

ismekes:

lépésenként

	maximális $O(n^2)$	lehetővé $O(n+c)$
van-e az $i \rightarrow j$ kötés	$A[i][j] \rightarrow 0$ $O(1)$	$L[i] \rightarrow$ láncos végig $O(L[i] \text{ hosszú}) = d(i)$
minden $d(i)$ kiszámítás	$\sum_{j=1}^n A[i][j]$ $O(n)$	$L[i] \rightarrow$ láncos végig összesítés $O(L[i] \text{ hosszú}) = d(i)$
minden $d_i(i)$	$\sum_{i=1}^n A[i][j]$ $O(n)$	$L[i] \rightarrow$ láncos végig összesítés $O(L[i] \text{ hosszú}) = d_i(i)$
$d_{\text{össz}}(i)$	$\sum_{i=1}^n A[i][j]$ $O(n)$	$i=1 \dots n$ $L[i]$ láncos végigjárás $O(n+c)$
i összehasonlítás végigjárás	$O(n)$	$d(i)$

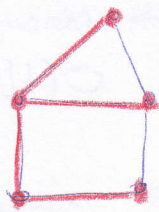
lépésenként függés a gráf megadásától módjaitól

($F \subseteq A$: körmentes, össefüggő irányított gráf)

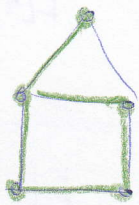
Fenikőfa : egy G irányított gráfban F

- ha $F \subseteq A$
- F minden ele G -nek is ele
(F szubgráfja G -nek)
- G minden csúcsa benne van F -ben

Pl:



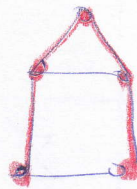
IFEN



NEM



NEM



IFEN

Mikor van G -ben fenikőfa?

Tétel: Ha van G -ben fenikőfa, akkor G összefüggő.

Biz: Ha G -ben van F fenikőfa $\Rightarrow F$ elein át minden csúcsból minden csúcs elérhető, de F elei G -nek is elei, ami alapján G is összefüggő.

Tétel: Összefüggő gráfban van fenikőfa.

Biz: Ha G körmentes, akkor $F = G$

Ha G nem körmentes $\rightarrow$ van benne legalább

egy kör $\rightarrow$ hagyjuk el ennek egy élét

$\rightarrow G' \rightarrow G'$ minden csúcsát lefedi és

G' továbbra is összefüggő $\rightarrow$ ha G' körmentes

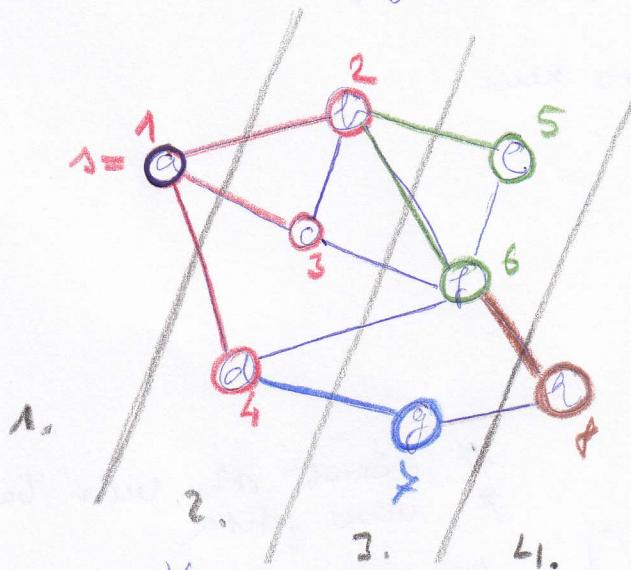
akkor $F = G' \rightarrow$ ha nem újra is újra töröljük

[8] - ? - egy kör elét $\rightarrow G'' \dots$
végtelenül megyünk tovább körökkel.

BFS (Bread First Search) : algoritmus.

