Adatstruktúrák és algoritmusok
2011. május 3., kedd

DAG, PERT-módszer, Bellman-Ford és Floyd algoritmusok

1. 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.

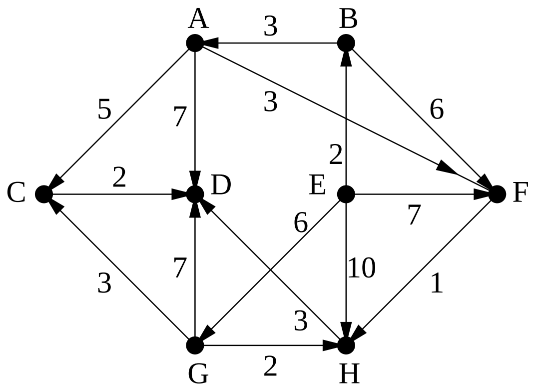
2. Bizonyítsuk be, hogy minden $G = (V, E)$ irányított gráf felbontható két DAG-ra; pontosabban az élhalmazának van olyan E_1, E_2 partíciója ($E = E_1 \cup E_2$ és $E_1 \cap E_2 = \emptyset$), hogy a $G_1 = (V, E_1)$ és a $G_2 = (V, E_2)$ gráfok DAG-ok!

3. 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. Adjunk algoritmust, amely $O(n^2)$ lépésben megadja a lehetséges legtöbb emberből álló torony összeállítását.

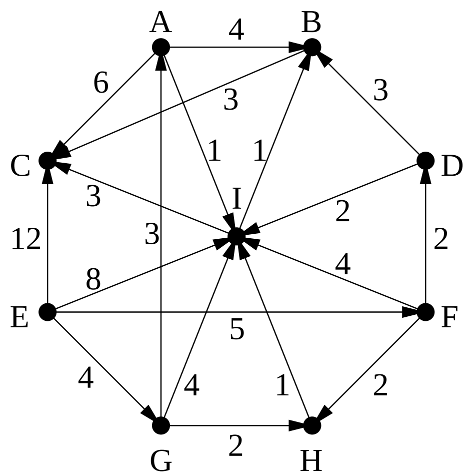
4. Adjunk algoritmust, mely egy éllistával megadott irányítatlan gráfban vagy talál egy kört,vagy igazolja a gráf körmentességét $O(|V|)$ időben (függetlenül attól, hogy $|E|$ akár sokkal nagyobb is lehet, mint $|V|$)!

5. Egy város úthálózata a $c_1, c_2, \ldots, c_n$ csomópontok (kereszteződések) között menő $u_1, u_2, \ldots, u_m$ egyirányú utcákból áll. Tudjuk azt is, hogy nem lehet semmilyen (az utcákat az irányításuknak megfelelően használó) útvonallal sem körbe menni. A városházánál egy kitüntetett csomópont van (c_1). Adjunk minél hatékonyabb módszert azon csomópontok meghatározására, amelyekből egyértelmű útvonalon lehet eljutni a városházához! Elemezzük a módszer költségét!

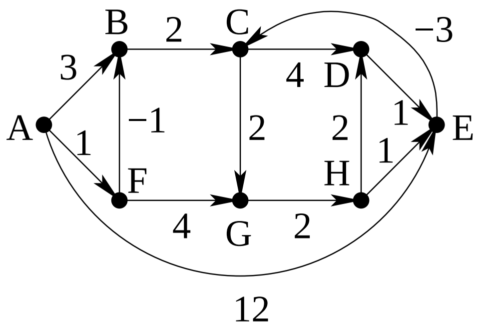
6. Határozzuk meg az alábbi PERT-gráf esetében a részfeladatok legkorábbi kezdési idejét, a teljes feladat elvégzéséhez szükséges minimális időt és a kritikus részfeladatokat (csúcsokat)!



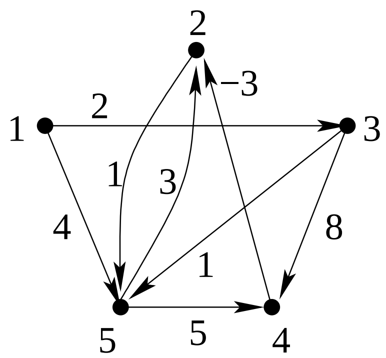
7. Határozzuk meg az alábbi PERT-gráf esetében a részfeladatok legkorábbi kezdési idejét, a teljes feladat elvégzéséhez szükséges minimális időt és a kritikus részfeladatokat (csúcsokat)!



8. Határozza meg az A csúcsból az összes többi csúcsba vezető legrövidebb út hosszát az alábbi gráfban a Bellman-Ford algoritmussal:



9. Az alábbi gráfon a Floyd-algoritmust futtatjuk. Az algoritmus során (a 4. javítási menet végén) az F_4 táblázat tartalmazza az ismert úthosszakat.



$$F_4 = \begin{pmatrix} 0 & 7 & 2 & 10 & 3 \\ \infty & 0 & \infty & \infty & 1 \\ \infty & 5 & 0 & 8 & 1 \\ \infty & -3 & \infty & 0 & -2 \\ \infty & 2 & \infty & 5 & 0 \end{pmatrix}$$

Hogyan változik a táblázat amikor minden csúcspárra újra elvégezzük a frissítést?

10. Egy ügyfél elektronikus archívumában n féle grafikus formátum egyikében szeretné tárolni a képeit. Rendelkezésre állnak bizonyos konverterprogramok. Az i-edik formátumról a j-edikre fordító program futási ideje t_{ij} (ha létezik). Javasoljunk módszert annak meghatározására, hogy melyik legyen a tárolási formátum, ha a megrendelő kívánsága az, hogy a különféle formátumokra történő konverziók átlagos futási ideje a lehető legrövidebb legyen!

11. 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.)

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.