Tóth Ágnes
www.cs.bme.hu/˜tothagi

1. gyakorlat
Tagadás, rekurzió, általános feladatok

1. Jelöljük $T(n)$-nel egy algoritmus legnagyobb lehetséges lépésszámát az n méretű inputokon. Tudjuk, hogy $T(1) = 2$ és $T(n) = T(n-1) + 3$, amennyiben $n \geq 2$. Adjuk meg $T(n)$-t zárt alakban!

2. Mi lehet $T(n)$, ha $T(1) = 2$ és $T(n) = 3 \cdot T(n-1) + 1$, ha $n \geq 2$?

3. Mi a tagadása az alábbi állításoknak? Igazak ezek az állítások?
   (a) Minden kedden van algel gyakorlat.
   (b) Minden olyan hallgató, aki jár algel gyakorlatra, átmegy a vizsgán.
   (c) Minden olyan 17 lábú zsiráf, aki jár algel gyakorlatra, az átmegy a vizsgán.

4. Tegyük fel, hogy van egy számítógépes programunk, ami egy k méretű feladaton a jelenlegi gépünkön 1 nap alatt fut le. Beszereztünk egy százszor gyorsabb számítógépet. Ugyanazon programmal mekkora feladatot lehet az új gépen egy nap alatt megoldani, ha a program lépésszáma n méretű feladat esetén
   (a) n-nel, (b) n^3-bel, (c) 2^n-nel arányos? (fs: 2/1.)

5. Hány összehasonlítással lehet megtalálni n elem közül a legkisebbet? (fs: 3/3.)

6. Egy f fokú létrán bizonyos fokok annyira rozogák, hogy ha rálépünk, leszakadnak. Szerencsére tudjuk hogy melyik fokok ilyenek, hova nem szabad lépnünk. Egy lépéssel legfeljebb 3 fokot tudunk lépni. Adjon algoritmust ami meghatározza, hogy a létra aljától fel tudunk-e jutni a létra legfelső fokára! (Feltehető, hogy a legfelső fokra rá szabad lépni.) Az algoritmus lépésszáma legyen $c \cdot f$, ahol c valami fix konstans.
   Hogyan kell módosítani az algoritmust, hogy azt is kiszámolja, hogy hányféleképpen lehet feljutni a legfelső fokra?

7. Jelölje egy algoritmus maximális lépésszámát az n hosszú bemeneteken $L(n)$. Azt tudjuk, hogy minden $n > 3$ egész számra $L(n) \leq L(n-1) + \frac{n}{2}$ teljesül, és hogy $L(3) = 3$. Milyen felső becslést adhatunk ez alapján $L(n)$-re? A holnapi előadás után erre is tudni kell válaszolni: Következik-e ebből, hogy az algoritmus lépésszáma $O(n^2)$?

8. 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. (fs: 1/10.)

9. Mi az alábbi állításoknak a tagadása? (Két állítás akkor tagadása egymásnak, ha a két állítás közül minden esetben pontosan az egyik igaz.) Próbáljuk úgy megfogalmazni a tagadásokat, hogy ne szerepeljen bennük tagadószó.
   (a) Az évfolyamon minden hallgató fiú.
   (b) A teremben van olyan fiú, aki magasabb, mint 170cm.
   (c) Van olyan hallgató, aki sokat tanul, de nem megy át a vizsgán.
   (d) Mindenki, aki átmegy a vizsgán, sokat tanult.

10. Egy $2\times n$-es sakktábla mezőin n piros és $n-1$ kék négyzetet helyezünk el. Ezeket olyan módon akarjuk átrendezni, hogy a felső sorban piros, az alsóban kék négyzetek legyenek, s a bal alsó sarok maradjon üres. Ehhez egy-egy lépés során az üres mezőre tolhatjuk valamelyik szomszédját. Bizonyítsuk be, hogy
    (a) van olyan algoritmus, ami ezt megoldja $c \cdot n^2$ lépéssel, ahol c vmi fix konstans (azaz azt kell belátni, hogy $O(n^2)$ lépés elégséges ehhez);
    (b) létezik olyan d konstans, hogy minden algoritmus, ami ezt megoldja, szerencsétlen inputon használ legalább $d \cdot n^2$ lépést (azaz itt azt kell belátni, hogy a feladat megoldásához $\Omega(n^2)$ lépés szükséges). (fs: 1/11.)

11. Tudjuk, hogy minden hömpörő surjancs. Mondjuk meg minden alábbi állításra, hogy biztosan igaz, lehetséges, vagy biztosan hamis!
    (a) Tudjuk valamiről, hogy nem hömpörő. Azt állítom, hogy ez surjancs.
    (b) Tudjuk valamiről, hogy hömpörő. Azt állítom, hogy ez hogy ez nem surjancs.
    (c) Tudjuk valamiről, hogy nem surjancs. Azt állítom, hogy ez hömpörő.
    (d) Tudjuk valamiről, hogy nem surjancs. Azt állítom, hogy ez nem hömpörő.
    (e) Tudjuk valamiről, hogy surjancs. Azt állítom, hogy ez nem hömpörő.

    (Ha nehéz a feladat, akkor legyen a hömpörő=kertitörpe és surjancs=szobor)

Tóth Ágnes
www.cs.bme.hu/~tothagi

2. gyakorlat
Ordó, omega, teta

1. Az alábbi függvényeket rendezze olyan sorozatba, hogy ha f_i után közvetlenül f_j következik a sorban, akkor $f_i(n) = O(f_j(n))$ teljesüljön! Indokolja is meg, miért jó a választott sorrend!

$$f_1(n) = 8n^{2.5}, \quad f_2(n) = 5\sqrt{n} + 1000n, \quad f_3(n) = 2^{\log^2 n}, \quad f_4(n) = 2007n^2 \log n.$$

2. Az $\mathcal{A}$ algoritmusról azt tudjuk, hogy n hosszú bemeneteken a lépésszáma $O(n^2)$. Lehetséges-e, hogy
(a) minden n hosszú bemeneten $O(n)$ lépést használ?
(b) van olyan x, hogy az x bemeneten az algoritmus lépésszáma $10|x|^2 \log |x| - 800$ (ahol $|x|$ az x bemenet hosszát jelöli)?

3. Az alábbi függvényeket rendezze olyan sorozatba, hogy ha f_i után közvetlenül f_j következik a sorban, akkor $f_i(n) = O(f_j(n))$ teljesüljön! (vz: 2007.05.22./4.)

$$f_1(n) = \frac{1}{100} n^2 \log n, \qquad f_2(n) = 10^{10}(\log n)^3 - 100 \log n \qquad f_3(n) = 8^{\log n}.$$

4. Bizonyítsuk be, hogy
(a) $\log_2 f(n) = \Theta(\log_{100} f(n))$ $(f(n) > 0)$. (b) $f(x) = a_k x^k + a_{k-1}x^{k-1} + \ldots + a_0$ $(a_k \neq 0)$ $\implies$ $f(n) = \Theta(n^k)$.
(c) $2^{n+1} = O(2^n)$, de $2^{2n} \neq O(2^n)$. (d) $\max(f(n), g(n)) = \Theta(f(n) + g(n))$ $(f(n), g(n) > 0)$. (fs: 2/2.)

5. Jelölje egy algoritmus maximális lépésszámát az n csúcsú gráfokon $L(n)$. Azt tudjuk, hogy ha n páros, akkor $L(n) = L(\frac{n}{2}) + 5$ teljesül, ha pedig $n > 1$ páratlan, akkor $L(n) = L(n-2) + 3$. Következik-e ebből, hogy az algoritmus lépésszáma $O(n^2)$? (vz: 2008.03.28./1.)

6. Legyen f és g két, a pozitív számokat a pozitív számokba képező függvény. Tudjuk, hogy $f(x) = O(h(x))$ és $g(x) = O(h(x))$. Igaz-e, hogy
(a) ha g szigorúan monoton növő és $h(x) = 3x$, akkor $f(g(x)) = O(h(x))$;
(b) $f(g(x)) = O(h(x))$ minden h függvényre?

7. Az $\mathcal{A}$ algoritmusról azt tudjuk, hogy összefüggő gráfokon $O(n+e)$ lépést tesz. Mutassa meg, hogy az is igaz, hogy összefüggő gráfokon az algoritmus lépésszáma $O(e)$.

---

8. Az alábbi függvényeket rendezze olyan sorozatba, hogy ha f_i után közvetlenül f_j következik a sorban, akkor $f_i(n) = O(f_j(n))$ teljesüljön! (vz: 2008.05.06./1.)

$$f_1(n) = \frac{12}{365} 8^n \qquad f_2(n) = 2^{n^2} \qquad f_3(n) = 2008n^{10}.$$

9. Az alábbi függvényeket rendezze olyan sorozatba, hogy ha f_i után közvetlenül f_j következik a sorban, akkor $f_i(n) = O(f_j(n))$ teljesüljön! (vz: 2007.06.19./4.)

$$f_1(n) = 2^{100n} - 2^{50n} \qquad f_2(n) = 2007n^3 \qquad f_3(n) = 3^{3n}.$$

10. Állapítsa meg, hogy az alábbi függvények esetén mely párokra teljesül, hogy $f_i(n) = O(f_j(n))$. Válaszát indokolja is!

$$f_1(n) = 11n^2, \quad f_2(n) = 8n^2 \log n, \quad f_3(n) = n^2 + 100000.$$

11. Igaz-e, hogy
(a) ha $f = O(g)$ és $g = O(h)$, akkor $f = O(h)$?
(b) ha $f = \Omega(g)$ és $g = \Omega(h)$, akkor $f = \Omega(h)$?

12. Jelölje egy algoritmus maximális lépésszámát az n hosszú bemeneteken $L(n)$. Azt tudjuk, hogy minden $n = 2k > 4$ páros számra $L(2k) \leq L(2k-2) + 1$ teljesül, és hogy $L(4) = 10$. Következik-e ebből, hogy az algoritmus lépésszáma $O(n)$? (vz: 2007.06.05./4.)

13. Egy $\mathcal{A}$ algoritmusról azt tudjuk, hogy az n hosszú bemeneteken a lépésszáma $O(n \log n)$. Lehetséges-e, hogy
(a) van olyan x bemenet, amin a lépésszáma $|x|^3$?
(b) minden x bemeneten legfeljebb $2007|x|$ lépést használ?
(Szokás szerint $|x|$ az x szó hosszát jelöli.) (vz: 2007.06.12./4.)

