Algoritmusok és gráfok
TIZENEGYEDIK GYAKORLAT, 2019. december 6.
Megoldások

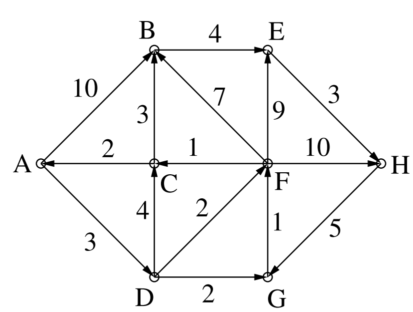
4. Az alábbi irányított G_2 gráf csúcsai a, b, c, d, e, f, g, h, élei pedig $ae(5), af(4), ag(1), ba(3), ch(1), da(1), de(-10), eg(1), fc(8), fg(-4), fh(3), gh(-12)$.
A d, b, a, f, c, e, g, h topologikus sorrenddel alkalmazza a DAG-ban használható tanult eljárást a legrövidebb b-ből induló utak meghatározására a *távolság* és *honnan* tömbök kitöltésével.

Megoldás

Nézzük végig csúcsról csúcsra, hogyan találjuk meg a *távolság* és a *honnan* tömb értékeit:

- Mivel d a kezdő b előtt van a toplogikus sorrendben, ezért rá *távolság[d]*$= \infty$, *honnan*$[d] = *$.
- Mivel b a kezdőcsúcs: *távolság[b]*$= 0$, *honnan*$[b] = b$.
- Az a csúcsba a d és a b csúcsból vezet él, ezért
távolság$[a] = \min_{u \to a} \{$*távolság*$[u] + c(u, a)\} = \min\{$*távolság*$[d] + c(d, a),$ *távolság*$[b] + c(b, a)\} =$
$= \min\{\infty + 1, 0 + 3 = 3\}$ és *honnan*$[a] = d$.
- Az f csúcsba csak a b-ből vezet él, ezért
távolság$[f] = \min_{u \to f} \{$*távolság*$[u]+c(u, f)\} =$ *távolság*$[b]+c(b, f) = 3+4 = 7$ és *honnan*$[f] = a$.
- A c csúcsba csak az f-ből vezet él, ezért
távolság$[c] = \min_{u \to c} \{$*távolság*$[u]+c(u, c)\} =$ *távolság*$[f]+c(f, c) = 7+8 = 15$ és *honnan*$[c] = f$.
- Az e két három irányból lehet elérni: az a vagy a d csúcsból, ezért:
távolság$[e] = \min_{u \to e} \{$*távolság*$[u] + c(u, e)\} = \min\{$*távolság*$[a] + c(a, e),$ *távolság*$[d] + c(d, e)\}$
$= \min\{3 + 5, \infty - 10\} = 8$
Az is kiderült, hogy a minimum az a csúcsnál volt, vagyis *honnan*$[e] = a$.
- A g csúcsot három irányból lehet elérni: az a, az e vagy az f csúcsból, ezért:
távolság$[g] = \min_{u \to g} \{$*távolság*$[u] + c(u, g)\} =$
$= \min\{$*távolság*$[a] + c(a, g),$ *távolság*$[e] + c(e, g),$ *távolság*$[f] + c(f, g)\} = \min\{3 + 1, 8 + 1, 7 - 4\} = 3$
Az is kiderült, hogy a minimum az f csúcsnál volt, vagyis *honnan*$[g] = f$.
- A h csúcsot is három irányból lehet elérni: a c, az f vagy a g csúcsból, ezért:
távolság$[h] = \min_{u \to h} \{$*távolság*$[u] + c(u, h)\} =$
$= \min\{$*távolság*$[c] + c(c, h),$ *távolság*$[f] + c(f, h),$ *távolság*$[g] + c(g, h)\} = \min\{15 + 1, 7 + 3, 3 - 12\} = -9$
Az is kiderült, hogy a minimum a g csúcsnál volt, vagyis *honnan*$[h] = g$.

5. Dijkstra-algoritmussal határozza meg a-ból az összes többi pontba vezető legrövidebb út hosszát és magukat a legrövidebb utakat is. (Indokolni nem kell, de látszódjon, lépésenként hogyan változik a *távolság*, a d és a *honnan* tömb és a KÉSZ halmaz.)



