Mélységi keresés és alkalmazásai (fellépő éltípusok, mélységi- és befejezési számozásból az éltípus meghatározása, irányított kör létezésének eldöntése DFS-sel).

- Általános gárfbejárás: A gráfbejárási algoritmus az inputgráf csúcsait és éleit fedezi fel. Minden csúcs az eléretlen → elért → befejezett állapotokat veszi fel. A bejárás akkor ér véget, amint minden csúcs befejezetté vált.

  1. Van elért csúcs. Választunk egyet, mondjuk u-t.
     (1a) Ha van olyan uv él, amire v eléretlen, akkor v elérté válik.
     (1b) Ha nincs ilyen uv él, akkor u befejezetté válik.
  2. Nincs elért csúcs.
     (2a) Ha van eléretlen u csúcs, akkor u-t elértté tesszük.
     (2b) Ha nincs eléretlen csúcs (azaz minden csúcs befejezett), akkor END.

  Az élek osztályozása:

  faél: Olyan él, ami mentén egy csúcs elértté vált.

  uv **előreél:** nem faél, de u-ból v-be faélekből irányított út vezet.

  uv **visszaél:** v-ből u-ba faélekből irányított út vezet.

  keresztél: minden más él (u és v közt nincs leszármazott viszony).

- **Mélységi keresés (DFS (Depth First Search))**

  (A mélységi bejárás avagy DFS alatt olyan gráfbejárást értünk, amikor mindig a legutolsónak elért csúcsból kerül elérésre a soron következőnek elért csúcs. Az elérési illetve befejezési sorrendből adódik minden v csúcshoz egy $m(v)$ mélységi ill. $b(v)$ befejezési szám.)

  "Mélységi bejárás (DFS): A bejárás során mindig a legutolsónak elért csúcsot választjuk az $\boxed{1.}$ esetben.

  Mélységi és befejezési számozás: DFS után $m(v)$ ill. $b(v)$ a v csúcs elérési ill. befejezési sorrendben kapott sorszáma.

  Megj: A BFS konkrét megvalósításában szükség van arra, hogy az elért csúcsokat úgy tároljuk, hogy könnyű legyen kiválasztani az elért csúcsok közül a legkorábban elértet. Erre egy célszerű adatstruktúra a *sor* (avagy *FIFO lista* (First In First Out)). Ha a BFS megvalósításában ezt az adatstruktúrát *veremre* (más néven *LIFO listára* (Last In First Out)) cseréljük, akkor a DFS egy megvalósítása adódik.

  Megf: Tegyük fel, hogy a G gráf éleit DFS után osztályoztuk.

  (1) Ha uv faél, akkor $m(u) < m(v)$ és $b(u) > b(v)$.

  Biz: v-t u-ból értük el, ezért $m(u) < m(v)$. A v elérésekor u és v elért állapotúak. A DFS szerint v-t u előtt fejezzük be.

  (2) Ha uv előreél, akkor $m(u) < m(v)$ és $b(u) > b(v)$.

  Biz: u-ból v-be faéleken keresztül vezet irányított út. (1) miatt az út mentén a mélységi szám növekszik, befejezési csökken.

  (3) Ha uv visszaél, akkor $m(u) > m(v)$ és $b(u) < b(v)$.

  Biz: v-ből u-ba faéleken keresztül vezet irányított út. (1) miatt az út mentén a mélységi szám növekszik, a befejezési csökken.

  (4) Ha uv keresztél, akkor $m(u) > m(v)$ és $b(u) > b(v)$.

  Biz: $m(u) < m(v)$ esetén a DFS miatt v az u leszármazottja lenne. Ezért $m(u) > m(u)$. Ha u-t a v befejezése előtt érnénk el, akkor u a v leszármazottja lenne. Ezért az alábbi sorrendben történik u és v evolúciója: v elérése, v befejezése, u elérése, u befejezése.

  (5) Irányítatlan gráf DFS bejárása után nincs keresztél.

  Biz: Indirekt. Ha uv keresztél, akkor (4) miatt $m(u) > m(v)$, továbbá vu is keresztél, ezért $m(v) > m(u)$. Ellentmondás.

  (6) Ha DFS után van visszaél, akkor G tartalmaz irányított kört.

  Biz: A DFS fa visszaélhez tartozó alapköre a G egy irányított köre.

  (7) Ha DFS után nincs visszaél, akkor G-ben nincs irányított kör.

  Biz: Bármely irányított körnek van olyan uv éle, amire $b(u) < b(v)$. Ez az él csak visszaél lehet.

  A mélységi bejárás lépésszáma lineáris, azaz van olyan c konstans, hogy tetszőleges u csúcsú, m élű gráf DFS-éhez legfeljebb $c(n+m)$ lépés szükséges.

- **Directed Acyclic Graphs**

  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 különböző számmal megszámozzuk, és minden élt a kisebb számot viselő csúcsból a nagyobba 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.

  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. $(V = \{v_1, v_2, \ldots, v_n\}, v_i v_j \in E \Rightarrow i < j)$

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

  Biz: Tegyük fel, hogy $\exists$ toplogikus 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.