14. Ugyanarra a feladatra van két algoritmusunk A és B, a maximális lépésszámukat leíró függvények legyenek f_A és f_B. Tudjuk, hogy $f_A(n) = O(f_B(n) \cdot \log n)$. Következik-e ebből, hogy
    (a) A minden bemenetre gyorsabb, mint B?
    (b) A nagy bemenetekre gyorsabb, mint B? (vz: 2008.05.27./4.)

15. Jelölje egy algoritmus maximális lépésszámát az n hosszú bemeneteken $T(n)$. Azt tudjuk, hogy minden $n > 1$ egész számra $T(n) \leq T(\lceil \frac{n}{2} \rceil) + n$ teljesül, és hogy $T(1) = 1$ és azt is tudjuk, hogy a $T(n)$ függvény nem csökkenő. Bizonyítsa be, hogy az algoritmus lépésszáma $O(n \log n)$.

3. gyakorlat
Dinamikus programozás

1. Az $1, 2, \ldots, n$ számoknak adott két permutációja, $x_1, \ldots, x_n$ és $y_1, \ldots, y_n$. A két sorozat egy közös részsorozata egy $1 \leq i_1 < \cdots < i_k \leq n$, és egy $1 \leq j_1 < \ldots < j_k \leq n$ indexsorozattal adható meg, ahol $x_{i_m} = y_{j_m}$ teljesül minden $1 \leq m \leq k$ esetén. Adjon $O(n^2)$ lépésszámú algoritmust, ami az x és y sorozatokban meghatároz egy leghosszabb közös részsorozatot.

2. Egy $n \times n$ méretű táblázat minden eleme egy egész szám. A táblázat bal alsó sarkából akarunk eljutni a jobb felső sarkába úgy, hogy egy lépésben a táblázatban vagy felfelé vagy jobbra egyet lépünk. Azt szeretnénk, hogy a lépegetés során látott elemek növekvő sorrendben kövessék egymást. Egy ilyen út értéke a benne szereplő számok összege. Adjon $O(n^2)$ futási idejű algoritmust, ami meghatározza, hogy az adott táblázatban a szabályok szerinti utak értékei között mekkora a legnagyobb! (vz: 2008.03.28./3.)

3. Legyen $w = w_1 w_2 \cdots w_n$ egy n betűből álló szó. Hívjuk részszónak w egy tetszőleges $w_i w_{i+1} \cdots w_{i+k}$ darabját $(1 \leq i \leq n-1,\ 1 \leq k \leq n-i)$. Adjon algoritmust, ami $O(n)$ lépésben meghatározza az összes a-val kezdődő és b-re végződő részszó számát. (vz: 2007.05.29./8.)

4. Legyenek $a_1, a_2, \ldots, a_n$ tetszőleges egész számok és $m < n^2$ egész. Adjon algoritmust, amely a bináris alakjukkal megadott $a_1, a_2, \ldots, a_n$ és m számokról eldönti polinom időben, hogy az $a_1, a_2, \ldots, a_n$ számok közül kiválasztható-e néhány úgy, hogy az összegük m-mel osztva egyet adjon maradékul. (vz: 2004.06.17./6.)

5. Egy n szóból álló szöveget kell sorokra tördelni. A szöveg i-edik szava ℓ_i karakterből áll, egy sor s karakter hosszú. Ha egy sor a szöveg i-edik szavával kezdődik és a j-edik szóval végződik, akkor az elválasztó szóközöket is figyelembe véve $t = s - (\ell_i + \ell_{i+1} + \cdots + \ell_j + j - i)$ üres hely marad a sor végén. Egy ilyen sor hibája legyen t^2. A tördelés hibája a nem üres sorok hibáinak összege. Adjon $O(n^2)$ lépéses algoritmust egy legkisebb hibájú tördelés meghatározására! (A szavak sorrendje rögzített.) (vz: 2006.04.07./6.)

---

6. Egy n és egy m karakterből álló szövegben meg akarjuk találni a legnagyobb azonos darabot, azaz ha az egyik szöveg $a_1 a_2 \cdots a_n$ és a másik $b_1 b_2 \cdots b_m$, akkor olyan
$1 \leq i \leq n$ és $1 \leq j \leq m$ indexeket keresünk, hogy
$$a_{i+1} = b_{j+1}, a_{i+2} = b_{j+2}, \ldots, a_{i+t} = b_{j+t}$$
teljesüljön a lehető legnagyobb t számra. Adjon erre a feladatra $O(mn)$ lépést használó algoritmust. (vz: 2007.06.12./8.)

7. Legyenek $a_1, a_2, \ldots, a_n$ tetszőleges egész számok és legyen b is egész szám. Adjon algoritmust, amely a bináris alakjukkal megadott $a_1, a_2, \ldots, a_n$ és b számokhoz $O(nb)$ időben megadja, hogy a b szám hányféleképpen áll elő az $a_1, a_2, \ldots, a_n$ számok közül néhány összegeként.

8. Legyen adott egy $n \times n$ pixelből álló fekete-fehér kép. Szeretnénk a képen a bal felső saroktól a jobb alsó sarokig egy jobbra-lefele haladó határvonalat húzni úgy, hogy a vonaltól jobbra-felfele eső fekete, valamint a vonaltól balra-lefele eső fehér pixelek számának az összege a lehető legkisebb legyen. Oldjuk meg ezt a feladatot $O(n^2)$ időben! (fs: 7/5.)

9. Adott egy fa, melynek csúcsaihoz súlyok vannak rendelve. Adjon lineáris idejű algoritmust, ami meghatározza a fában található maximális összsúlyú független ponthalmaz súlyát!

4. gyakorlat
Szélességi bejárás, Bellman-Ford és Floyd algoritmus

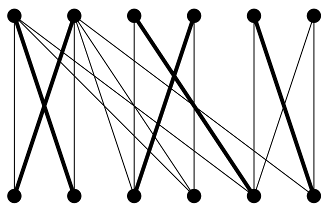
1. Adott a G irányítatlan gráf a következő éllistával: a:b,c; b:a,d; c:a,d; d:b,c,e,f; e:d,f,g; f:d,e,g,h; g:e,f,h; h:f,g; Keressünk G-ben a-ból kiinduló szélességi feszítőfát! Mennyi lesz a csúcsok a-tól való távolsága? (fs: 7/27d.)

2. Éllistával adott a súlyozott élű $G = (V, E)$ gráf. Tegyük fel, hogy az élek súlyai az 1,2,3 számok közül valók. Javasoljunk $O(n + e)$ költségű algoritmust az $s \in V$ pontból az összes további $v \in V$ pontokba vivő legrövidebb utak hosszának a meghatározására. (Itt n a G gráf csúcsainak, e pedig az éleinek a száma. (fs: 7/19.)

3. Egy $n \times n$-es sakktábla néhány mezőjén az ellenfél egy huszárja (lova) áll. Ha mi olyan mezőre lépünk, ahol az ellenfél le tud ütni, akkor le is üt, de egyébként az ellenfél nem lép. Valamelyik mezőn viszont a mi huszárunk áll. Adjunk $O(n^2)$ lépésszámú algoritmust, ami meghatározza, hogy mely másik mezőkre tudunk (lólépések sorozatával) eljutni a nélkül, hogy az ellenfél leütne! (vz: 2002.04.08./4.)

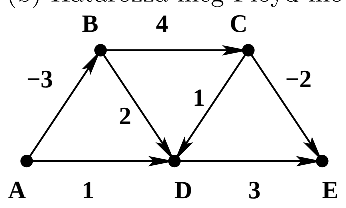
4. A $G(V, E)$ összefüggő, irányított gráf minden éle az $1, 2, \ldots, k$ számok valamelyikével van súlyozva. Egy út értéke legyen az úton található élek súlyainak *maximuma*. Határozza meg, hogy ha adott két csúcs $x, y \in V$, akkor mennyi a lehető legkisebb értékű x-ből y-ba vezető út értéke. Ha G éllistával adott és e éle van, akkor a lépésszám legyen $O(e \log k)$. (vz: 2003.03.31./6.)

5. Keressen javítóutat az alábbi páros gráfban!



6. Egy n pontú teljes gráf csúcsait kell kiszíneznünk csupa különböző színűre. Összesen $k \geq n$ féle szín áll rendelkezésre, de az egyes pontok színe nem teljesen tetszőleges. Minden v csúcshoz adott színeknek egy $S(v)$ listája, a v csúcsot csak az $S(v)$-ben szereplő színek valamelyikére színezhetjük. Adjon $O(nk^2)$ lépésszámú algoritmust, amely az $S(v)$ listák alapján eldönti, hogy van-e a megkötéseknek megfelelő színezés, és ha van ilyen, talál is egyet. (vz: 2005.04.08./6.)

7. (a) Határozza meg az A csúcsból az összes többi csúcsba vezető legrövidebb út hosszát a Bellman-Ford algoritmussal!
(b) Határozza meg Floyd módszerével az összes pontpárra a legrövidebb utak hosszát!



8. Nyári utazásunkra valutát akarunk váltani. A pénzváltó n különböző valutával foglalkozik, a j. fajta 1 egységéért r_{ij}-t kell fizetni az i. pénznemben. (Pl. ha a j. a dollár, az i. a forint, akkor most $r_{ij} = 222$ lehet.) Az r_{ij} tömb felhasználásával adjon $O(n^3)$ lépéses algoritmust, amely minden valutapárra meghatározza, hogy mi az elérhető legjobb átváltási arány, ha feltesszük, hogy az átváltásokért nem számolnak fel külön költséget. (Az i-ről a j-re való átváltás történhet több lépcsőben is, érdemes lehet előbb i-ről k_1-re konvertálni, onnan k_2-re, stb és végül j-re.) (vz: 2003.06.20./6.)

---

9. Éllistával adott egy G gráf, melynek n csúcsa és e éle van. A gráf minden csúcsához hozzá van rendelve egy 1 és k közötti egész szám (címke). Találjunk (ha létezik) olyan *tarka* utat a gráfban, amelyben minden $1 \leq i \leq k$ címke pontosan egyszer fordul elő. Az algoritmus lépésszáma legyen $O(k!\,(e + n))$. (vz: 2003.05.30./4.)

