

## Algoritmusok és adatszerkezetek 1.

### Minta 1. Zh

#### 1. feladat:

Rendezze az előadásról ismert módon összefésülő rendezéssel (mergeSort(A)) az alábbi tömböt! (6 pont)

$$A = \langle 2, 8, 3, 5, 7, 4, 6, 10, 5', 1 \rangle$$

(Az aposztróf a kulcs mellett a kulcs második előfordulását jelöli, az értékét nem módosítja.)

Sorszámozza be a merge(A,B,C) segéd eljárás hívásai által kiszámolt rendezett résztömböket az eljárás hívások sorrendjének megfelelően!

#### 2. feladat:

a) Készítse el a következő kifejezés lengyelformáját úgy, hogy mindig megadja a verem tartalmát, amikor a veremből kivétel történik! Azonos prioritás esetén alkalmazzunk balról jobbra szabályt! (5 pont)

$$(x*y+10*a)*((b+c^d)*k-28)/(15*f+20*g)$$

b) Határozza meg a kifejezés értékét az alábbi lengyelformájából úgy, hogy minden műveleti jel előtt és után megadja a verem tartalmát! A műveletek kiszámításánál használja az alább megadott értékeket! (5 pont)

$$a,b,*,16,+,x,y,^,2,^,u,v,3,-,*,+,f,g,+,h,3,-,4,*,-,+,-$$

ahol

$$a = 4, b = 8, x = 3, y = 1, u = 4, v = 5, f = 12, g = 2, h = -1.$$

#### 3. feladat:

Egy tanár egy zh eredményét egy **sorban** tárolja. Egy elem tartalma egy (név, pontszám) rekord. A sorban név szerint rendezett sorrendben vannak az adatok.

Adott a minimális pontszám. Válogassuk ki egy másik sorba azokat, akik nem érték el a minimumot, az eredeti sorban maradjanak azok, akik megfeleltek.

Az algoritmust a sor műveletei segítségével adja meg! Csak az említett két sor használható! (10 pont)

#### 4. feladat:

Egy fejelemes **kétirányú ciklikus** listában (C2L) pontszám szerint növekvő sorrendben vannak tárolva a felvételizők eredményei (egy elem adat tartalma: név, pontszám). Egyik felvételiző reklamációját elfogadják, és megemelik a pontszámát! Javítsuk az adott felvételiző pontszámát, és átláncolással tegyük rendezetté ismét a listát! (12 pont)

(A listákat kizárólag az előadásról ismert  $unlink(q:E2^*)$ ,  $precede(q,r:E2^*)$  és  $follow(p,q:E2^*)$  eljárások segítségével szabad módosítani.)

#### 5. feladat:

Egy maximum **prioritási sor** absztrakt típust **fejelemes rendezetlen** egyirányú listával (H1L) valósítunk meg.

Adja meg a prioritási sort megvalósító osztályt (*adattagjait és metódusait*), valamint a következő metódusok algoritmusát is: **konstruktor**, **üres-e()**:logikai, **sorból()**:T ! (12 pont)