Graphentheorie

Graph: $G = (V, E)$
Baum: $|E| = |V| - 1$
Spannbaum von G: Teilgraph von G und Baum der alle Knoten von G entällt, not unique. Bipartit: $G = (V_1 \cup V_2, E)$, G ist bipartit $\Leftrightarrow$ G enthällt keinen Kreis gerader länge

Flussnetze:
$N = (D, \kappa, s, q)$, D Digraph, $\kappa : E \rightarrow \mathbb{R}_0^+$ Kostenfunktion

Schnitt eines Flussnetzes:
Teilmenge S, die die Quelle aber nicht die Senke entällt.
Kapazität eines Schnittes: $\kappa(S)$ = Kapazität der Endknoten des Schnittes. Minimaler Schnitt S = $\forall S'\ \kappa(S) \leq \kappa(S')$
maximaler Fluss == min Schnitt

Planarität

für ebene Darstellungen gelten: $n - m + f = 2$,n=Knoten, m=Kanten, f=Flächen
if $n \geq 3$ $3n - 6$ Kanten höchsten
if $n \geq 3$ hat höchstens $g \geq 3$, g = Umfang des Graphen???? $max\{g(n-2)/(g-2)mn-1\}Kanten$
ein Graph ist planar $\Leftrightarrow$ kein subgraph von G ist homöomorph zu k_5, $k_{3,3}$

Datenstrukturen

Adjazenzmatrix

> $n \cdot n$, immer symetrisch
> $a_{ij} = 1$ falls $v_i v_j \in E$, 0 sonst

Inzidenzmatrix

> $n \cdot m$, $e_{ij} = 1$ wenn v_i mit e_j inzidiert, 0 sonst
> spaltensume immer 2, reihensumme = Grad des Knoten

Netzwerke

Floyd-Warshal (S.288)

> Kürzeste Abstände für alle Knoten $O(|V|^3)$
> siehe auch Dijkstra (S.289)
>
> for k=1 to n do: $d(u, w) = min(d^{k-1}(u, w), d^{k-1}(u, v_k) + d^{k-1}(v_k, w))$
> Mit jeder Iteration gucken ob es einen kürzeren Weg über den Knoten v_k gibt

Kurskal (S.291)

> min. Spannbäume

Ford-Fulkerson (S.293)

> bestimmet maximalen Fluss in N
> erst alle Knoten markieren, dann Fluss vergrössern und erneut markieren.