, $(2,3)$, $(3,4)$, $(4,1)$, $(5,4)$, $(7,5)$, $(5,6)$ élekkel. A gráf azon csúcsainak a száma, amelyek esetén a befoka egyenlő a kifokával: (4p.)

a. 2 b. 3 c. 4 d. 5

2. Adott a mellékelt ábrán látható, öt csúcsot tartalmazó irányítás nélküli gráf. Legkevesebb hány élet kell hozzáadni a gráfhoz ahhoz, hogy a kapott gráf bármely két különböző csúcsa között létezzen legalább egy, 2 hosszúságú út: (4p.)



a. 1 b. 2 c. 3 d. 4

Írja a vizsgalapra a következő feladatok megoldásait.

3. Adottak a mellékelt deklarációk, amelyben az **s** változó az **xOy** koordináta-rendszerben megadott szakasz végpontjainak koordinátáit tárolja. Írjon egy olyan utasítást, amely kiírja a képernyőre az **s** változóban tárolt szakasz hosszának a négyzetét. (6p.)

```
struct punct {
    float x,y;
};
struct segment {
    punct A,B;
}s;
```

4. Az alábbi utasítássorozatban, az **i** és **j** változók egész típusúak, az **A** változó pedig egy 5 sor és 5 oszlopot tartalmazó kétdimenziós tömböt tárol. A sorok és oszlopok 1-től 5-ig vannak sorszámozva, a tömb elemi egész számok.

Más változók használata nélkül, írjon egy vagy több utasítást, amelyeket behelyettesítve a pontozott sor helyére, az utasítássorozat végrehajtása után, az **A** változóban tárolt tömb a mellékelt ábrán látható elemeket tartalmazza.

```
for(i=1;i<=5;i++)
    for(j=1;j<=5;j++)
        .....
(6p.)
```

2	3	4	0	1
3	4	0	1	2
4	0	1	2	3
0	1	2	3	4
1	2	3	4	0

5. Adott két karakterlánc, az **a** hossza **na** és **b** hossza **nb**. Az **a** karakterláncot a **b** karakterlánc **prefix**-ének nevezzük, ha $na \leq nb$ és a **b** karakterlánc első **na** karaktere megegyezik az **a**-val. Az **a** karakterláncot a **b** karakterlánc **suffix**-ának nevezzük, ha $na \leq nb$ és a **b** karakterlánc utolsó **na** karaktere megegyezik az **a**-val.

Írjon egy C/C++ programot, amely beolvasson két, legfeljebb 30 karakterből álló szót, amelyek csak az angol ábécé kisbetűit tartalmazhatják. Az első szó után Entert ütünk. A program kiírja a képernyőre, az első szó minden sufixát, amelyek prefixei a második szónak, egy-egy szóközzel elválasztva, amint a példa mutatja. Ha nem léteznek ilyen részlánckok, akkor a képernyőre a **NU EXISTA** üzenetet kell írni.

Példák:

ha a beolvasott szavak, a megadott sorrendben

rebele
elegant

a képernyőre kiírt részlánckok (nem feltétlenül ebben a sorrendben): e ele

Illetve, ha a beolvasott szavak, a megadott sorrendben

strai
rai

a kiírt részlánck rai

(10p.)

III. TÉTEL

(30 pont)

Az 1-es item esetén írja a vizsgalpra a helyes válasznak megfelelő betűt.

1. A backtracking módszert használva, előállítjuk lexikografikus sorrendben az összes, a $\{0, 1, 2\}$ halmazból vett 5 számjegyből álló sorozatot, azzal a tulajdonsággal, hogy nincs két páros számjegy egymás utáni helyen. Az első nyolc előállított sorozat, ebben a sorrendben a következő: 01010, 01011, 01012, 01101, 01110, 01111, 01112, 01121. A kilencedik sorozat: (4p.)
- a. 01120 b. 01201 c. 01210 d. 10101

Írja a vizsgalpra a következő feladatok megoldásait.

2. Adott a mellékelt `f` alprogram. Írja le, hogy mi lesz kiírva a következő meghívás esetén: (6p.)
- ```
f(9);
```

```
void f(int x)
{ cout<<x; | printf("%d",x);
 if(x>0)
 { if(x%2==0)
 cout<<'*'; | printf("");
 f(x/2);
 }
}
```
3. Adott a `double` alprogram, a következő két paraméterrel:
- `n`, amelyben egy  $2 < n < 50$  feltételnek megfelelő természetes számot kap;
  - `v`, amelyben egy egydimenziós tömböt kap, amely `n` darab, legtöbb 4 számjegyből álló egész számot tartalmaz. A tömb legalább egyik eleme szigorúan pozitív.
- Az alprogram a tömb minden szigorúan pozitív eleme után, beszúr egy vele azonos értékű elemet, a példához hasonlóan. A megváltoztatott tömb, valamint az `n` aktualizált értéke, szintén a `v`, illetve `n` paramétereken keresztül lesznek visszaadva. Írja le az alprogram teljes definícióját.
- Példa:** ha `n=6` és `v=(4, -5, 0, 9, 9, -2)`, akkor hívás után `n=9`, illetve `v=(4, 4, -5, 0, 9, 9, 9, 9, -2)`. (10p.)
4. Egy legalább két számjegyből álló természetes számot **x-rendezettnek** nevezünk, ha az összes számjegye növekvő sorrendben van, és bármely két egymásután következő számjegye különbségének abszolút értéke egyenlő `x`-szel.
- Példák:** a 2468 szám 2-rendezett, a 147 szám 3-rendezett; a 179 és 131 számok nem tartoznak az említett típushoz.
- Olvasson be a billentyűzetről egy `x` ( $1 \leq x \leq 8$ ) természetes számot és írassa a `BAC.TXT` állományba az összes különböző **x-rendezett** természetes számot. Minden szám legyen az állomány külön sorába írva.
- A kért számok meghatározására, a végrehajtási idő szempontjából hatékony algoritmust használjon.
- a) Írja le saját szavaival a használt algoritmust, és indokolja meg a hatékonyságát. (4p.)
- b) Írja le az előbb leírt algoritmusnak megfelelő C/C++ programot. (6p.)