Sílességi bejárás:

Cél: irányítatlan G -ben felkölés
(irányítatlan gráf ciklusmentes bejárása)



- Van egy kezdőcsomópont: s
 - 1 körben s szomszédainak megkeresése
 - 2 körben s szomszédainak új szomszédainak
↳ HA MEG NEM VOLT
- Ha u_1 és u_2 két elemről tudjuk meg, hogy u_1 és u_2 között van egy út, akkor u_1 szomszédait u_2 szomszédai elöljárókat is megkeressük.
- ezután k körben az előző körben bejártak új szomszédait járjuk be

Állítás: Ha G összefüggő, akkor a felfedezés élek
(amikkel új ismeretlen csomópontokhoz juthatunk) felkölés
alkotnak. $\neq$

Biz.: ① T összefüggő mint az U és minden
kapcsolat a korábbiakhoz

② T kömenszer mint az U és minden U csomópont
veszték

③ ~~minden~~ G minden csomópont elemének
mint ha minden csomópont

BFS(G, s): algoritmus:

bejövőki nyilvántartás:

$$\text{bejövőki}[v] = \begin{cases} 1 & \text{ha } v = s \\ 0 & \text{egyébként} \end{cases}$$

töltés az U és U
van jónak

$T = \emptyset$ # jelölés az U halmaz

$Q = \{s\}$ # az U és az U halmaz
az U és az U halmaz
az U : Q

az U : FIFO lista

az U az U halmaz
az U az U halmaz

az U : az U halmaz

az U az U halmaz
az U az U halmaz

pseudó kód:

while $Q \neq \emptyset$:

$v := Q$ első eleme

Q -ból v -t kivessz

v mindenkivel összekapcsolva:

if $bej\acute{a}rva[w] == 0$:

$bej\acute{a}rva[w] = 1$

w -t Q -ba rakd

T -be $(v \rightarrow w)$ él hozzáadása

példa: 3. oldal

a, b, c, d, e, f, g, h gráf csomópontjai

bej\acute{a}rva							
a	b	c	d	e	f	g	h
1	0	0	0	0	0	0	0

Q

a

T

$\emptyset$

$v = a$

összekapcsolva: ~~b~~, c, d

1	1	1	1				
---	---	---	---	--	--	--	--

~~b~~, c, d

ab, ac, ad

$v = b$

összekapcsolva: ~~a~~, ~~c~~, e, f

1	1	1	1	1	1		
---	---	---	---	---	---	--	--

c, d, e, f

ab, ac, ad,

bc, bf

$v = c$

összekapcsolva: a, ~~b~~, f

a, e, f

ab, ac, ad,

bc, bf

1	1	1	1	1	1	1	
---	---	---	---	---	---	---	--

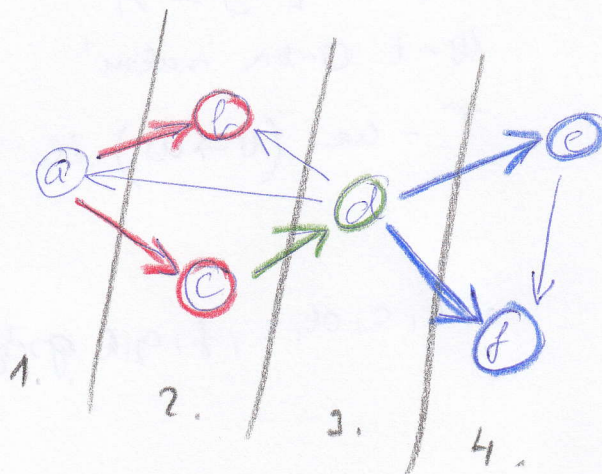
e, f, g

Megjegyzés:

Ha G irányított és a működés ugyanígy működik, csak u akkor nyomozja v -re ha $(v \rightarrow u)$ él létezik

és a végére G minden olyan csomópontjára eljutunk amihez van út s -ből

pelda



LS2:

① inicializálás:

tömb létrehozása:	n	} $n+2$ $O(n)$
T	1	
Q	1	

② legfeljebb n -szer fut le while ciklus,
mert minden csomópont csak egyszer lehet a
sor elején

v minden u nyomozására legfeljebb

4 lépés $\rightarrow d(v)$

mindig megadással
végigmenni a nyomozáskor
 $\rightarrow n \rightarrow O(n)$

először megadással

$O(d(v))$

$\Rightarrow$ Összesen lépés szám:

malix	lista
$O(n^2)$	$O(n + e)$

Mire jó még BFS?

① G összefüggőségének eldöntése:
Teljesen kezdőponttól
BFS befuttatása után ha maradt
alulról még bejárva $\rightarrow$ nem összefüggő
egyébként összefüggő

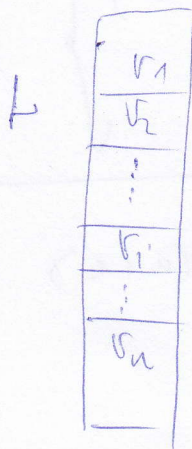
② Ha nem összefüggő a bejárást az első
nem bejárt csomóponthoz lehet újra elindítani
a BFS-t ... mindaddig amíg az ~~az~~
összes csomó be nem járva.