10. Egy számítógéphálózatban n számítógép van. Minden olyan eseményt, hogy az i-edik gép üzenetet küld a j-ediknek (i, j, t) formában feljegyezünk, ahol a t egész szám az üzenet küldésének időpontját jelöli. Ugyanabban a t időpontban egy gép több gépnek is küldhet üzenetet. Ha a t időpontban az i-edik gép vírusos volt, akkor egy (i, j, t) üzenet hatására a j-edik gép megfertőződhet, ami azt jelenti, hogy a $t + 1$ időponttól kezdve már a j-edik gép is vírusos lehet. Legyen adott az (i, j, t) hármasoknak egy m hosszú listája, valamint x, y és $t_0 < t_1$ egész számok. Azt kell eldöntenünk, hogy ha az x-edik gép a t_0 időpontban vírusos volt, akkor lehet-e emiatt az y-adik gép a t_1 időpontban vírusos. Adjon algoritmust, ami ezt a kérdést $O((t_1 - t_0)n + m)$ lépés után megválaszolja. (vz: 2007.06.12./5.)

11. Határozza meg a legrövidebb utak hosszát az A csúcsból az alábbi gráfban, a Bellman-Ford algoritmust futtatva. Lépésenként jelezze, hogyan változik az algoritmus által kitöltött T tömb.
A:B(3),F(1),E(12); B:C(2); C:D(4),G(2); D:E(1); E:C(-3); F:B(-1),G(4),; G:H(2); H:D(2),E(1).

12. Legyen $G = (V, E)$ mátrixszal adott n-pontú, súlyozott élű irányított gráf. Tegyük fel, hogy G nem tartalmaz negatív összhosszúságú irányított kört, továbbá azt, hogy a G-beli egyszerű irányított utak legfeljebb 25 élből állnak. Javasoljunk $O(n^2)$ költségű módszert az $1 \in V$ pontból az összes további $v \in V$ pontokba vivő legrövidebb utak hosszának a meghatározására. (fs: 7/6.)

13. Egy bajnokságban $2n$ csapat vesz részt. Minden fordulóban minden csapat pontosan egy mérkőzést játszik. Minden mérkőzést a két résztvevő csapat valamelyikének a pályáján játszanak. A következő k forduló mindegyikére már adott, hogy ki kivel fog játszani (a beosztás tetszőleges lehet, pl. ugyanaz a két csapat többször is játszhat egymás ellen). Az viszont még nincs meghatározva, hogy melyik mérkőzés kinek a pályáján történjen. Olyan pályabeosztást szeretnénk készíteni az adott mérkőzésekhez, hogy minden csapat felváltva játszon a saját pályáján és idegenben (azaz, amelyik csapat az első fordulóban otthon játszik, az legközelebb idegenben, utána megint otthon, stb). Adjon $O(kn)$ lépésszámú algoritmust, ami elkészít egy ilyen pályabeosztástvagy jelzi, hogy ez nem lehetséges. (vz: 2006.06.19./5.)

14. Kutyasétáltatáskor egy parkban egy gazda egy rögzített, egyenes szakaszokból álló útvonalon halad, aminek töréspontjai $t_1, \ldots, t_n$, a bejáratot jelölje t_0, a kijáratot t_{n+1}. A kutyája szabadon szaladgál, de a t_i pontokban találkozik a gazdájával. A t_i és t_{i+1} pontokban való találkozás között a kutya szeretne egy fát is meglátogatni (minden $i = 0, 1, \ldots, n$ esetén legfeljebb egyet-egyet). Legyenek adottak az $s(t_i, t_{i+1})$ távolságok ($0 \leq i \leq n$), valamint minden fának az összes t_i ponttól vett távolsága. Tegyük fel, hogy két találkozás között a kutya legfeljebb kétszer akkora távolságot tud megtenni, mint a gazda. Adjon algoritmust, ami segít a kutyának eldönteni, hogy mikor melyik fát látogassa meg ha a kutya célja, hogy minél több fánál járjon. Az algoritmus lépésszáma legyen $O(n^2 f + n f^2)$, ahol f a parkban levő fák számát jelöli. (vz: 2007.04.27./8.)

5. gyakorlat
Dijkstra algoritmus, kupac adatszerkezet

1. Egy irányított gráf csúcshalmaza $\{a, b, c, d, e, f\}$, az élek és súlyaik pedig az alábbiak: $s(a, b) = 5$, $s(a, e) = 6$, $s(b, c) = 4$, $s(b, d) = 6$, $s(c, a) = 3$, $s(c, d) = 1$, $s(d, e) = 2$, $s(e, c) = 2$, $s(e, f) = 1$, $s(f, b) = 3$, $s(f, c) = 1$, $s(f, d) = 1$.
   a) Dijkstra módszerével határozza meg a-ból az összes többi csúcsba vezető legrövidebb út hosszát. (Indokolni nem kell, de látszódjon, lépésenként hogyan változik a távolságokat tároló D tömb és a KÉSZ halmaz.)
   b) Egy él súlyát 1-gyel csökkentjük. Mely élek esetében nem változnak meg ezzel az a-tól mért távolságok? (vz: 2003.03.31./1.)

2. Adjuk meg az összes olyan minimális élszámú irányított gráfot (élsúlyokkal együtt), amely(ek)re az alábbi táblázat a Dijkstra–algoritmusban szereplő D[ ] tömb változásait mutathatja. Adja meg a legrövidebb utakat tartalmazó P[ ] tömb állapotait is. (fs: 7/2.)

v_1	v_2	v_3	v_4	v_5	v_6
0	2	6	∞	∞	7
0	2	5	9	∞	6
0	2	5	6	9	6
0	2	5	6	8	6
0	2	5	6	7	6

3. A $G = (V, E)$ irányított gráfban a csúcsok egy része fontos, ezeknek a csúcsoknak a halmaza az $\emptyset \neq F \subseteq V$. A gráf minden éléhez tartozik egy pozitív élsúly. Az $u \in F$ fontos csúcs távolsága a $v \in F$ fontos csúcstól a legrövidebb olyan u-ból v-be menő út hossza, aminek nincs u-tól és v-től különböző fontos csúcsa. Legyen a gráf a mátrixával adott, és minden csúcsra adott az is, hogy fontos csúcs-e. Adjon algoritmust ami $O(|V|^2|F|)$ lépésben meghatározza az összes fontos csúcspár közötti távolságot!) (vz: 2008.03.28./4.)

4. a) Építsen kupacot az órán tanult lineáris idejű módszerrel az alábbi tömbből: $31, 6, 50, 7, 2, 51$.
   b) Szúrja be az így kapott tömbbe az 1, majd ezután az 5 számot!
   c) Hajtson végre két egymást követő MINTÖR-t az így kapott kupacon!

5. Adjunk hatékony algoritmust egy kupac tizedik legkisebb elemének a megtalálására! Elemezzük a módszer költségét is! (fs: 3/37.)

6. Egy rendezett halmazból n elem kupacban van elhelyezve. Bizonyítsuk be, hogy a legnagyobb elem megkereséséhez $\Omega(n)$ összehasonlítás szükséges! (fs: 3/40.)

7. Tervezzünk olyan adatszerkezetet, ami egy rendezett halmaz elemeinek tárolására szolgál. A megvalósítandó műveletek:

   - **Felépít**(n) n elemből felépíti a struktúrát
   - **Mintör**, **Maxtör** a min. illetve max. elem törlése
   - **Beszúr**(x) az x elemet a struktúrába illeszti.

   Az egyes műveletek uniform költsége ne legyen több, mint **Felépít**: $O(n)$; **Mintör, Maxtör, Beszúr**: $O(\log n)$, ahol n a tárolt elemek száma. (fs: 3/32.)

---

8. Egy irányított gráf csúcshalmaza $\{a, b, c, d, e, f\}$, az élek és súlyaik pedig a következőek: $s(a, b) = 6$, $s(a, c) = 5$, $s(a, e) = 8$, $s(b, a) = 5$, $s(b, e) = 1$, $s(b, f) = 2$, $s(c, b) = 2$, $s(c, f) = 4$, $s(e, b) = 6$, $s(e, d) = 3$, $s(f, d) = 1$, $s(f, e) = 1$.
   a) Dijkstra-algoritmussal határozza meg a-ból az összes többi pontba vezető legrövidebb út hosszát. (Indokolni nem kell, de lépésenként írja fel a távolságokat tartalmazó D tömb és a KÉSZ halmaz állapotát.)
   b) Vegyük hozzá a gráfhoz az (b, d) élet. Milyen $s(b, d) \geq 0$ súlyok esetén változnának meg ezzel a legrövidebb utak hosszai? (vz: 2004.03.29./1.)

9. Egy vasúti menetrend segítségével meg akarjuk határozni, hogyan tudunk leggyorsabban eljutni A városból B-be. Azaz adott, hogy melyik városból hova megy közvetlenül vonat, mennyi idő alatt ér oda, ill. ha át akarunk szállni egyik vonatról a másikra, az adott helyen mennyi a várakozási idő. Ezek ismeretében adjunk hatékony algoritmust, amely meghatároz egy legkevesebb ideig tartó utat! (fs: 7/36.)

10. 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. (vz: 2004.05.27./6.

11. Adott egy kupac, mely n darab számot tartalmaz. Egy új kupacot szeretnénk építeni az eredeti kupac elemeinek (-1)-szereseiből. (Ehhez, ha akarjuk, használhatjuk az eredeti kupacot.) Mutassa meg, hogy az új kupac elkészítéséhez használt összehasonlítások száma $\Theta(n)$. (vz: 2006.04.07./4.)

12. Ha adott n szám, akkor hívjuk közülük középső elemnek a rendezés szerinti $\lceil n/2 \rceil$-ediket. Kezdetben adottak az $a_1, a_2, \ldots, a_n$ egész számok, amikről tudjuk, hogy az a_1 a középső elem, egyébként a számok rendezetlenek. Ezekből építsen fel egy adatszerkezetet, amiben két művelet van:
BESZÚR: egy új elemet illeszt az adatszerkezetbe,
KÖZÉPTÖR: az aktuális középső elemet törli.
Mindkét művelet megvalósítása $O(\log k)$ összehasonlítást használjon, amikor k tárolt elem van, az adatszerkezet kezdeti felépítése legyen $O(n)$ összehasonlítás! (vz: 2007.04.27./4.)

