Algel III. gyakorlat

Utak, párosítások

2012. február 20.

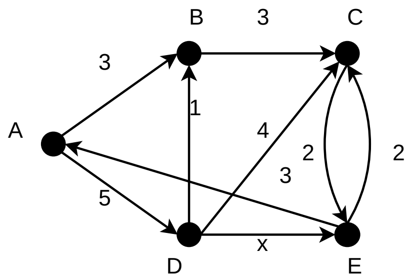
2. **Határozzuk meg a fenti jobb oldali gráfban Bellmann-Ford algoritmussal a legrövidebb s-ből induló utakat a többi csúcsba, nyomon követve az algoritmust!**
   Külön pdf-ben, nagyon részletesen.

5. **Hogy néz ki egy irányítatlan teljes gráf szélességi bejárása?**
   Egy csillag (1 db $n-1$ fokú pont körül $n-1$ db elsőfokú pont).

6. **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 meghatározására!**
   A 2 hosszú éleket helyettesítsük 2, a 3 hosszúakat 3 élellel. Az utak hossza így az élszám lesz (minden út annyi élből áll így, amilyen hosszú az eredeti gráfban lenne), így a szélességi bejárás használható legrövidebb út keresésére. A lépésszám $O(3n+3e) = O(n+e)$.

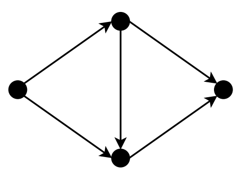
7. **Legfeljebb hány komponensből állhat egy irányított gráf szélességi bejárása során keletkező erdő?**
   Vegyünk egy n hosszú irányított utat! Ha a szélességi bejárást az utolsó pontból indítjuk, akkor elakad, utána az utolsó előttiből, ott is elakad stb., így pont n komponenst fog csinálni. Ennél többet nem is lehet n pontból.

8. **[ZH: 2009. április 24.] Dijsktra-algoritmussal határozza meg az alábbi gráfban az A pontból az összes többi pontba menő legrövidebb utak hosszát az x pozitív valós paraméter függvényében. Minden lépésnél írja fel a távolságokat tartalmazó D tömb állapotát és a KÉSZ halmaz elemeit.**

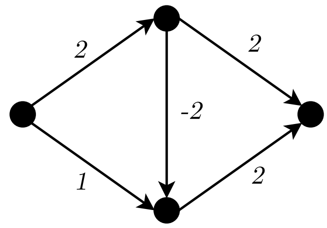


A	B	C	D	E	KÉSZ
$\underline{0}$	**3**	∞	5	∞	A
$\underline{0}$	$\underline{3}$	6	**5**	∞	A,B
$\underline{0}$	$\underline{3}$	6	$\underline{5}$	$5+x$	A,B,D
Ha $0 < x \leq 1$:					
$\underline{0}$	$\underline{3}$	**6**	$\underline{5}$	$\underline{5+x}$	A,B,D,E
$\underline{0}$	$\underline{3}$	$\underline{6}$	$\underline{5}$	$\underline{5+x}$	A,B,C,D,E
Ha $1 < x$:					
$\underline{0}$	$\underline{3}$	$\underline{6}$	$\underline{5}$	$\min(8, 5+x)$	A,B,C,D
$\underline{0}$	$\underline{3}$	$\underline{6}$	$\underline{5}$	$\underline{\min(8, 5+x)}$	A,B,C,D,E

9. **Határozzuk meg a jobb oldali gráfban az élsúlyokat úgy, hogy a Dijkstra algoritmus rossz eredményt adjon!**



Attól, hogy rakunk bele negatív élsúlyt, még akár működhetne! Úgy kell negatív élsúlyt belerakni, hogy romoljon el, pl:



A bal oldali csúcsból futtatva az alsó csúcsra az alogritmus 1-et mond, pedig a legrövidebb út oda 0 lenne.

10. **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. Adjuk meg a legrövidebb utakat tartalmazó $P[\,]$ tömb állapotait is!**

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

A táblázat alapján vizsgáljuk az algoritmus futását. Mivel a legkisebb élszámú gráf kell, ezért csak oda húzunk be élet, ahol az egyes lépésekben javítottunk (ide pedig kötelező). Látszik, hogy v_1-ből indulunk. Az első sor miatt innen v_2-be, v_3-ba és v_6-ba vezet él, a megfelelő súlyokkal. A KÉSZ-be v_2-t választjuk, és a javítások miatt rájövünk, hogy kell lennie egy v_2v_3, v_2v_4 és egy v_2v_6 élnek, 3, 7 és 4 súlyokkal. A továbbiakban minden ugyanígy (a gráfot nem rajzolom fel). Csak az utolsó előtti lépésben lesz választási lehetőség, hogy v_4-et vagy v_6-ot vegyük be a KÉSZ-be. Ettől függően lesz egy 2 súlyú v_4v_5 és egy 1 súlyú v_6v_5 élünk, vagy pont fordítva. A P tömb változásai (csak az első esetben):

v_1	v_2	v_3	v_4	v_5	v_6
–	v_1	v_1	v_1	v_1	v_1
–	v_1	v_2	v_2	v_1	v_2
–	v_1	v_2	v_3	v_3	v_2
–	v_1	v_2	v_3	v_4	v_2
–	v_1	v_2	v_3	v_6	v_2

