

DAG, jellemzése, **topologikus sorrend** keresése. Leghosszabb utak keresése, **PERT-módszer**, kritikus utak és tevékenységek.

- **DAG (Direct Acyclic Graphs)**, jellemzése

Def: A $G = (V, E)$ irányított gráf **aciklikus** (más néven **DAG**), ha G nem tartalmaz irányított kört.

Példa: DAG-ot úgy kaphatunk, hogy egy G irányítatlan gráf csúcsait csupa különbözőszámmal megszámozzuk, és minden élt a kisebb számot viselő csúcsból a nagyobbba irányítunk.

Ha ugyanis lenne az így megirányított gráfban irányított kör, akkor az élei mentén a számok végig növekednének, ami lehetetlen. Azt fogjuk igazolni, hogy a fenti példa minden DAG-ot leír.

- **topologikus sorrend** keresése

Def: A $G = (V, E)$ irányított gráf csúcsainak **topologikus sorrendje** alatt a csúcsok olyan sorrendjét értjük, amire igaz, hogy minden irányított él a sorban előbb álló csúcsból vezet a sorban későbbi csúcsba. ($V = \{v_1, v_2, \dots, v_n\}, v_i v_j \in E \Rightarrow i < j$)

Tétel: (G irányított gráf DAG) $\Leftrightarrow (V(G)$ -nek \exists topologikus sorrendje).

Biz: Tegyük fel, hogy \exists topologikus sorrend. Láttuk, hogy G ekkor DAG. ✓

Biz: Most tegyük fel, hogy G DAG, és futtassunk rajra egy DFS-t. Láttuk, hogy a DFS után nem lesz visszaél, ezért minden uv irányított élre $b(u) > b(v)$ teljesül. Ezért a csúcsok befejezési sorrendjének megfordítása a G csúcsainak egy topologikus sorrendje. □

Köv: Irányított gráf aciklikussága DFS-sel gyorsan eldönthető: ha van visszaél, akkor a visszaél DFS-fabeli alapköre G egy irányított köre, így G nem DAG. Ha pedig nincs visszaél, akkor a fordított befejezési sorrend a G egy topologikus sorrendje, G tehát DAG.

Megj: DAG-ban topologikus sorrendet forráskeresések és forrástörlések alkalmazásával is találhatunk.

- Leghosszabb utak keresése

Ötlet: Az $l'(uv) = -l(uv)$ élhosszokkal a leghosszabb utak legrövidebbekké válnak. Olyanokat pedig tudunk keresni.

Gond: A módszerünk csak konzervatív élhosszokra működik. Irányítatlan gráfon ez nemnegatív élhosszokat jelent, ezért ez az ötlet itt nem segít. Irányított esetben nem baj a negatív élhossz, feltéve, hogy G DAG. Ekkor Ford, Floyd bármelyike használható.

Jó hír: Van egy még gyorsabb módszer: a dinamikus programozás. Ennek segítségével tetszőleges G DAG minden v csúcsához ki tudjuk számítani a v -be vezető leghosszabb utat. (Sőt! ...)

Leghosszabb út DAG-ban:

Input: $G = (V, E)$ DAG, $l : E \rightarrow \mathbb{R}$.

Output: $\max\{l(P) : Pv\text{-be vezető út}\}$ minden $v \in V$ csúcsra.

Működés:

- 1 $V = \{v_1, v_2, \dots, v_n\}$ topologikus sorrend meghatározása.
- 2 $i = 1, 2, \dots, n : f(v_i) = \max\{\max\{f(v_j) + l(v_j v_i) : v_j v_i \in E\}, 0\}$

Output: $f(v) \forall v \in V$

Helyesség: Ha a v_i -be vezető leghosszabb út utolsó előtti csúcsa v_j , akkor $f(v_i) = f(v_j) + l(v_j v_i)$.

Megj: Ha a fenti algoritmusban minden csúcsra megjelöljük az $f(v)$ értéket beállító élt (éleket), akkor a megjelölt élek minden v csúcsba megadnak egy leghosszabb utat. Sőt: minden v -be vezető leghosszabb megkapható így.

- **A PERT probléma**

Egy a, b, \dots tevékenységekből álló projektet kell végrehajtanunk.

Precedenciafeltételek: bizonyos (u, v) párok esetén előírás, hogy az u tevékenységet a v előtt kell elvégezni, ezért v az u kezdetét követően $c(uv)$ időkorlát elteltével kezdhető.

Cél: minden v tevékenységhez olyan $k(v) \geq 0$ kezdési időpont meghatározása, ami nem sérti a precedenciafeltételeket, és a projekt végrehajtási ideje (a legnagyobb $k(v)$ érték) minimális.

G **irányított gráf** csúcsai a tevékenységek, élei pedig a precedenciafeltételek, az uv él hossza $c(uv)$.

Megf:

(1) Ha G nem DAG, akkor a projekt nem hajtható végre.

(2) Ha G DAG, akkor minden v tevékenység legkorábbi kezdési időpontja a v -be vezető leghosszabb út hossza.

Köv: A PERT probléma megoldása nem más, mint a G DAG minden csúcsára az oda vezető leghosszabb út meghatározása.

- Kritikus utak és tevékenységek

Terminológia: G leghosszabb útja **kritikus út**, amiből több is lehet. Kritikus út csúcsai a **kritikus tevékenységek**.

Megf: Ha egy kritikus tevékenység nem kezdődik el a lehető legkorábbi időpontban, akkor az egész projekt végrehajtása csúszik.