

**III. Tétel (30 pont)**

**Az 1-es feladat esetén a helyes válasz betűjelét írja a vizsgalapra.**

1. A backtracking módszert alkalmazva, generáljuk az  $\{1,2,3,4\}$  halmaz összes permutációját. Ha az első három permutáció sorban: 1234, 1243, 1324, határozza meg, hogy melyik lesz a 3412 után generált permutáció. **(4p.)**
- a. 3421                      b. 3413                      c. 4123                      d. 3214

**Az alábbi feladatok esetén a választ írja a vizsgalapra.**

2. Adott az `f` alprogram mellékelt definíciója. Mi lesz `int f(int x)` az `f(7)` illetve az `f(100)` értéke? **(6p.)**
- ```
{if(x%6==0)return x;  
  else return f(x-1);  
}
```
3. Írja meg a `P`, háromparaméteres alprogram teljes definícióját, amely az első `a` paraméterben megkap egy maximum 100 elemű egydimenziós tömböt, amelyben az elemek legfeljebb 4 számjegyű egész számok, a második `n` paraméterben megkapja a tömb elemeinek tulajdonképpen számát, valamint a `k` paraméterben egy természetes számot ( $k < 101$ ), és visszatéríti azt a legnagyobb összeget, amelyet a tömb `k` darab eleméből lehet alkotni.
- Példa:** ha  $n=6$  és  $k=4$ , valamint a sorozat elemei (5, 2, 5, 4, 1, 3), akkor a hivatkozás után visszatérített érték 17. **(10p.)**
4. A `numere.txt` szöveges állomány első sorában egy  $n$  ( $0 < n < 100000$ ) természetes szám van, a második sorában  $n$  darab, egyenként legfeljebb 2 számjegyből álló természetes szám, egy-egy szóközzel elválasztva.
- a) Írjon egy C/C++ programot, amely a végrehajtási idő szempontjából hatékony algoritmus szerint meghatározza azokat a számokat, amelyek csak egyszer szerepelnek az állomány második sorában. Az így meghatározott számokat növekvő sorrendben, egy-egy szóközzel elválasztva ki kell írni a képernyőre.
- Példa:** a `numere.txt` állomány alábbi tartalma esetén:
- ```
7  
3 5 2 1 5 23 1
```
- a képernyőre kiírt számok: 2 3 23. **(6p.)**
- b) Írja le saját szavaival az alkalmazott módszert, megmagyarázva annak hatékonyságát. (3-4 sor). **(4p.)**