11. **[ZH: 2009. április 24.] Egy kezdő autóvezető a városban való közlekedése során szeretne gyakorlatának megfelelő útvonalat választani. Az úthálózat egy irányítatlan gráfként van megadva, a csúcsok a kereszteződések, az élek az utak, a csúcsoknál adott, hogy nehéz-e számára a kereszteződés. (Az hogy nehéz, a kereszteződés tulajdonsága, nem azon múlik, merről érkezik oda és és merre akar rajta áthaladni.) Írjon le egy algoritmust, amivel meg lehet határozni, hogy az autós az egyik adott csúcsnál levő otthonából mely csúcsokba tud autóval úgy eljutni, hogy útja során két nehéz csúcs soha nem jön közvetlenül egymás után. Az algoritmus lépésszáma éllistás megadás esetén legyen $O(n + e)$, ahol n a csúcsok és e az élek száma.**
A gráfból kihagyjuk a nehezek közti éleket (4 pont). Szélességi bejárással megkeressük, hogy mely csúcsok vannak az otthonnal egy komponensben, ezek lesznek elérhetők (2 pont). Ez jó, mert pontosan azokat az éleket hagytuk el, amiken nem mehetünk át, minden más benne maradt (ezt lehet precízebben is) (2 pont). Lépésszám: éltörlések $O(e)$, bejárás $O(n + e)$, ami összesen $O(n + e)$ (2 pont).

12. **Egy G gráfban pontosan egy él súlya negatív, és nincs a gráfban negatív összsúlyú irányított kör. Adjunk $O(n^2)$ lépésszámú algoritmust az $s \in V(G)$ pontból az összes többi pontba vezető legrövidebb utak meghatározására!**
Egy legrövidebb út kétféle lehet: vagy használja a negatív élet, vagy nem. Először futtassunk

egy Dijsktrát (a szépség kedvéért hagyjuk ki a negatív élet), az i csúcsba vezető legrövidebb út hosszát jelölje d_i^+. Jelölje u azt a csúcsot, amiből a negatív él indul. Világos, hogy d_u^+ helyes, hiszen a negatív élen nem mehetünk át az ő eléréséhez az eredeti gráfban. Belőle is indíthatunk egy Dijsktrát (az eredeti gráfon), hiszen ekkor először a negatív él másik végpontját veszi be a KÉSZ halmazba, ami a definíciónak megfelel. Ezeket a legrövidebb értékeket jelölje d_i^-. A legrövidebb utak tehát $\min(d_i^+, d_u^+ + d_i^-)$ képlettel számolhatók. Lépésszám egy keresés és két Dijsktra, azaz $O(e+n^2+n^2) = O(n^2)$ (az ugye mindenkinek világos, hogy egy egyszerű gráfban $e = O(n^2)$).

13. **[ZH: 2007. április 27.] Kutyasétáltatáskor egy parkban egy gazda 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 ($0 \le i \le n$), 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 $O(n^2f + nf^2)$, ahol f a parkban levő fák számát jelöli.**
    Ez csak annyiban trükkös, hogy nem legrövidebb út, hanem párosítás. Ugyanis minden gazdaponthoz párosítani akarunk egy fát. Egyik pontosztály tehát a gazda pontjai, másik a fák, akkor megy él, ha adott pontból elérhető a fa úgy, hogy utána a kutya vissza tud érni. Itt kell max. párosítás, ami mehet magyar módszerrel, ami $O(|V||E|) = O((n + f)(nf)) = O(n^2f + nf^2)$, a felépítés pedig megy $O(nf)$-ben, ami nyilván belefér (a távolságszámítás egy pont és fa és pont között konstans idő).

14. **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!**
    A Bellmann-Ford algoritmus során a táblázat i-edik sorában a kezdőpontból a legfeljebb i élszámú legrövidebb utak hosszai szerepelnek (ezt nem kell külön bizonyítani!). Mivel tudjuk, hogy G-ben minden út legfeljebb 25 élből áll, a táblázatot elég a 25. sorig kitölteni. A lépésszám így $O(25n^2) = O(n^2)$ lesz.

15. **Adott éllistával egy n pontú, e élű G összefüggő irányítatlan gráf. Adjunk $O(e)$ költségű algoritmust olyan $X \subset V(G)$ központi ponthalmaz keresésére, melyre $|X| \le n/2$ teljesül! Az $X \subseteq V(G)$ egy *központi ponthalmaz*, ha G minden pontja vagy X-beli, vagy egyetlen éllel elérhető valamelyik X-beli pontból.**
    Az összefüggőség és irányítatlanság miatt a szélességi bejárás egy szélességi feszítőfát fog adni. A páratlan szinteken lévő pontok száma legyen x, értelemszerűen a párosoké ekkor $n - x$. Ha $x < n - x$, akkor válasszuk központi halmaznak a páratlan szinten lévő pontokat, egyébként a párosakat. Könnyen látszik, hogy minden pontra igaz, hogy vagy benne van a kiválasztott halmazban, vagy elérhető belőle egy lépésben (elég csak a szomszéd szintre menni). A kiválasztott halmaz a skatulya-elv miatt biztos legfeljebb $n/2$ méretű lesz. A lépésszám $O(n + e)$, ami az összefüggőség miatt $O(e)$.