operations research
QUANTITATIVE INFRASTRUKTURSYSTEMMODELLIERUNG

Tutorium

Operations Research – Grundlagen

Technische Universität Berlin
Fachgebiet **W**irtschafts- und **I**nfrastruktur**P**olitik

WIP

Flüsse in Netzwerken:

Ford-Fulkerson und Max-Flow

Tutoriumsaufgaben:

3.6, 3.7	Ford Fulkerson Algorithmus
3.8	Flussmaximierungsproblem
3.9	Modellierung mit Max-Flow

Allgemeines

Netzwerk: Tupel (V,E,s,t,u,c) bestehend aus

- einem endlichen, gerichteten Graphen (V,E)
- mind. zwei spezifischen Knoten s (Quelle), t (Senke)
- einer Kapazitätsfunktion $u : E \rightarrow R+$
- (optional) einer Kostenfunktion $c : E \rightarrow R+$

Residualgraph: zeigt die restlichen Kapazitäten eines Netzwerkes an

Ford Fulkerson Algorithmus

目的是什么?
为了在网络中确定两个节点之间的最大流量。

步骤是什么?
1. 初始化：将每条边上的流量设置为0。
2. 构建网络的剩余图。在初始化之后，这个剩余图看起来和原始图完全一样。
3. 找到一条从s到t的路径P，使得在剩余图中仍存在路径上的容量。
4. 确定路径在剩余图中的容量cap(P)。
5. 减去剩余图中所有边的容量cap(P)。
6. 重复步骤3直到在剩余图中不存在从s到t的路径具有正容量为止。

Wozu?

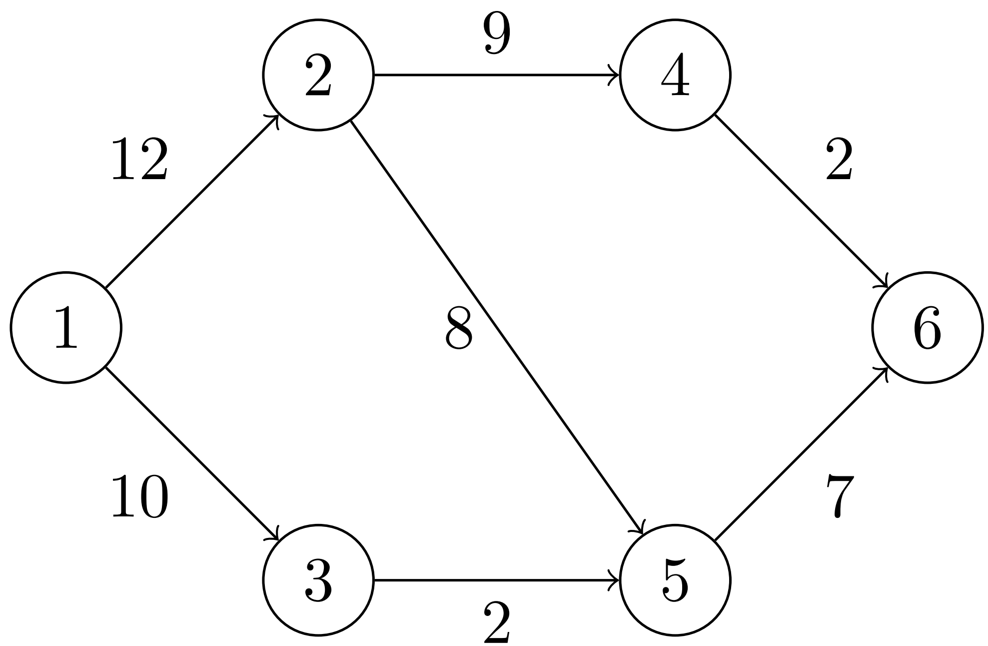
Um in einem Netzwerk den *maximalen Fluss zwischen zwei Knoten* zu ermitteln

Vorgehen?