13. Egy orvosi rendelőben 2 orvos rendel, A és B. Bizonyos betegeket csak az egyik orvos láthat el (minden ilyen betegre adott, hogy ez az orvos A vagy B), más betegeket mindkettőjük elláthat. Emellett minden beteg kap egy prioritást, mely az eset súlyosságát jelzi. Adjunk olyan adatstruktúrát, amely a következő műveleteket támogatja:
BEHÍV(X): a legkisebb prioritású beteget adja vissza az X orvos által elláthatóak közül ($X \in \{A, B\}$).
TÖRÖL: töröl egy beteget az adatszerkezetből.
BESZÚR: egy új beteget szúr be az adatszerkezetbe.
A BEHÍV(X) művelet lépéssszáma legyen konstans, a másik kettőé pedig $O(\log k)$, ha k beteg van! (vz: ?)

14. A mátrixával adott irányítatlan $G(V, E)$ gráf egy város úthálózatát reprezentálja. A gráf bizonyos csúcsaiban parkoló van, minden parkolóhoz adott az ottani parkolás költsége. Egy parkoló a városrendezés szerint felesleges, ha a parkolótól legfeljebb k utcányira (azaz legfeljebb k éllel elérhetően) van nála olcsóbb parkoló. Adjon $O(k|V|^2)$ idejű algoritmust, ami eldönti, hogy van-e felesleges parkoló a városban. (2004.03.29./6.)

6. gyakorlat
Dijkstra algoritmus, kupac adatszerkezet, lineáris és bináris keresés, beszúrásos és összefésüléses rendezés

1. Vidéken autózunk, ahol benzinkút csak bizonyos falvakban van. Az A falubeli benzinkúttól indulunk és a B faluba akarunk elérni (ahol szintén van benzinkút). A falvak közötti utakat egy n csúcsú e élű, összefüggő, irányítatlan gráf írja le, melynek csúcsai a falvak, az élek pedig a falvak közötti utakat jelentik, egy él súlya a két falut összekötő útszakasz hossza. A gráf az éllistájával adott, és ezen kívül adott még az a k falu, amelyben van benzinkút. Adjon $O(ke\log n)$ lépésszámú algoritmust, amely meghatározza az A-ból B-be vivő legrövidebb olyan útvonalat, melynek során soha nem kell 600 kilométernél többet autóznunk két benzinkút között. (vz: 2006.04.07./5.)

2. Adott az $A[1:n]$ csupa különböző egész számot növekvő sorrendben tartalmazó tömb. (A tömbben negatív számok is lehetnek!) Adjunk hatékony algoritmust egy olyan i index meghatározására, melyre $A[i]=i$ (feltéve, hogy van ilyen i): igyekezzünk minél kevesebb elem megvizsgálásával megoldani a feladatot! (fs: 3/44.)

3. Rendezze az $11, 3, 27, 2, 5, 1, 4, 8$ tömböt (a) összefésüléses rendezéssel, (b) beszúrásos rendezéssel.

4. Legyen adott egy egészekből álló $A[1:n]$ tömb valamint egy b egész szám. Szeretnénk hatékonyan eldönteni, hogy van-e két olyan $i,j\in\{1,\dots,n\}$ index, melyekre $A[i]+A[j]=b$. Oldjuk meg ezt a feladatot $O(n\log n)$ időben! (fs: 3/21.)

5. Egy csupa különböző egészekből álló sorozat *bitonikus*, ha először nő, utána pedig fogy, vagy fordítva: először fogy, utána nő. Például az $(1,3,7,21,12,9,5)$, $(9,7,5,4,6,8)$ és $(1,2,3,4,5)$ sorozatok bitonikusak. Adjunk $O(n)$ összehasonlítást használó rendező algoritmust n elemű bitonikus sorozatok rendezésére! (fs: 3/5.)

6. A (növekvően) rendezett $A[1:n]$ tömb k darab elemét valaki megváltoztatta. A változtatások helyeit nem ismerjük. Javasoljunk $O(n+k\log k)$ uniform költségű algoritmust az így módosított tömb rendezésére! (fs: 3/10.)

7. Az n méretű (nem feltétlenül rendezett) A tömb elemei különböző pozitív egész számok. Adjon algoritmust, amely meghatároz egy $1\le k\le n$ számot és kiválaszt k különböző elemet az A tömbből úgy, hogy a kiválasztott elemek összege nem több mint k^3. Ha nincs ilyen k, akkor az algoritmus jelezze ezt a tényt. Az algoritmus lépésszáma legyen $O(n\log n)$. (Két szám összehasonlítása, összeadása vagy szorzása egy lépésnek számít.) (vz: 2004.06.10./4.)

---

8. A valós számokból álló $a_1,\dots,a_n$ sorozat olyan, hogy az $a_1^2,a_2^2,\dots,a_n^2$ sorozat egy darabig nő, utána csökken. Adjon $O(n)$ összehasonlítást használó algoritmust, ami rendezi az $a_1,\dots,a_n$ sorozatot. (vz: 2007.04.27./1.)

9. Adott a síkon n pont, melyek koordinátái (a_1,b_1), $(a_2,b_2),\dots,(a_n,b_n)$. Olyan $P=(x,y)$ pontot keresünk a síkon, amire az alábbi összeg minimális.
$$\sum_{i=1}^{n}(|a_i-x|+|b_i-y|)$$
Adjon algoritmust, ami $O(n\log n)$ lépésben meghatároz egy ilyen P pontot. (vz: 2007.06.19./5.)

10. Adott egy kupac és egy k kulcs. Adjunk algoritmust a kupac k-nál kisebb kulcsú elemeinek megkeresésére! Ha m ilyen elem van, akkor az algoritmus $O(m)$ lépést használhat. (fs: 3/39.)

11. Legyen adott egy rendezett univerzum $2n$ különböző eleméből álló S halmaz. Szeretnénk az S elemeit egy $A[1:2n]$ tömbbe elhelyezni úgy, hogy
$$A[1]<A[2]>A[3]<A[4]>\cdots<A[2n-2]>A[2n-1]<A[2n]$$
teljesüljön. Adjunk meg egy $O(n)$ összehasonlítást használó algoritmust erre a feladatra! (fs: 3/24.)

12. Egy n elemű sorozat csupa 0-ból és 1-esből áll. Rendezzük a sorozatot $n-1$ összehasonlítással! (fs: 3/26.)

13. Legyen adott egy csupa különböző egész számot tároló n elemű A tömb, és egy $1\le k\le n$ szám. A k darab legkisebb abszolút értékű tömbbeli elemet akarjuk meghatározni. Ha több megoldás is van, elég csak egy ilyen k-ast megadni. Adjon algoritmust, ami meghatároz k darab ilyen értéket és a lépésszáma $k\le\lfloor\log n\rfloor$ esetben $O(n)$.

14. Az $A[1 \dots n]$ tömbben egész számokat tárolunk, ugyanaz a szám többször is szerepelhet. Határozzuk meg $O(n \log n)$ lépésben az összes olyan számot, amelyik egynél többször fordul elő a tömbben. (vz: 2004.03.29./2.)

15. Az $A[1 : n]$ tömbben egész számokat tárolunk, ugyanaz a szám többször is szerepelhet. Határozzuk meg $O(n \log n)$ lépésben a leggyakoribb számokat, vagyis azokat, amelyeknél többször semelyik másik szám sem fordul elő a tömbben.

16. Adott egy $n \times n$-es mátrix. Adj $O(n^2 \log n)$ összehasonlítást használó algoritmust, amely eldönti, van-e két olyan sor, amelyeknek az első oszlopbeli elemei különböznek, viszont az összes többi oszlopban megegyeznek!(vz: 2002.06.25./5.)

7. gyakorlat
Buborék- és kupacos rendezés; gyorsrendezés; láda- és radixrendezés

1. Rendezze az $7, 3, 12, 1, 5, 4$ tömböt (a) buborékrendezéssel és (b) kupacos rendezéssel.

2. Adott egy egész számokat tartalmazó $A[1..n]$ tömb, amelyben legfeljebb n elempár áll inverzióban egymással (két elem akkor áll inverzióban, ha a nagyobb megelőzi a kisebbet). Igaz-e, hogy a buborék-rendezés rendezi az A tömböt
a) legfeljebb n összehasonlítással?
b) legfeljebb n cserével? (fs: 3/18.)

3. Adott egy dobozban n különböző méretű anyacsavar, valamint egy másik dobozban a hozzájuk illő apacsavarok. Kizárólag a következő összehasonlítási lehetőségünk van: Egy apacsavarhoz hozzápróbálunk egy anyacsavart. A próbának háromféle kimenete lehet: apa < anya, apa = anya, vagy apa > anya; annak megfelelően, hogy az apacsavar külső átmérője hogyan viszonyul az anyacsavar belső átmérőjéhez. Szeretnénk az anyacsavarokhoz megtalálni a megfelelő apacsavarokat. Adjunk erre a feladatra **átlagosan** $O(n \log n)$ összehasonlítást felhasználó módszert! (fs: 3/27.)

4. Rendezzük a következő láncokat a radix rendezés segítségével: *abc*, *acb*, *bca*, *bbc*, *acc*, *bac*, *baa*. (fs: 3/2.)

5. Vázoljunk egy $O(n)$ időigényű algoritmust (az időkorlát bizonyításával együtt) n olyan egész számból álló sorozat rendezésére, melynek elemei az
(a) $\{1, \ldots, 3n\}$ tartományba esnek!
(b) $\{1, \ldots, n^7 - 1\}$ tartományba esnek! (fs: 3/29.)

---

6. Adott n különböző elem, ezek közül keressük a kicsiket. A beszúrásos, az összefésüléses, illetve a kupacos rendezést a szokásos módon futtatva nagyságrendileg hány összehasonlítást végzünk, amíg megtudjuk, hogy melyik az első k darab legkisebb elem? (vz: 2006.04.07./2.)

7. A $G = (V, E)$ többszörös élet nem tartalmazó irányított gráf csúcsai legyenek $\{v_1, v_2, \ldots, v_n\}$. Tegyük fel, hogy a gráf olyan éllistával adott, amelyben minden csúcsnál a szomszédok tetszőleges sorrendben vannak felsorolva. Adjon algoritmust, ami $O(|V| + |E|)$ lépésben olyan éllistát hoz létre, amiben a szomszédok minden csúcsnál növekvő sorrendben vannak.