③ a kezdőcsomóponthoz (s) való távolságot
a hinta követésével lehet meghatározni
(hány élre kell végighaladni s -től v -ig
való út mentén)

KÖZMŰLTATÁS

6/9/9

Indígytalan, egyenlő Belléskvél adott.

van-e megoldás? $O(n+e)$ lépésben

Algo: végigmegyek az élleken
 és $\forall v_i$ esetén kiírdmolek
 $d(v_i) - 1$ és ha $d(v_i) = 2$
 akkor kén $\rightarrow$ van 2- pár
 kiírdmolek más nyelven

for $i = 1$ to n .

végigmegyek $L(v_i) - n$ és
 kiírdmolek a kén $d(v_i)$
 ha $d(v_i) == 2$

Jószág: kiírdmolek $\forall d(v_i) - 1$

LS: v_i -vel a kiírdmolek $1 + d(v_i)$ lépés
 önmolek végig a kiírdmolek $O(n+e)$

b) max ~~szám~~ fokszám :

Algo: végig az élleken mind v_i -ket
 $d(v_i)$ kiírdmolek és kiírdmolek kiírdmolek
 és kiírdmolek

$d(v_1)$	$d(v_2)$	$\dots$	$d(v_n)$
----------	----------	---------	----------

oután maximum keresése a tömbben

Fő: $\forall$ példánál legyen működés

Maximum keresése után tömbbe való visszaírás

L2:

$$\left. \begin{array}{l} \text{előzetes} : O(n+e) \\ \text{maximumkeresés} : O(n) \end{array} \right\} +$$

$$O(n) + O(n+e) = O(n+e)$$

9) leggyakoribb példák

~~Algo~~ Algo: tömb elemeinek minél több feladat

oután: 1. max példán minél több
előzetes 3 tömb

2. oután: (előzetes alapján) for $i=0$ to $n-1$
~~if (arr[i] > max)~~

$$B[arr[i]] += 1$$

3. vagy maximum keresése B-ben

Jódd: 1. utdw minden példán megvan
2. utdw példánál valószínűsége
3. ebben megkeresés a maximum

L2: $O(n+e) + O(n) + \underbrace{O(\log(B))}_{O(n)} = O(n+e)$

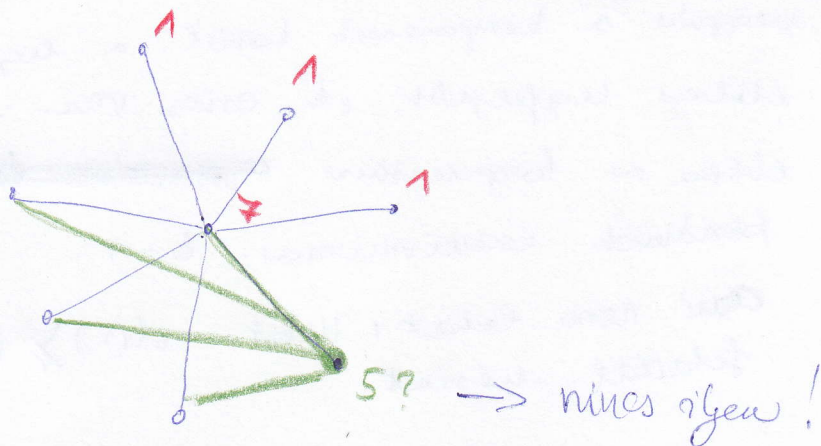
Más megfigyelés: 1.) bármely megfigyelés

2.) két rendszer $m = n$ láthatósága

3.) ~~figyelés~~ még a rendszer tömbje is
kiszámolható a csatlakozások és a legnagyobb
a megoldás.

6/10/a)

~~1, 1, 1~~ 2, 3 4, 5, ~~6~~



6) 1, 1, 1, 2, 3, 4, 5, 6

bármely öszi: páros kell, hogy legyen

$$\sum d(i) = 2e$$

$$1+1+1+2+3+4+5+6 = 2e$$

$$23 = 2e$$

NEH PÁROS!

6/13

P : egyszerű, irányítottan

$$n = 2k$$

$$d(i) \geq k$$

Bm.: (indirect)

Tfl.: $\leftarrow$ legalább 2 komponens van

- vegyük a komponensek közül a legkisebbet
 $\Rightarrow$ ebben legfeljebb " k " csúcs van.

ebben a komponensben ~~legfeljebb $k-1$ csúcs~~ a
legkisebb maximuma $k-1$

ami nem lehet, mert $d(i) \geq k$ kell a
feladat szerint