Algoritmusok és adatszerkezetek 2.

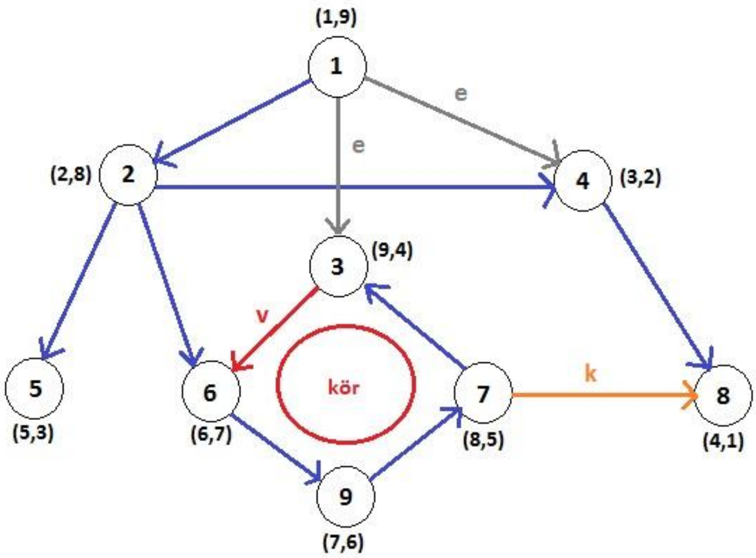
Fekete István előadása alapján
Készítette: Nagy Krisztián

7. előadás

27. Mélységi bejárás

A feladat „algoritmikus specifikációja”:

Megállapodás: Az élek (a csúcsok alapján) alfabetikus sorrendben kerülnek feldolgozásra.

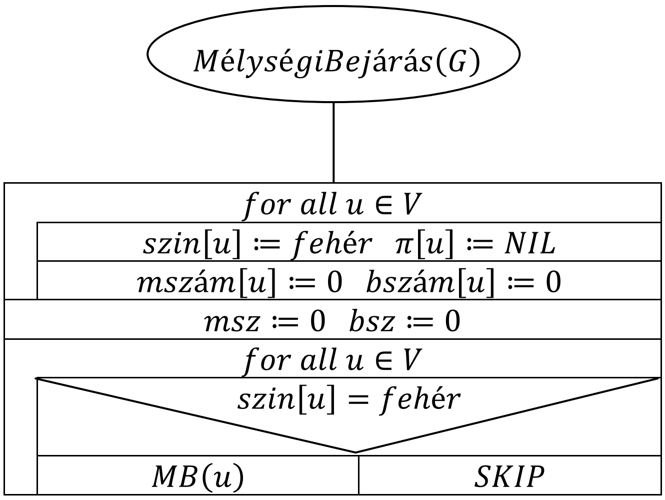


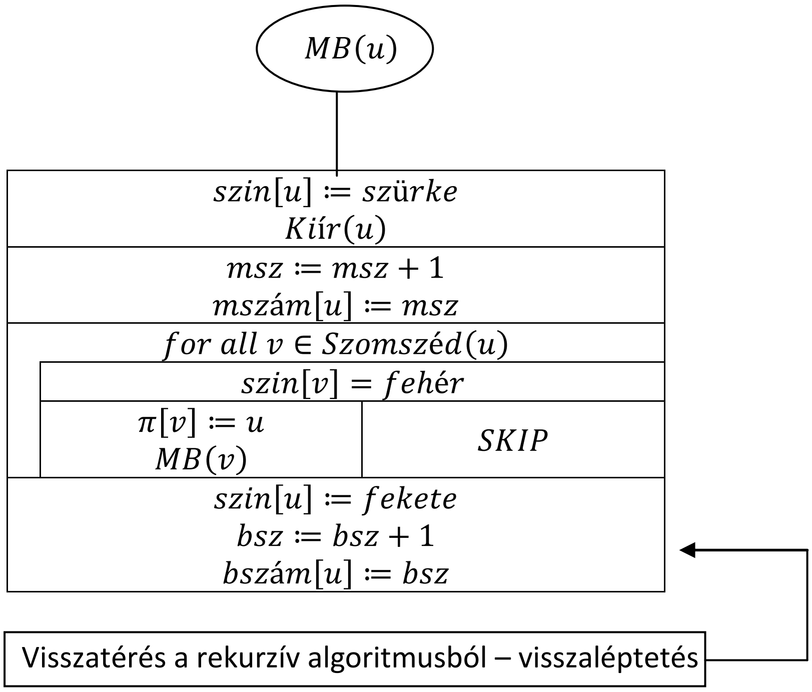
Jelölések (kiegészítés):