8. A 4 elemű I abc felett adott két szó: $x = x_1 x_2 \cdots x_n$ és $y = y_1 y_2 \cdots y_k$, ahol $1 \leq k \leq n$ és $x_i, y_j \in I$. Keressük az x szóban az olyan részszavakat, amelyek anagrammái y-nak, azaz az olyan i indexeket, hogy az $x_i, x_{i+1}, \ldots, x_{i+k-1}$ betűk megfelelő sorrendbe rakva az y szót adják. Adjon algoritmust, ami x-ben az összes ilyen i helyet $O(n)$ lépésben meghatározza. (vz: 2006.06.19./6.)

9. Minden nap több új megmunkálandó munkadarab érkezik a műhelybe, de naponta csak eggyel végeznek. Tegyük fel, hogy M napon át minden nap M újabb munkadarab érkezik. A munkadarabok meg vannak számozva 1-től M^2-ig, de tetszőleges sorrendben érkezhetnek. A műhelyben minden nap egy darabot, a felgyülemlett munkadarabok közül a legkisebb sorszámút csinálják meg. Jelölje A_i az i-edik nap érkező munkadarabok halmazát, $|A_i| = M$ és $A_1 \cup \ldots \cup A_M = \{1, \ldots, M^2\}$. Adjon algoritmust, amely az A_i halmazokból $O(M^2)$ lépésben meghatározza, hogy az M nap közül melyik nap melyik munkadarab fog elkészülni. (vz: 2005.06.23./6.)

8. gyakorlat
Keresőfák, pre-,in-,postorder bejárás; Piros-fekete fák

1. (a) Építsen beszúrásokkal bináris keresőfát az alábbi sorrendben érkező számokból: 7,3,2,9,8,12,6,4.
(b) Milyen sorrendben írja ki a preorder, inorder és posztorder bejárás a csúcsokat?
(c) Szúrja be az (a) résznél adott fába az 5-t, aztán törölje ki a 2,6 és 7 elemeket.

2. Egy bináris keresőfában csupa különböző egész számot tárolunk. Lehetséges-e, hogy egy KERES(x) hívás során a keresési út mentén a 20, 18, 3, 15, 5, 8, 9 kulcsokat látjuk ebben a sorrendben? Ha nem lehetséges, indokolja meg miért nem, ha pedig lehetséges, határozza meg az összes olyan x egész számot, amire ez megtörténhet. (vz: 2004.03.29./3.)

3. Egy bináris fa inorder bejárása:
$$j, b, k, g, i, a, c, d, f, e, h$$
preorder bejárása:
$$a, b, j, g, k, i, d, c, e, f, h.$$
Rekonstruáld a fát! (vz: 2002.06.11./3.)

4. Építsen piros-fekete fát az alábbi sorrendben érkező számokból: 1,2,3,4,5,6.

5. Egy piros-fekete fában valamelyik, a gyökértől egy levélig vezető úton sorban az alábbi színű pontok vannak: fekete, piros, fekete, fekete. Mennyi a fában tárolt elemek számának a minimuma? (2008.06.17./4.)

6. Egy piros-fekete fában lehetséges-e, hogy a piros-fekete tulajdonság megsértése nélkül
(a) néhány piros csúcsot átváltoztathatunk feketére?
(b) valamelyik, de csak egy piros csúcsot átváltoztathatunk feketére?
(Mást nem változtatunk a fán.) (vz: 2007.04.27./3.)

7. Adott egy n csúcsú és egy k csúcsú piros-fekete fa. A két fában tárolt összes elemből $O(n+k)$ lépésben készítsen egy rendezett tömböt. (vz: 2007.06.05./5.)

---

8. Adott n pont a síkon, melyek páronként mindkét koordinátájukban különböznek. Bizonyítsuk be, hogy egy és csak egy bináris fa létezik, melynek pontjai az adott n pont, és az első koordináta szerint a keresőfa tulajdonsággal, a második szerint pedig a kupac tulajdonsággal rendelkezik. (Vigyázat: a kupac tulajdonságba nem értendő bele, hogy a fa teljes bináris fa legyen, mint amilyet a tanult ”kupacépítő” algoritmus létrehoz.) (fs: 4/15.)

9. Lehetséges-e, hogy egy piros-fekete fából a tárolt elemeket preorder bejárás szerinti sorrendben kiolvasva ezt kapjuk: 6, 1, 5, 3, 2, 4? (vz: 2008.05.09./6.)

10. Egy bináris keresőfa ”valamely bejárásán” mindig a $\{pre,\ in,\ post\}$-order valamelyikét értjük.
(a) Mely bejárásoknál lehetséges az, hogy a tárolt elemek legnagyobbika megelőzi a legkisebbet?
(b) Tegyük fel, hogy egy bináris keresőfában az $1, 2, \ldots, n$ számok vannak tárolva, továbbá hogy a fa valamely bejárásánal a számok az $n, n-1, \ldots, 1$ sorrendben következnek. Határozzuk meg, melyik lehetett ez a bejárás és milyen lehetett ez a bináris keresőfa! (fs: 4/3.)

11. Egy kezdetben üres piros-fekete fába az $1, 2, \ldots n$ számokat szúrjuk be (ilyen sorrendben), milyen lépésekben lehet az 1 színe piros?

12. Tervezzen adatstruktúrát a következő feltételekkel. Természetes számokat kell tárolni, egy szám többször is szerepelhet. A szükséges műveletek:
BESZÚR(i): i egy újabb példányát tároljuk
TÖRÖL(i): i egy példányát töröljük
MINDTÖRÖL(i): i összes példányát töröljük
DARAB(i): visszaadja, hogy hány példány van i-ből
ELEM(K): megmondja, a nagyság szerinti rendezésben a K-adik elem értékét.
Az adatstruktúra legyen olyan, hogy ha m-*féle* elemet tárolunk, akkor mindegyik művelet lépésigénye $O(\log m)$.
(Például ha a tárolt elemek 1,1,3,3,3,8, akkor DARAB(1)=2, ELEM(4)=3 és m =3.) (vz: 2003.03.31./4.)

13. Adott egy $n = 2^k - 1$ pontú teljes bináris keresőfa. A fában tárolt elemek egészek az $I = [1, 2^k]$ intervallumból és egy szám legfeljebb egyszer fodul elő a fában. Utóbbi feltétel szerint pontosan egy olyan $i \in I$ egész van, amely *nincs* a fában. Adjunk egy hatékony módszert i meghatározására. (fs: 4/6.)

9. gyakorlat
2-3 fák, B-fák

1. Illesszük be az alábbi 6 kulcsot egy kezdetben üres $(2,3)$-fába a megadott sorrendben: D, B, E, A, C, F. Rajzoljuk le az eredményül kapott fát! (fs: 4/7.)

2. Az $[1,178]$ intervallum összes egészei egy 2-3 fában helyezkednek el. Tudjuk, hogy a gyökérben két kulcs van, és az első kulcs a 17. Mi lehet a második? Miért? (fs: 4/9.)

3. Egy B_{20}-fának (huszadrendű B-fának) 10^9 levele van. Mekkora a fa szintjeinek minimális, illetve maximális száma? (fs: 4/10.)

4. Egy kezdetben üres 2-3-fába az $1, 2, \ldots, n$ számokat szúrtuk be ebben a sorrendben. Bizonyítsa be, hogy a keletkezett fában a harmadfokú csúcsok száma $O(\log n)$. (vz: 2004.03.29./5.)

5. Egy orvosi rendelőben a regisztrációnál kell bejelentkezni, ahol az ott dolgozók eldöntik, hogy a beteg az épp rendelő két orvos közül A-hoz vagy B-hez kell kerüljön, vagy bármelyikükhöz kerülhet. Ezen kívül, a beutaló ismeretében, a beteghez egy, a sürgősséget kifejező, számot is rendelnek. Amikor valamelyik orvos végzett egy beteggel, akkor azon betegek közül, akiket nem csak a másik orvos láthat el, behívja a legnagyobb sürgősségi számút. Tegyük fel, hogy a kiosztott sürgősségi számok egymástól különbözőek. Írjon le egy olyan adatszerkezetet, ami abban az esetben, ha n beteg várakozik, akkor a regisztráción az új beteg beillesztését, illetve az orvosoknak a következő beteg kiválasztását $O(\log n)$ lépésben lehetővé teszi. (vz: 2008.03.28./5.)

6. Írjon le egy olyan adatszerkezetet, amivel egész számok véges sok részhalmazát tárolhatjuk, ha minden tárolandó T_i halmaznak véges sok eleme van.
Három műveletet definiálunk, a BESZÚR lépésszáma legyen $O(|T_i|)$, a másik két műveleté pedig $O(|T_i| + |T_j|)$.

   BESZÚR(i,x): a T_i halmazhoz hozzáveszi az x egész számot
   METSZETMÉRET(i,j): megadja a két halmaz metszetének $|T_i \cap T_j|$ elemszámát
   UNIÓMÉRET(i,j): megadja a két halmaz uniójának $|T_i \cup T_j|$ elemszámát. (vz: 2007.05.29./.5)

---

7. Az S_1 és S_2 kulcshalmazokat kiegészített 2-3-fákban tároljuk. Ezek az eredeti 2-3-fától abban különböznek csak, hogy minden csúcsban fel van jegyezve az onnan induló részfa magassága (szintjeinek száma). Tegyük még fel, hogy az S_1-beli kulcsok mind kisebbek az S_2-belieknél. Javasoljunk hatékony algoritmust a két fa egyesítésére. A cél tehát egy olyan kiegészített 2-3-fa, amelyben a kulcsok $S_1 \cup S_2$ elemei. (fs: 4/12.)

8. Egy 2-3 fában egy rendezett halmaz 10 000 elemét szeretnénk tárolni. Milyen korlátok közé esik a fa magassága? (fs: 4/8.)

9. Egy 2-3 fába egymás után 1000 új elemet illesztettünk be. Mutassa meg, hogy ha ennek során egyszer sem kellett csúcsot szétvágni, akkor a beillesztések sorozata előtt már legalább 2000 elemet tároltunk a fában. (vz: 2003.03.31./2.)

10. Vázolja a 2-3 fának (és műveleteinek) egy olyan módosítását, amiben továbbra is van KERES, BESZÚR, TÖRÖL, MIN, MAX művelet, és ezeken kívül van még RANG és K-ADIK művelet is, ahol RANG(x) azt adja vissza, hogy a tárolt elemek között az x a rendezés szerint hányadik elem, a K-ADIK(i) pedig, hogy a rendezés szerint a tárolt elemek közül melyik az i-edik. A módosítás során a felsorolt szokásos műveletek lépésszámának nagyságrendeje ne változzon, és mindkét új művelet lépésszáma legyen $O(\log n)$, ahol n a tárolt elemek száma. (vz: 2008.05.09./5.)

