

Név: Neptun kód:

Algoritmusok és adatszerkezetek II. vizsga, 2016. 05. 31.

1. Mutassa be és magyarázza el a $\langle 0,79; 0,13; 0,16; 0,64; 0,39; 0,20; 0,89; 0,53; 0,71; 0,42 \rangle$ listán az *Edényrendezés* algoritmusát $[0; 1)$ -beli kulcsokra!

Mekkora a (szokásos, beszűrő rendezéses változatának) minimális és maximális műveletigénye? Miért? Mekkora az átlagos műveletigénye? Hogyan tudnánk a maximális műveletigényt aszimptotikusan csökkenteni? (Az edények továbbra is láncolt listák.) Mit értünk *stabil rendezés* alatt? Hogyan tudnánk a *Edényrendezést* stabil rendezéssé alakítani? (20p)

2. A $T[0..m-1]$ hasító tábla rései kétirányú, nemciklikus, fejelem nélküli, rendezetlen láncolt listák pointerai. Adott a $k \bmod m$ hasító függvény. A kulcsütközéseket láncolással oldjuk fel. Mindegyik kulcs csak egyszer szerepelhet T -ben. **(2.a)** Írjuk meg a $search(T[], m, k)$ függvényt, ami visszaadja a T -beli, k kulcsú listaelem címét, vagy a \odot pointert, ha ilyen nincs! **(2.b)** Írjuk meg a $del(T[], m, p)$ függvényt, ami törli a T hasító táblából (és deallokálja is) a p pointer által mutatott listaelemet! **(2.c)** Mi a kitöltöttségi hányados? Milyen aszimptotikus becslést tudunk adni a fenti műveletek minimális, átlagos és maximális futási idejére? Miért? (20p)

3. Mit számol ki a *Quick Search* algoritmus? Mi az előnye, illetve hátránya a naiv mintaillesztő algoritmussal összehasonlítva?

Mutassa be a *Quick Search* algoritmus **(a)** előkészítő eljárásának működését az $\{A, B, C, D\}$ ábécé-vel az *ABACABA* mintán, és **(b)** e mintát illesztő eljárását az *ABABACABACABADABACABABA* szövegen! Mekkora az egyes eljárások aszimptotikus műveletigénye? (20p)

4. Mit számol ki a *Dijkstra* algoritmus? Mekkora a műveletigénye n csúcsú gráf esetén, ha a prioritásos sort rendezetlen tömbbel reprezentáljuk? Miért?

Szemléltesse a működését az alábbi irányítatlan gráfon az **a** csúcsból indítva, az előadásról ismert módon!¹. Rajzolja le a legrövidebb utak fáját is, ami az eredményből adódik!

$a - b, 2; c, 1; d, 4. \quad b - d, 0. \quad c - d, 2; e, 2. \quad d - e, 1. \quad e.$ (20p)

5. Mit számol ki a *Kruskal* algoritmus? Szemléltesse a működését az alábbi gráfon! (Elég az „él – feszítő erdő” párosok sorozatát megadni.)

Mondja ki a biztonságos élekről és a minimális feszítőfákról szóló tételt! Definíálja a tételben szereplő *vágás* és *könnyű él* fogalmakat! Hogyan következik a *Kruskal* algoritmus helyessége ebből a tételből?

$a - b, 0; d, 1. \quad b - c, 5; d, 2; e, 3. \quad c. \quad d - e, 4.$ (20p)

¹ $u - v_1, w_1; \dots v_k, w_k.$ azt jelenti, hogy a gráfban az u csúcs u -nál nagyobb indexű szomszédai $v_1, \dots v_k$, és a megfelelő irányítatlan élek súlyai sorban $w_1, \dots w_k$.