- Kék nyilak: a mélységi feszítőfa elemei
- (x, y):
  - x a mélységi bejárás során a belépési szám. Hanyadik lépésben érjük el az elemet.
  - y a mélységi bejárás során a kilépési vagy más néven befejezési szám

Éltípusok:

- Előreél: lásd szürke (e)
- Keresztél : lásd narancs (k)
- Visszaél : lásd piros (v)





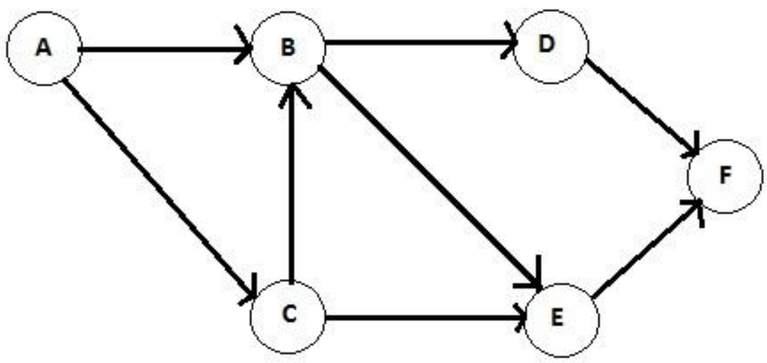
<u>Műveletigény:</u> $\theta(n+e)$

28. DAG topologikus rendezése

Irányított körmentes gráf   (Directed Acyclic Graph)

A kör a gyakorlatban általában közbeavatkozást jelent, ami nem kívánatos jelenség.
A DAG itt egy gyártási folyamat, ami előfeltételeket rögzít $\Longrightarrow$ linearizálás

Példa:



Megoldás: ACBDEF vagy ACBEDF

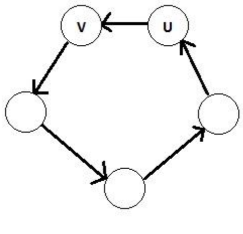
DAG tulajdonság ellenörzése

Állítás: Ha G-nek egy (bizonyos) mélységi bejárása talál visszaél-t, akkor G nem DAG.

Bizonyítás: Például: Ha (u, v) visszaél, akkor volt $v \rightsquigarrow u$ út, a mélységi bejárás szerint. Ezek együtt kört alkotnak.

Állítás: Ha G nem DAG, akkor bármely mélységi bejárás kimutat/talál visszaél-t

Bizonyítás:



A kiválasztott mélységi bejárás a v-re lép először a körben.
A mélységi bejárás v-ből eljut u-ba: $\exists v \rightsquigarrow u$ út. Ezután u-ból v-be átnézve (u, v) visszaélnek látszik.

Definíció:
$G = (V, E)\ DAG$
$V = \{u_1, u_2 \dots u_n\}$
$T = \{x_1, x_2 \dots x_n\}$ topologikus rendezés ⇔ $\forall (u, v) \in E$ élre, amennyiben $u = x_i\ v = x_j$, akkor $i < j$

Lemma: $G\ DAG \implies \exists u \in V: befok(u) = 0$
Bizonyítás: Indirekten, végtelen hátrálás ⟹ kör.

Tétel: Ha G-nek van topológikus rendezése ⇔ G DAG

Bizonyítás:

⟹: Ha $G\ nem\ DAG \implies$ ∃kör ⟹ ∄topologikus rendezés

⟸: $G\ DAG \implies \exists G$-nek topológikus rendezése: Teljes indukcióval, algoritmikusan látjuk be.
Lemma ⟹ ∃ olyan csúcs, amelybe nem vezet él ⟹ ez lesz T első eleme
$T(G) = \{A, T(G')\}: G' = G \backslash \{A\}$ ⟹ Ebből lehetne topologikus rendezés eljárás.