Algoritmusok és gráfok
EXTRA FELADATOK, 2019 ősz

Az extra feladatokat írásban lehet beadni az előadáson vagy le lehet adni a tanszéki adminisztrációban a nevemre (IB133) vagy el lehet küldeni pdf-ben a csima@cs.bme.hu címre, az előadás kezdetéig. Határidő előtt is be lehet adni és ha nem jó, de még van idő, akkor lehet újra próbálkozni. Minden jól megoldott extra feladat +1%-ot számít a félév végi jegyben (feltéve, hogy az aláírást sikerül megszerezni és a vizsga is sikeres).

1. (**határidő: szeptember 18.**) Adott egy $2k$ (azaz páros hosszúságú) hosszú tömb, mely csupa különböző egész számot tartalmaz (tegyük fel, hogy $2k \geq 2$, azaz a tömb nem üres). Adjon olyan algoritmust, amely $3k-2$ összehasonlítással megtalálja a $2k$ elem közül a legnagyobbat és legkisebbet!

2. (**ez nehéz, határidő: szeptember 25**.) Adott n chip, melyek képesek egymás tesztelésére a következő módon: ha összekapcsolunk két chipet, mindkét chip nyilatkozik a másikról, hogy hibásnak találta-e. Egy hibátlan chip korrektül felismeri, hogy a másik hibás -e, míg egy hibás chip akármilyen választ adhat. Tegyük fel, hogy a chipek több, mint a fele korrekt. Adjunk algoritmust, mely n-nél kevesebb fenti tesztet használva kikeres egy jó chipet.

3. (**határidő: szeptember 25.**) A különböző valós számokból álló $a_0, \ldots, a_{n-1}$ sorozatot szeretnénk úgy átrendezni, hogy az új sorrendben $a_{i_0} < a_{i_1} > a_{i_2} < a_{i_3} \ldots$ teljesüljön.
Adjon erre a feladatra $O(n)$ lépésszámú algoritmust.

4. **(határidő: október 2.)** Az órán láttunk egy egyszerű algoritmust, amivel $n-1$ összehasonlítással meg tudjuk találni a legkisebb értéket egy n méretű tömbben. Mutassa meg, hogy ennél semelyik algoritmus sem lehet gyorsabb, azaz ha valaki előáll egy olyan algoritmussal, ami képes összehasonlításokkal megtalálni bármely n méretű tömb legkisebb elemét, akkor ez az algoritmus biztosan használ legalább $n-1$ összehasonlítást.

   Segítség: A feladat attól nehéz, hogy úgy kell belátnunk ezt az állítást, hogy közben nem hivatkozhatunk az algoritmus felépítésére, hiszen erről nem tudunk semmit, az állításnak az összes elképzelhető, helyesen működő algoritmusra igaznak kell lennie. Használhatjuk viszont az alábbi megfigyelést: mielőtt elkezdjük futtatni az algoritmust (amikor még semmit nem tudunk a tömb elemeiről) bármelyik elem lehet a legkisebb. Akkor vagyunk készen, ha a kezdetben lehetséges n jelöltből a legkisebb elem szerepére már egy kivételével mindet kizártuk („Csak egy maradhat!").

5. (**határidő: október 9.**) Adott egy $n \times n$-es táblázat, melyben egész számok szerepelnek. Adjon $O(n^2 \log n)$ összehasonlítást használó algoritmust, amely eldönti, van-e két olyan sor, amelyek teljesen megegyeznek (oszlopról oszlopra). Egy összehasonlításban két darab egész számot tudunk összehasonlítani.
Segítség: az összefésüléses rendezés lépészáma $O(n \log n)$.

6. (**határidő: október 16.**) Watson doktor azzal áll Sherlock Holmes elé, hogy talált egy olyan összehasonlítás-alapú rendező algoritmust, ami akármelyik n méretű input tömböt helyesen rendezi úgy, hogy eközben a tömb mindegyik eleme legfeljebb 2019-szer vesz részt összehasonlításban. Hogyan tudja megmutatni Sherlock Holmes Watsonnak, hogy téved, az algoritmusa nem lehet jó?