11. Egy sportklub teniszezői kialakítottak egy erősorrendet. Ezt a következő szabályok szerint tartják karban:

    - új játékos a sorrend végére kerül;
    - a rangsor szerinti i-edik játékos kihívhatja az $i-1$-ediket; ha legyőzi, helyet cserélnek.

    Tervezzünk olyan hatékony adatszerkezetet, mely lehetővé teszi a rangsor számítógépes kezelését! A szükséges funkciók a következők:

    - **Beszúr**(név): az új jövevényt a rangsor végére teszi
    - **Kihív**(név): az $i-1$. helyen álló személy nevét adja, ha a "név"-vel azonosított személy az i. helyen áll és $i > 1$.
    - **Kicserél**(i): kicseréli a rangsor i. és $i-1$. helyén levő személyeket, ha $i > 1$.

    (fs: 4/31.)

Tóth Ágnes
www.cs.bme.hu/˜tothagi

10. gyakorlat
Hash-elés

1. A hash-függvény legyen $f(K) = K$, a táblaméret $M = 7$, és $1 \leq K \leq 20$. Helyezzük el a táblában a 3, 4, 7, 11, 14, 17, 20 kulcsokat ebben a sorrendben
   (a)lineáris
   (b)kvadratikus maradék
   próbálást használva az ütközések feloldására. (fs: 6/5.)

2. A $T[0 : M]$ táblában $2n$ elemet helyeztünk el az első $3n$ helyen ($3n < M$) egy ismeretlen hash-függvény segítségével. A táblában minden $3i$ indexű hely üresen maradt ($0 \leq i < n$). Legfeljebb hány ütközés lehetett, ha az ütközések feloldására
   a) lineáris próbálást
   b) kvadratikus maradék próbálást használtunk? (fs: 6/7.)

3. Előfordulhat-e nyitott címzéses hash-elés esetén, hogy az $n > 3$ méretű táblában csak 3 elem van, de a keresés lépésszáma n ? (vz: 2006.04.07./1.)

4. Egy m méretű hash-táblában már van néhány elem. Adjon $O(m)$ lépésszámú algoritmust, amely meghatározza, hogy egy újabb elem lineáris próbával történő beszúrásakor maximum hány ütközés történhet. (vz: 2005.04.08./2.)

5. Az 1 és 91 közötti összes 3-mal osztható egész számot valamilyen sorrendben egy M méretű hash-táblába raktuk a $h(x) = x(\mod M)$ hash-függvény segítségével, lineáris próbával. Ennek során hány ütközés fordulhatott elő, ha $M = 35$, illetve ha $M = 36$? (vz: 2008.06.03./4.)

6. A kezdetben üres M méretű hash-táblába sorban beraktuk a $k_1, k_2, \ldots, k_n$ kulcsokat a $h(x) \equiv x(\text{mod } M)$ hash-függvénnyel, lineáris próbával. Jelölje t_1 a keletkezett táblában az egymás melletti foglalt mezők maximális számát. Amikor ugyanezt a $k_1, k_2, \ldots, k_n$ sorozatot ugyanabban a sorrendben egy üres $2M$ méretű táblába rakjuk be a $h(x) \equiv x(\text{mod } 2M)$ hash-függvénnyel, lineáris próbával, akkor a kapott táblában legyen t_2 az egymás melletti foglalt mezők maximális száma.
   (a) Igazolja, hogy $t_2 \leq t_1$
   (b) Igaz-e, hogy $t_1 \leq 2t_2$? (vz: 2005.05.26./5.)

7. A $T[0 : M-1]$ táblában rekordokat tárolunk nyitott címzésű hashelt szervezéssel. Az ütközések feloldására lineáris próbálást alkalmazunk. Tehát ha a $h(K)$ sorszámú cella foglalt, akkor a K kulcsú rekordot a $h(K) - 1, h(K) - 2, \ldots$ sorszámú cellák közül az első üresbe tesszük. Tegyük fel, hogy a tábla használata során egy hibás törlés történt: egy cellából kitöröltünk egy rekordot a törlés-bit beállítása nélkül.
   (a) Igaz-e, hogy a hibás törlés helye mindig megtalálható?
   (b) Adjunk hatékony (lineáris időigényű) algoritmust a tábla megjavítására. (Módosítsuk úgy a táblát, hogy megszűnjenek a hibás törlés negatív következményei.) (fs: 6/8.)

---

8. Nyitott címzéssel hasheltünk egy 11 elemű táblába a $h(k) = k \pmod{11}$ hash-függvény és kvadratikus maradék próba segítségével. A következő kulcsok érkeztek (a megadott sorrendben): $6, 5, 7, 17, 16, 3, 2, 14$. Add meg a tábla végső állapotát! (vz: 2002.06.25./3.)

9. A $b_0 \ldots b_n$ alakú $n + 1$ hosszú bitsorozatokat akarjuk tárolni. Tudjuk, hogy a b_0 paritásbit, ami a sorozatban az egyesek számát párosra egészíti ki. Ha nyitott címzésű hash-elést használunk $h(x) \equiv x \pmod M$ hash-függvénnyel és lineáris próbával, akkor $M = 2^n$ vagy $M = 2^n + 1$ méretű hash-tábla esetén lesz kevesebb ütközés? (vz: 2003.06.06./4.)

10. Mutassuk meg, hogy (nyitott címzéses hashelés, lin. próbálkozás esetén) már két kulcshoz tartozó hash-fügvényérték megegyezése is okozhat "tetszőlegesen" nagy méretű csomósodást. (fs: 6/3.)

11. Egy n méretű hash táblába lineáris próbával szúrjuk be az $a_1, a_2, \ldots, a_k, a_{k+1}$ elemeket. Az első k elem ($2 \leq k < n$) beszúrása során összesen $\binom{k}{2}$ ütközés történt. Legrosszabb esetben hány ütközés lesz a_{k+1} beszúrásakor? (vz: 2002.04.08./1.)

12. A kezdetben üres $M = 9$ méretű hashtáblába a $h(x) = x \pmod 9$ hash-függvény segítségével az adott sorrendben rakja be a 4, 27, 18, 13, 9, 10, 30 elemeket
    (a) lineáris próbával;
    (b) kvadratikus próbával.
    Mindkét esetben minden lépés után írja le a kapott tömböt és jelölje, hogy az aktuáliselem hol okozott ütközést. (vz: 2007.04.27./5.)

Tóth Ágnes
www.cs.bme.hu/˜tothagi

11. gyakorlat
Mélységi bejárás, PERT módszer, DAG-ok

1. Adott a G irányítatlan gráf a következő éllistával: a:b,c; b:a,d; c:a,d; d:b,c,e,f; e:d,f,g; f:d,e,g,h; g:e,f,h; h:f,g; Keressünk G-ben a-ból kiinduló mélységi feszítőfát! (fs: 7/27c)

2. A 6 pontú G gráf csúcsait jelölje x, y, z, u, v, w. A gráf egy mélységi bejárásánál a mélységi, ill. a befejezési számok a következők: x: 1,6; y: 2,4; z: 6,5; u: 3,3; v: 4,1; w: 5,2. Adjuk meg a bejáráshoz tartozó mélységi feszítőfa éleit. Rekonstruálható-e G az előző számok ismeretében?

3. Éllistájukkal adottak az alábbi G_1 és G_2 irányított gráfok (zárójelben az élsúlyok).
G_1: a:b(3),c(8); b:d(-7); c:d(5); d:e(2); e:a(-10);
G_2: a:g(2),f(10); b:a(-2),g(1); c:-; d:-; e:c(5),d(6); f:e(7); g:f(1), e(8);
(a) Döntsük el mélységi bejárás segítségével, hogy ezek a gráfok DAG-ok-e!
(b) Amelyik gráf DAG, abban adjunk meg egy topologikus sorrendet, határozzuk meg az a jelű csúcsból a c-be vezető legrövidebb út hosszát és számítsuk ki a gráfban levő leghosszabb út hosszát is.

4. Cirkuszi akrobaták egymás vállára állva minél nagyobb tornyot szeretnének létrehozni (a toronyban minden szinten csak egy akrobata lesz). Esztétikai és gyakorlati szempontok miatt egy ember vállára csak olyan állhat, aki nála alacsonyabb és könnyebb is. A cirkuszban n akrobata van, adott mindegyikük magassága és súlya. Adjon algoritmust, amely $O(n^2)$ lépésben megadja a lehetséges legtöbb emberből álló torony összeállítását. (vz: 2005.04.08./5.)

5. Adjunk algoritmust, mely egy éllistával megadott irányítatlan gráfban vagy talál egy kört, vagy igazolja a gráf körmentességét $O(|V|)$ időben (függetlenül attól, hogy $|E|$ akár sokkal nagyobb is lehet, mint $|V|$)! (fs: 7/20.)

---

6. Legyen G egy irányítatlan összefüggő gráf. Igaz-e, hogy
(a) G minden f éléhez van G-nek olyan mélységi bejárása, amelyben f egy faél?
(b) G minden f éléhez van G-nek olyan szélességi bejárása, amelyben f egy faél?
(c) G minden F feszítőfájához van G-nek olyan mélységi bejárása, amelyben F minden éle faél?
(d) G minden F feszítőfájához van G-nek olyan szélességi bejárása, amelyben F minden éle faél? (2006.04.07./3.)

7. Egy számítógéphálózatban n számítógép van. Minden olyan eseményt, hogy az i-edik gép üzenetet küld a j-ediknek (i, j, t) formában feljegyezünk, ahol a t egész szám az üzenet küldésének időpontját jelöli. Ugyanabban a t időpontban egy gép több gépnek is küldhet üzenetet. Ha a t időpontban az i-edik gép vírusos volt, akkor egy (i, j, t) üzenet hatására a j-edik gép megfertőződhet, ami azt jelenti, hogy a $t+1$ időponttól kezdve már a j-edik gép is vírusos lehet. Legyen adott az (i, j, t) hármasoknak egy m hosszú listája, valamint x, y és $t_0 < t_1$ egész számok. Azt kell eldöntenünk, hogy ha az x-edik gép a t_0 időpontban vírusos volt, akkor lehet-e emiatt az y-adik gép a t_1 időpontban vírusos. Adjon algoritmust, ami ezt a kérdést $O((t_1 - t_0)n + m)$ lépés után megválaszolja. (2007.06.12./5.)

