

Algorithm Engineering

Übung 6

Aufgabe 1:

Ein *Feasible-Flow-Problem* wird definiert durch ein Netzwerk in dem außer den oberen Kapazitätsschranken u_{ij} für jede Kante $(i, j) \in E$, zusätzlich ein *Supply-Wert* $b(i)$ für jeden Knoten $i \in V$ gegeben ist. Gesucht ist eine Flussfunktion x , die die Kapazitätsbedingung erfüllt, so dass für alle Knoten $i \in V$ gilt:

$$b(i) = \sum_{j \in V \text{ mit } (i,j) \in E} x_{ij} - \sum_{j \in V \text{ mit } (j,i) \in E} x_{ji}$$

Überlegen Sie sich wie man dieses Problem mit einem Maxflow-Algorithmus lösen kann. *Hinweis:* Erweitern Sie das Netzwerk um 2 Knoten und einige Kanten.

Aufgabe 2

Implementieren Sie eine C++Funktion `orientation` zur Berechnung des *orientation*-Prädikats für 3 Punkte a, b, c .

Aufgabe 3

Schreiben Sie eine Funktion zur Berechnung des Schnittpunktes von zwei beliebigen Strecken $s1$ und $s2$. Testen Sie dazu zunächst, unter Verwendung der `orientation`-Funktion, ob sich die beiden Segmente in genau einem Punkt schneiden und berechnen Sie dann den Schnittpunkt der beiden entsprechenden Geraden.