7. (**határidő: október 16.**) Egy n csúcsú bináris fa (nem feltétlenül bináris keresőfa) csúcsaiban egész számokat (pozitív és negatív számok is lehetnek) tárolunk. Szeretnénk meghatározni azt a csúcsát a fának, aminek a részfájában az értékek összege a lehető legnagyobb. (Egy csúcs

részfájába a csúcs maga és az összes leszármazottja tartozik bele.) Adjon erre a feladatra O(n) lépésszámú eljárást.

8. (**határidő: október 30.**) Egy k szintű teljes bináris fa olyan bináris fa, ahol minden szint tele van. Órán láttuk, hogy egy ilyan fában $2^k - 1$ csúcs van. Tegyük fel, hogy egy k szintű teljes bináris keresőfában az $\{1, 2, 3, \ldots, 2^k\}$ számok közül tárolunk $2^k - 1$-et, vagyis egy szám hiányzik. Adjon $O(k)$ lépésszámú (tehát a szintek számával arányos lépésszámú) algoritmust ennek a számnak a megtalálására.
   Segítség: rajzoljon le néhány 3 szintű ilyen fát és próbáljon valami szabályszerűséget találni :).

9. (**határidő: november 6.**) Bizonyítsa be, hogy egy olyan buliban, ahol legalább két ember van, biztosan van két olyan ember, akinek ugyanannyi ismerőse van a buliban. Az ismeretségekről tegyük fel, hogy kölcsönösek.
   Segítség: Ezen a héten a gráfokról kezdünk tanulni.

10. (**határidő: november 13.**) Szomszédossági mátrixával adott egy irányított G gráf, melynek n csúcsa van. A gráf minden csúcsához hozzá van rendelve egy 1 és k közötti egész szám (címke), a címkék ismétlődhetnek. Találjunk (ha létezik) olyan irányított utat a gráfban, amelyben minden $1 \leq i \leq k$ címke pontosan egyszer fordul elő. Az algoritmus lépésszáma legyen $O(k!\, n^2)$.

11. (**határidő: november 20.**) Szomszédossági mátrixával adott a súlyozott élű irányítatlan G gráf. Tegyük fel, hogy az élek súlyai az 1,2,3 számok közül valók. Adjon $O(n^2)$ költségű algoritmust az $s \in V$ pontból az összes további $v \in V$ pontokba vivő legrövidebb utak hosszának a meghatározására. (Itt egy út hossza nem az éleinek a széma, hanem az úton található élek súlyainak összege.)

12. (**határidő: november 27.**) Egy mátrixával adott irányított G gráfban minden csúcs ki van színezve, piros, zöld vagy kék színre (ez az információ egy, a csúcsokkal indexelt C tömbben adott). Adott egy piros s és egy piros t csúcs és olyan $O(n^2)$ lépésszámú algoritmust kell adnia, ami meghatározza a legrövidebb olyan út hosszát (az út hossza az éleinek a száma) s-ből t-be, ami legfeljebb egy kék csúcsot tartalmaz, minden más csúcs az úton piros.

13. (**határidő: december 4.**) Egy gráfot meg lehet adni szomszédossági mátrix helyett úgy nevezett éllistával is, ami a következőt jelenti: egy, a csúcsokkal indexelt tömböt tárolunk, ahol minden cella az indexével jelölt csúcsból kivezető éleket tartalmazza. Ez Pythonban egy olyan listát jelent, melynek elemei maguk is listák, a megfelelő csúcsok szomszédainak listái. Az éllista mérete irányított gráf esetén $n + e$ (mert n mérteű a tömb és a kis listák összehossza az élek száma, azaz e), irányítatlan gráf esetén $n + 2e$ (mert itt minden él kétszer szerepel).
   Írja át a szélességi bejárás tanult pszeudokódját éllistás változatra és mutassa meg, hogy ha a gráf éllistával van megadva, akkor a szélességi bejárás lépésszáma $O(n + e)$.

14. (**határidő: december 11.)** A mátrixával adott G irányított gráf élei között van egy negatív súlyú él, a többi él súlya pozitív. A gráfban nincs negatív súlyú kör. Adjon $O(n^2)$ lépésszámú algoritmust az $s \in V(G)$ pontból az összes többi pontba vezető legrövidebb utak meghatározására.