8. Bizonyítsuk be, hogy minden $G = (V, E)$ irányított gráf felbontható két DAG-ra; pontosabban az élhalmazának van olyan E_1, E_2 partíciója ($E = E_1 \cup E_2$ és $E_1 \cap E_2 = \emptyset$), hogy a $G_1 = (V, E_1)$ és a $G_2 = (V, E_2)$ gráfok DAG-ok! (fs: 7/7.)

9. Legyen adott egy $n \times n$ pixelből álló fekete-fehér kép. Szeretnénk a képen a bal felső saroktól a jobb alsó sarokig egy jobbra-lefele haladó határvonalat húzni úgy, hogy a vonaltól jobbra-felfele eső fekete, valamint a vonaltól balra-lefele eső fehér pixelek számának az összege a lehető legkisebb legyen. Oldjuk meg ezt a feladatot $O(n^2)$ időben! (fs: 7/5.)

10. Van n fájlunk, az i-edik fájl hosszát jelölje a h_i. Tegyük fel, hogy a h_i számok egészek. Mentéshez két egyformán L méretű lemez áll rendelkezésünkre (L pozitív egész szám). A cél, hogy minél nagyobb k számra az első k darab fájl mindegyikét mentsük ki a lemezekre. Fájlokat szétvágni nem szabad, minden fájl teljes egészében kerül az egyik vagy a másik lemezre. Adjon algoritmust, ami adott L és h_i számokhoz meghatározza, hogy melyik fájlt melyik lemezre tegyük ahhoz, hogy k a lehető legnagyobb legyen. Az algoritmus lépésszáma legyen $O(L^2)$. (2006.06.26./6.)

4/4. A $G(V, E)$ összefüggő, irányított gráf minden éle az $1, 2, \ldots, k$ számok valamelyikével van súlyozva. Egy út értéke legyen az úton található élek súlyainak *maximuma*. Határozza meg, hogy ha adott két csúcs $x, y \in V$, akkor mennyi a lehető legkisebb értékű x-ből y-ba vezető út értéke. Ha G éllistával adott és e éle van, akkor a lépésszám legyen $O(e \log k)$. (vz: 2003.03.31./6.)

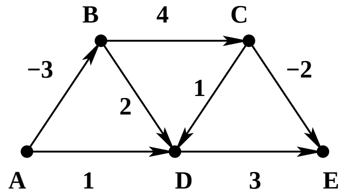
4/9. Éllistával adott egy G gráf, melynek n csúcsa és e éle van. A gráf minden csúcsához hozzá van rendelve egy 1 és k közötti egész szám (címke). Találjunk (ha létezik) olyan *tarka* utat a gráfban, amelyben minden $1 \leq i \leq k$ címke pontosan egyszer fordul elő. Az algoritmus lépésszáma legyen $O(k!\,(e + n))$. (vz: 2003.05.30./4.)

9/6. Írjon le egy olyan adatszerkezetet, amivel egész számok véges sok részhalmazát tárolhatjuk, ha minden tárolandó T_i halmaznak véges sok eleme van.
Három műveletet definiálunk, a BESZÚR lépésszáma legyen $O(|T_i|)$, a másik két műveleté pedig $O(|T_i| + |T_j|)$.

BESZÚR(i,x): a T_i halmazhoz hozzáveszi az x egész számot
METSZETMÉRET(i,j): megadja a két halmaz metszetének $|T_i \cap T_j|$ elemszámát
UNIÓMÉRET(i,j): megadja a két halmaz uniójának $|T_i \cup T_j|$ elemszámát. (vz: 2007.05.29./.5)

10/7. A $T[0 : M-1]$ táblában rekordokat tárolunk nyitott címzésű hashelt szervezéssel. Az ütközések feloldására lineáris próbálást alkalmazunk. Tehát ha a $h(K)$ sorszámú cella foglalt, akkor a K kulcsú rekordot a $h(K)-1, h(K)-2, \ldots$ sorszámú cellák közül az első üresbe tesszük. Tegyük fel, hogy a tábla használata során egy hibás törlés történt: egy cellából kitöröltünk egy rekordot a törlés-bit beállítása nélkül.
(a) Igaz-e, hogy a hibás törlés helye mindig megtalálható?
(b) Adjunk hatékony (lineáris időigényű) algoritmust a tábla megjavítására. (Módosítsuk úgy a táblát, hogy megszűnjenek a hibás törlés negatív következményei.) (fs: 6/8.)

4/7. (a) Határozza meg az A csúcsból az összes többi csúcsba vezető legrövidebb út hosszát a Bellman-Ford algoritmussal!
(b) Határozza meg Floyd módszerével az összes pontpárra a legrövidebb utak hosszát!



12. gyakorlat
Minimális súlyú feszítőfák

1. G irányítatlan gráf a következő éllistával (zárójelben a költségek, az élek mindkét végpontjukból fel vannak sorolva):
a:b(2),c(3); b:a(2),d(2); c:a(3),d(1); d:b(2),c(1),e(2),f(4); e:d(2),f(1),g(2); f:d(4),e(1),g(2),h(1); g:e(2),f(2),h(3); h:f(1),g(3);
Keressünk G-ben
(a) Prim algoritmusával minimális költségű feszítőfát!
(b) Kruskal algoritmusával minimális költségű feszítőfát! (fs: 7/27.ab)

2. Útépítéskor a környéken sok helyen felszedték a járdát. Az építők 1-től n-ig megszámozták a fontos pontokat (kapualj, útkereszteződés, stb.). A környék állapotát két $n \times n$ táblázat írja le. A J táblázatban $J[i,j] = 1$, ha az i és j pontok az utcán szomszédosak és megmaradt az ezeket összekötő részen a járda, egyébként az érték 0. A P táblázat az ideiglenesen elhelyezhető pallókat írja le: ha az i és j pontok összektöthetőek egy pallóval, akkor $P[i,j]$ ennek a pallónak a költsége. Amennyiben a két pont nem köthető össze egy pallóval, akkor a táblázatban $*$ szerepel. (Minden palló pontosan két pontot érint.) Szeretnénk biztosítani, hogy mindenhonnan mindenhova el tudjunk jutni (hol járdán hol pallón haladva). Az építők célja, hogy úgy válasszák meg a pallók helyét, hogy minél kevesebb pallót kelljen használniuk, és ezen belül a pallók értékeinek összege minimális legyen. Írjon le egy algoritmust, ami $O(n^2)$ lépésben javasol egy ilyen elhelyezést. (Egy pontra tetszőlegesen sok palló illeszkedhet, és a gyalogosok az egy pontra illeszkedő pallók bármelyikéről bármelyikére át tudnak lépni.) (vz: 2007.06.05./6.)

3. Éllistával adott a $G = (V, E)$ egyszerű, összefüggő gráf. A gráf élei súlyozottak, a súlyfüggvény $c : E \to \{-1, 1\}$. Adjon algoritmust, ami G-ben $O(|V| + |E|)$ lépésben meghatározza, hogy mennyi a minimális súlya egy olyan részgráfnak, ami G minden pontját tartalmazza és összefüggő. (vz: 2008.06.03./6.)

4. Mátrixával adott egy $G(V, E)$ irányított gráf, melynek minden éléhez egy pozitív súly tartozik. A gráf minden csúcsa vagy egy raktárat vagy egy boltot jelképez, az élsúlyok a megfelelő távolságokat jelentik. Olyan G' részgráfját keressük G-nek, amely minden csúcsot tartalmaz, és amelyben minden bolthoz van legalább egy raktár, ahonnan oda tudunk szállítani (azaz van köztük út a gráfban). Adjon $O(n^2)$ lépésszámú algoritmust egy a feltételeknek megfelelő minimális összsúlyú G' részgráf megkeresésére. (vz: 2003.06.06./5.)

5. Legyen adva egy (egyszerű, irányítatlan, öszefüggő) n pontú G gráf éllistával, az élek súlyozásával együtt. Tegyük fel, hogy a G-ből a v_1 csúcs, valamint a v_1-re illeszkedő élek elhagyásával keletkező G' gráf még mindig összefüggő, és adott G' egy minimális költségű feszítőfája. Adjunk minél hatékonyabb algoritmust a G gráf egy minimális költségű feszítőfájának az elkészítésére! (Teljes értékű megoldás: $O(n \log n)$ idejű algoritmus.) (fs: 7/28.)

6. A szoftverpiacon n féle grafikus formátum közötti oda-vissza konverzióra használatos programok kaphatók: az i-edik és a j-edik között oda-vissza fordító program ára a_{ij}, futási ideje pedig t_{ij} (ha létezik).
(a) Javasoljunk módszert annak megtervezésére, hogy minden egyes formátumról a saját grafikus terminálunk által megértett formátumra a lehető leggyorsabban konvertáljunk! (Az ár nem számít.)
(b) Javasoljunk módszert annak eldöntésére, hogy mely programokat vásároljuk meg, ha azt szeretnénk a lehető legolcsóbban megoldani, hogy a megvett programok segítségével bármelyik formátumról bármelyik más formátumra képesek legyünk konvertálni. (Itt a futási idő nem számít). (fs: 7/32.)

---

7. Legyen $G = (V, E)$ egy súlyozott irányítatlan gráf, amiben minden él súlya pozitív. Tegyük fel, hogy G összefüggő, de nem teljes gráf. A G gráfhoz egy 0 súlyú élt akarunk hozzáadni úgy, hogy a keletkező G' gráfban a minimális feszítőfa súlya a lehető legkisebb legyen. Adjon algoritmust ami a mátrixával adott G gráfra $O(|V|^3)$ lépésben meghatározza, hogy melyik két, a G-ben nem összekötött pont közé húzzuk be az új élet. (vz: 2006.06.12./5.)

8. Legyen G egy összefüggő gráf, az élein pozitív élsúlyokkal. A G gráfnak most csak az olyan feszítőfák érdekelnek minket, melyekben legfeljebb 3 darab nem elsőfokú pont van. Adjon polinom idejű algoritmust, amely meghatároz az ilyen tulajdonságú feszítőfák közül egy minimális súlyút.) (vz: 2005.06.09./5.)