Megoldás

Az alábbi táblázatok mutatják a három tömb változásait (a táblázatok 1. sora adja meg, hogy az algoritmus elején hogyan néznek ki a tömbök, utána pedig minden újabb sor azt mutatja, hogy hogyan változnak a futás során az egyes körökben (amikor egy-egy új csúcs bekerül a KÉSZ-be) a tömbök.

távolság:

A	B	C	D	E	F	G	H
0	∞	∞	∞	∞	∞	∞	∞
0	∞	∞	3	∞	∞	∞	∞
0	∞	∞	3	∞	5	∞	∞
0	∞	∞	3	∞	5	5	∞
0	∞	6	3	∞	5	5	∞
0	9	6	3	∞	5	5	∞
0	9	6	3	13	5	5	∞
0	9	6	3	13	5	5	15

d:

A	B	C	D	E	F	G	H
*	10	∞	3	∞	∞	∞	∞
*	10	7	*	∞	5	5	∞
*	10	6	*	14	*	5	15
*	10	6	*	14	*	*	15
*	9	*	*	14	*	*	15
*	*	*	*	13	*	*	15
*	*	*	*	*	*	*	15
*	*	*	*	*	*	*	*

honnan:

A	B	C	D	E	F	G	H
A	A	*	A	*	*	*	*
A	A	D	A	*	D	D	*
A	A	F	A	F	D	D	F
A	A	F	A	F	D	D	F
A	C	F	A	F	D	D	F
A	C	F	A	B	D	D	F
A	C	F	A	B	D	D	F

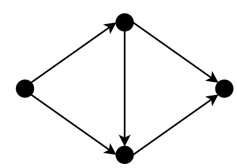
5. Adja meg azt a minimális élszámú irányított gráfot (élsúlyokkal együtt), amelyre az alábbi táblázat a Dijkstra–algoritmusban szereplő $d[\,]$ tömb változásait mutathatja. Adja meg a legrövidebb utakat tartalmazó *honnan*[ ] tömb állapotait is.

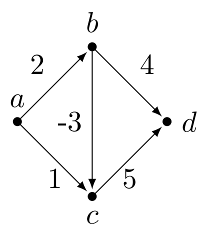
v_1	v_2	v_3	v_4	v_5	v_6
$*$	2	6	∞	∞	7
$*$	$*$	5	9	∞	6
$*$	$*$	$*$	6	9	6
$*$	$*$	$*$	$*$	8	6
$*$	$*$	$*$	$*$	7	$*$

Megoldás

Az első sorból látszik, hogy a kezdőcsúcs a v_1 és ebből van él (súlya 2) v_2-be, van él (súlya 6) v_3-ba és van él v_6-ba (súlya 7).
Ezután v_2 kerül KÉSZ-be, minden olyan érték ami változik a második sorban egy v_2-ből kiinduló él miatt változik: biztosan van egy 3-as él v_2-ből v_3-ba, 7-es él v_2-ből v_4-be és 4-es él v_2-ből v_6-ba.
Ezután v_3 kerül KÉSZ-be, minden olyan érték ami változik a harmadik sorban egy v_3-ból kiinduló él miatt változik: biztosan van egy 1-es él v_3-ból v_4-b3 és 4-es él v_3-ból v_5-be. Ahol nincs változás ott lehet, hogy volt él (csak nagy volt a súlya), de az is lehet, hogy nem volt él és mivel nekünk csak azokat az éleket kell megtalálnunk, amik biztosan voltak a gráfban, ezért ezekkel a helyekkel nem kell törődnünk.
Ezután v_4 kerül KÉSZ-be, minden olyan érték ami változik a negyedik sorban egy v_4-ből kiinduló él miatt változik: biztosan van egy 2-es él v_4-ből v_5-be.
Ezután v_6 kerül KÉSZ-be, minden olyan érték ami változik az utolsó sorban egy v_6-ból kiinduló él miatt változik: biztosan van egy 1-es él v_6-ből v_5-be.

6. Rendeljen hozzá élsúlyokat az alábbi gráf éleihez úgy, hogy a keletkező gráfban Dijkstra algoritmusa rosszul számolja ki a legrövidebb utak hosszait.





Itt Dijskstra algoritmusa a c csúcs távolságát az első lépésben 1-re állítja, pedig van -1-es út is.