

## Mintazh az 1. témakörhöz

### 1. feladat (8p)

a) A definíció alapján mutassuk meg, hogy:

$$\lg(n!) = O(n \lg(n))$$

b) Állítsa aszimptotikusan **nővekvőleg** a következő függvényeket, az egyenlőséget is jelölje.

$$3^n - 1, \quad 2n! + 3, \quad 2^{n+2}, \quad (n+1)\log_2 n, \quad 6n^{0.3} + 2n, \quad \lg(n^{0.1} - 100)$$

### 2. feladat (16p)

Egy prioritásos sort **kétirányú, rendezett, fejelemes** listával ábrázolunk. A lista egy elme (kulcs, prioritás, előremutató, visszamutató) négyesből áll. A lista prioritás szerint csökkenőleg rendezett. Adja meg a következő műveletek struktogramját: Üres(PS)-létrehozza az üres sort, Betesz(PS,kulcs,prior)-beteszi a sorba az adott kulcsú és prioritású elemet, Kivesz(PS,kulcs)-kiveszi a legnagyobb prioritású elemet, ha a sor üres, jelezzen hibát, PriorNövel(kulcs,érték)-megnöveli az adott kulcsú elem prioritását az adott értékkel, ha nincs ilyen elem, adjon hibaüzenetet. Növelés után a sor karbantartását el kell végezni (rendezettséget helyreállítani)!

### 2. feladat (másik variáció) (16p)

A zsák adatszerkezetet **kétirányú, fejelemes, érték szerint nővekvőleg rendezett** listával ábrázoljuk. A zsák abban tér el a halmaztól, hogy egy adott értékből több példány is lehet a zsákban. A lista egy eleme a következő négyesből áll: **(érték, multiplicitás, vmut, emut)**. Az egyik zsák fejelemére az ZS1, a másakra az ZS2 pointer mutat. Készítsen algoritmust, mely ZS1 zsákból „kivonja” a ZS2 zsákot. (Az azonos értékű elemek esetén ZS2-beli elem multiplicitásával csökkenteni a ZS1-beli elem multiplicitását, ha így 0 vagy negatív darabszámot kapunk, az elemet törölni kell ZS1 zsákból.) ZS2 listát a feldolgozás közben szabadítsuk fel. Az algoritmus műveletigénye  $O(n+m)$ , ahol  $n$  és  $m$  a listák elemszáma.

### 3. feladat (14p)

Adott egy fejelem nélküli, egyirányú ciklikus lista. Egy eleme az (érték, mut) párból áll. A lista első elemére a FEJ pointer mutat. Induljon el az első elemtől, és menjen el a  $k$ -edik elemig, az értékét írja ki, majd törölje az elemet a listából. Ismét menjen előre  $k$  elemet, írja ki, törölje. Ezt ismételve járjon a listán körbe-körbe, míg el nem fogynak az elemek (hasonlóan, mint a „kiszámolás játék”). A  $k$  és FEJ bemenő paraméter legyen, feltehető, hogy  $k \geq 1$ . A lista lehet üres is!

### 4. feladat (14p)

Egy rádióadónak az a feladata, hogy újra és újra leadja ugyanazt a jelsorozatot. A jelsorozat végén egy szünet-jelet ad, a példákban ezt # jellel jelöltük. Az adást egy szekvenciális fájlban rögzítettük, készítsen algoritmust, mely sor segítségével eldönti, hogy volt-e hiba az adás során. Az input fájlt csak egyszer olvashatjuk végig.

Példák:

a a 0 b # a a 0 b # a a 0 b	hibátlan
c 0 b # c 0 b # 0 b	hibás
a a a b c	hibátlan (nem volt ismétlés)
a b c d # a b c d # a b c d a	hibás

**5. feladat (8p)**

A tanult algoritmust alkalmazva határozzuk meg az alábbi kifejezés **lengyel formáját!** A lengyel formában minden operandus fölé rajzolja le a verem pillanatnyi tartalmát!

**Adja meg a kifejezésben szereplő operátorok precedenciáját is!**

(A mínusz előjelet is operátorként kell kezelni, precedenciáját úgy állapítsa meg, hogy a  $-2^3$  kifejezést  $(-2)^3$  –ként kell kiszámítani, az értékadó operátort is sorolja be az operátorok közé.)

$$x = (-\sin(x^2 + 5) * 3 - a) - (\cos(y - z * 3)^2 + b) / d + 7 * k$$