

Topologikus rendezés, PERT-módszer

Adatstruktúrák és algoritmusok

9. gyakorlat

2014. április 8.

1. Éllistájukkal adottak az alábbi G_1 és G_2 irányított gráfok.

G_1 : $a : b, c$; $b : d$; $c : d$; $d : e$; $e : a$

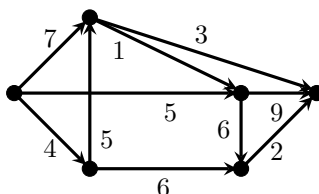
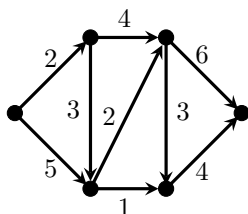
G_2 : $a : g, f$; $b : a, g$; $e : c, d$; $f : e$; $g : f, e$

(a) Döntsük el mélységi bejárás segítségével, hogy ezek a gráfok aciklikusak-e?

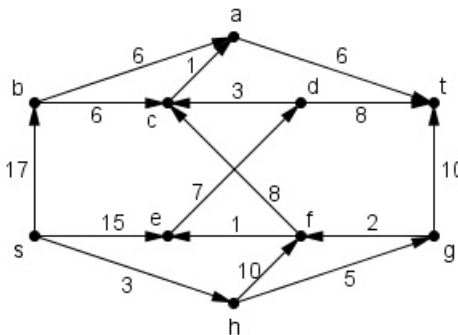
(b) Amelyik gráf DAG, abban adjunk meg egy topologikus sorrendet.

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. Határozzuk meg a következő gráfok által szemléltetett tevékenységhez szükséges időt!



4. Határozzuk meg az ábrán látható PERT probléma legrövidebb végrehajtási idejét, és állapítsuk meg, mik a kritikus tevékenységek.



5. 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.

6. Van b darab borítékunk, az i -ediknek a hossza h_i , a magassága m_i . Az i -edik borítékba akkor tudjuk berakni a j -edik borítékot, ha $h_j < h_i$ és $m_j < m_i$ is teljesül (nem forgatjuk és nem is hajtogatjuk a borítékokat). Célunk, hogy minél hosszabb olyan láncot alakítsunk ki, hogy az i -edikben benne van a j -edik, abban a k -adik, stb. Legyen adott egy $L > 0$ egész és a h_i és m_i számok. Hogyan lehet $O(b^2)$ lépésben eldönteni, hogy kialakítható-e a borítékokból egy L hosszú lánc?

7. Egy falutörténet írója n korábbi lakosról gyűjtött információkat. A kérdésekre kapott válaszok a következő típusúak voltak.

- S_i személy meghalt S_j születése előtt;
- S_i személy élete során született S_j ;
- S_i személy korábban született, mint S_j ;

- S_i korábban halt meg, mint S_j .

Egy S_i, S_j párra nem biztos, hogy szerepel minden választípus, és olyan pár is lehet, amely egyetlen válaszban sem szerepel együtt. Mivel az emberek időnként rosszul emlékeznek, nem biztos, hogy minden információ helyes. Adjon algoritmust, amivel k db fenti típusú válaszról $O(n + k)$ lépésben eldönthető, hogy van-e közöttük ellentmondás.