9. Irányítatlan gráf tárolására adjon meg egy adatszerkezetet az alábbi műveletekkel:
ÚJCSÚCS(v): a gráfhoz hozzáad egy új csúcsot;
ÚJÉL(u, v): a már létező u és v csúcsok közé felvesz egy élet;
VANÚT(u, v): *igen* értéket ad vissza, ha vezet az u és v csúcsok között út, egyébként pedig *nem* értéket.
Ha a tárolt gráfnak n csúcsa van, akkor mindhárom művelet lépésszáma legyen $O(\log n)$. (vz: 2005.06.23./5.)

10. Adott (éllistával) egy G irányítatlan gráf, melynek bizonyos élei zöld színűek. Adjunk hatékony algoritmust olyan G-beli feszítőfa keresésére, melyben pontosan 2 zöld él szerepel! Elemezzük a módszer költségét! (fs: 7/21.)

11. Egy 20 szobás iroda számítógépeit hálózatba szeretnénk kötni. Az iroda szobái egy 2 méter széles folyosó két oldalán helyezkednek el; mindegyik szoba 3 méter széles (a folyosóval párhuzamos szélességről van szó). A folyosó egy lépcsőházból nyílik. Mindegyik szobában egy számítógép van, éspedig a folyosó felőli falnak a lépcsőház felőli sarkában. Oldjuk meg a lehető legrövidebb összhosszú vezetékkel, hogy bármely számítógépről bármely másik (esetleg közvetve) elérhető legyen a hálózaton. (Bármely két számítógép között vezethetünk egyenesvonalú vezetéket a padlóban. Nem szükséges, hogy egy összeköttetés a falakkal párhuzamos legyen.) (fs: 7/31.)

12. Éllistával adott egy összefüggő, egyszerű, irányítatlan G gráf csupa különböző élsúlyokkal. Jelöljük n-nel a csúcsok, e-vel pedig az élek számát. Mutassunk egy lineáris (azaz $O(e)$) uniform költségű algoritmust, ami a G gráf egy minimális feszítőfájának $[2/3n]$ élét előállítja! (Egy olyan $[2/3n]$ elemszámú élhalmazt keresünk, ami biztosan része egy minimális költségű feszítőfának.) (fs: 7/46.)

13. A $G(V,E)$ egyszerű összefüggő gráf minden f éléhez egy $s(f)$ súlyt rendeltünk. Legyen F_1 és F_2 a G gráf két különböző, minimális súlyú feszítőfája. Jelölje f_1 az F_1 fa egy tetszőleges élét. Bizonyítsa be, hogy van az F_2 fának olyan f_2 éle, hogy $s(f_1) = s(f_2)$. (vz: 2004.06.10./6.)

13. gyakorlat
P, NP, coNP, Karp-redukció

1. Bizonyítsa be, hogy az alábbi P_1, P_2, P_3 és P_5 eldöntési problémák NP-beliek, a P_4 pedig coNP-beli. Melyekről tudja belátni, hogy P-ben vannak?
P_1 : adott G páros gráf és k pozitív egész esetén van-e G-ben k élből álló párosítás?
P_2 : adott G irányítatlan gráfban van-e Euler kör?
P_3 : adott G irányítatlan gráf és k pozitív egész esetén van-e G-ben k darab független pont?
P_4 : adott m egész szám prím-e?
P_5 : adott $(s_1, \ldots, s_n)$ pozitív egészek és adott b egész pozitív szám esetén ki lehet-e választani néhány s_i-t, melyek összege b?

2. Adjon Karp-redukciót a 3-SZÍN eldöntési problémáról a 4-SZÍN eldöntési problémára!

3. Bizonyítsa be, hogy P-beli a következő eldöntési probléma: egy adott 4 színnel színezhető G gráf csúcsai kiszínezhetőek-e a piros, kék, zöld, sárga színekkel úgy, hogy pontosan egy csúcs legyen piros és pontosan két csúcs kék.

4. A G irányítatlan gráf minden x pontjához tartozik egy $s(x)$ súly. Célunk, hogy olyan feszítőfát találjunk a gráfban, amiben a levelekhez tartozó súlyok összege minimális. Fogalmazza meg a feladathoz tartozó eldöntési problémát, majd adjon Karp-redukciót a H-út feladatról erre a problémára.

5. Tegyük fel, hogy van egy olyan X eljárásunk, ami egy input G gráfra és k számra 1 lépés alatt megmondja, hogy van-e G-ben legalább k méretű független ponthalmaz.
(a) Tervezz olyan, a X eljárást használó algoritmust, ami polinom időben kiszámolja $\alpha(G)$-t, a független pontok maximális számát!
(b) Tervezz olyan, a X eljárást használó algoritmust, amely polinom időben talál egy $\alpha(G)$ méretű független ponthalmazt!

6. Bizonyítsa be az alábbi két problémáról, hogy NP-beliek. Melyikről tudja belátni, hogy P-ben van? Melyikről látja, hogy coNP-beli?
P_1 : adott G irányítatlan gráfban van-e legfeljebb 100 élből álló kör?
P_2 : adott G irányítatlan gráf és k pozitív egész esetén van-e G-ben legfeljebb k élből álló kör?

7. Tegyük fel, hogy van egy X programunk, amely egy n csúcsú G gráfról egy időegység alatt megmondja, hogy az kiszínezhető-e 3 színnel. Tervezz olyan X-t használó algoritmust, amely polinom időben megtalálja G egy 3 színnel való színezését (ha van ilyen egyáltalán)!

8. A $G = (V, E)$ egyszerű, irányítatlan gráfban legyen $X \subseteq V$ és $\overline{X} = V - X$ az X halmaz komplementere. Jelölje $m(X)$ az olyan élek számát, melyek X és $\overline{X}$ között futnak. Legyen *maxvágás* az az eldöntési feladat, hogy adott G gráf és k egész szám esetén létezik-e olyan X részhalmaza a csúcsoknak, hogy $m(X) \geq k$ és legyen *maxfelezés* az az eldöntési feladat, hogy adott G gráf és k egész szám esetén létezik-e olyan X részhalmaza a csúcsokn, hogy $m(X) \geq k$ és $|X| = |\overline{X}|$.
Igazolja, hogy *maxvágás* $\prec$ *maxfelezés.*

Tóth Ágnes
www.cs.bme.hu/˜tothagi

14. gyakorlat
Karp-redukció, NP-teljesség

1. Bizonyítsa be, hogy NP-teljes az alábbi eldöntési probléma: az adott $a_1, \ldots, a_n$ egész számok három részre oszthatóak-e úgy, hogy mindhárom rész összege ugyanannyi legyen?

2. Mutassa meg, hogy az alábbi eldöntési probléma P-ben van, vagy azt, hogy NP-teljes: adott egy $G(V, E)$ egyszerű gráf, melyre igaz, hogy $|E| \leq 2|V|$, kérdés, hogy a G gráf színezhető-e 3 színnel. (vz: 2005.06.09./4.)

3. Tudjuk, hogy P-beli annak eldöntése, hogy egy gráf síkgráf-e. Legyen a SÍK-MAXKLIKK eldöntési probléma a következő: adott egy G gráf és egy k egész, kérdés, hogy G olyan síkgráf-e, amiben van k pontú klikk. Mutassa meg, hogy ez a probléma NP-teljes, vagy mutassa meg, hogy P-ben van.

4. Jelölje P_1 azt az eldöntési problémát, hogy egy irányítatlan gráf összefüggő-e, P_2 pedig azt, hogy egy irányítatlan gráfban van-e Hamilton-kör. Lehetséges-e, hogy $P_1 \prec P_2$, illetve hogy $P_2 \prec P_1$? Válaszát indokolja is meg!

5. Egy n emberből álló szervezetben b féle bizottság működik. Bizottsági ülések időpontját akarjuk kitűzni. Két különböző bizottság ülése akkor lehet azonos napon, ha nincs olyan ember, aki mindkét bizottságnak tagja. Legyen adott egy k pozitív egész szám és minden bizottsághoz a tagok névsora. Azt szeretnénk eldönteni, hogy a b bizottsági ülés kitűzhető-e összesen legfeljebb k különböző napra. Vagy adjon egy, a kívánt beosztást megtaláló polinomiális algoritmust vagy mutassa meg, hogy a feladathoz tartozó eldöntési probléma NP-teljes. (vz: 2006.06.26./5.)

6. Egy hivatal új épületbe fog költözni. Az épület minden emeletén ugyanakkora terület használható fel irodák kialakítására. Minden részleg megmondta, hogy összesen mekkora irodaterületre tart igényt. Azt akarjuk eldönteni, hogy meg lehet-e oldani a költözést úgy, hogy egyetlen részleg se legyen kettévágva, azaz egy részleg teljes egészében egy emeleten legyen (de egy emeletre kerülhet több részleg is). Igazolja, hogy a kapcsolódó eldöntési probléma P-ben van, vagy azt, hogy NP-teljes. (vz: 2007.06.19./7.)

7. Tekintsük a Hátizsák problémának azt a folytonos változatát, amikor a tárgyak tetszőlegesen darabolhatóak, egy s_i súlyú v_i értékű tárgynak vehetjük az r-edrészét ($0 \leq r \leq 1$ racionális szám), és akkor ennek a résznek rs_i a súlya, rv_i az értéke. Definiálja az ehhez tartozó FOLYTHÁTIZSÁK eldöntési problémát és vagy mutassa meg, hogy P-ben van vagy azt, hogy NP-teljes. (vz: 2008.06.03./8.)

8. Mutassa meg, hogy az alábbi eldöntési probléma NP-teljes, úgy, hogy visszavezeti rá a MAXFTLEN ismerten NP-teljes problémát: adott G gráf és $a, b > 0$ egészek esetén van-e a G gráfnak a $K_{a,b}$ teljes páros gráffal izomorf feszített részgráfja?

9. Tegyük fel, hogy $P \neq NP$. Az alábbi feltételek közül melyikből következik és melyikből nem következik hogy az X eldöntési probléma nem P-beli?
   (a) Egy NP-teljes Y problémára X Karp-redukálható.
   (b) Egy NP-teljes Y probléma Karp-redukálható X-re.
   (c) az X probléma NP-beli. (vz: 2008.06.10./7.)

10. Igazolja, hogy ha $coNP \neq NP$, akkor $MAXKLIKK \notin P$.