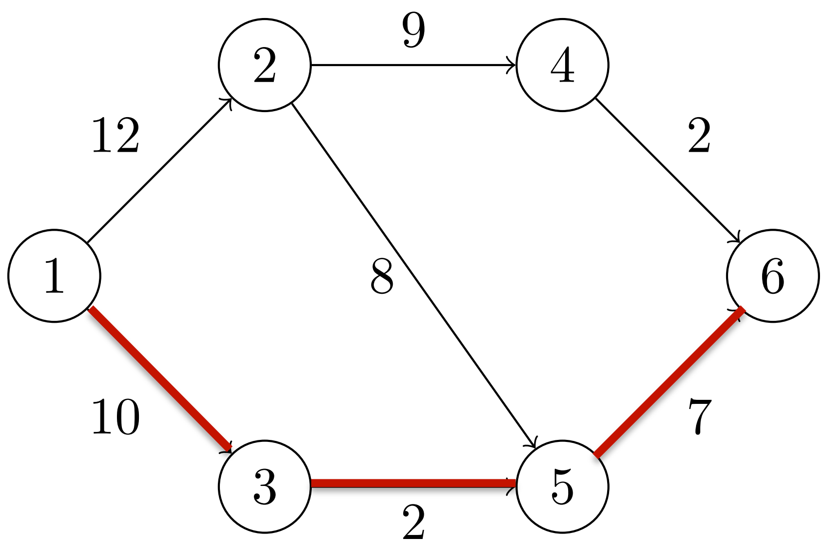
1. Initialisierung: Setze den Fluss auf jeder Kante auf 0.
2. Bilde den Residualgraphen für das Netzwerk. Direkt nach der Initialisierung sieht dieser genau wie der Originalgraph aus.
3. Finde einen Pfad P von s zu t für den es noch Kapazität im Residualgraphen gibt.
4. Bestimme die Kapazität $cap(P)$ des Pfads im Residualgraph.
5. Ziehe die Kapazität $cap(p)$ von allen Kanten im Residualgraphen ab.
6. Wiederhole den Prozess ab Schritt 3 bis kein Pfad mit positiver Kapazität mehr zwischen s und t im Residualgraphen existiert.

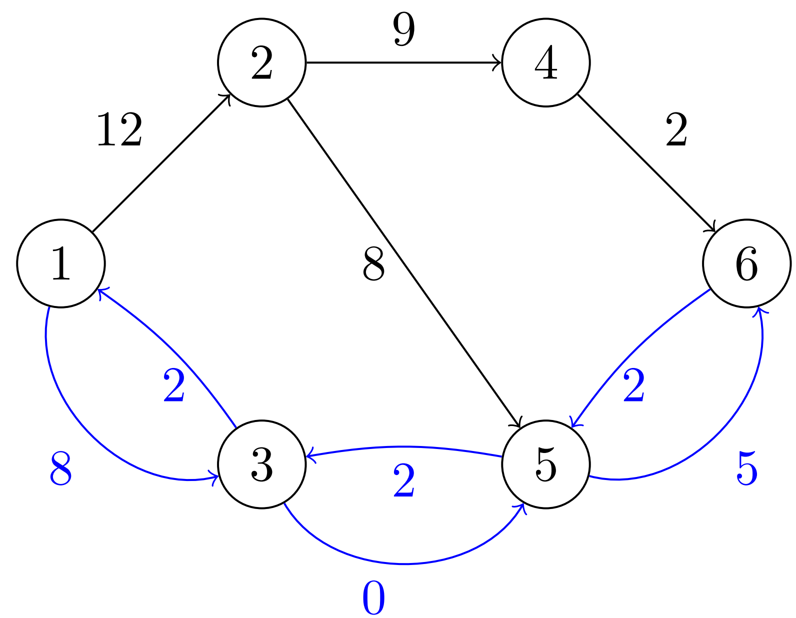
Ford Fulkerson Algorithmus: Aufgabe x.1

Wenden Sie den **Ford Fulkerson Algorithmus** an, um die **Gesamtkapazität** des Netzwerkes zu ermitteln (Quelle in Knoten 1 und Senke in Knoten 6).



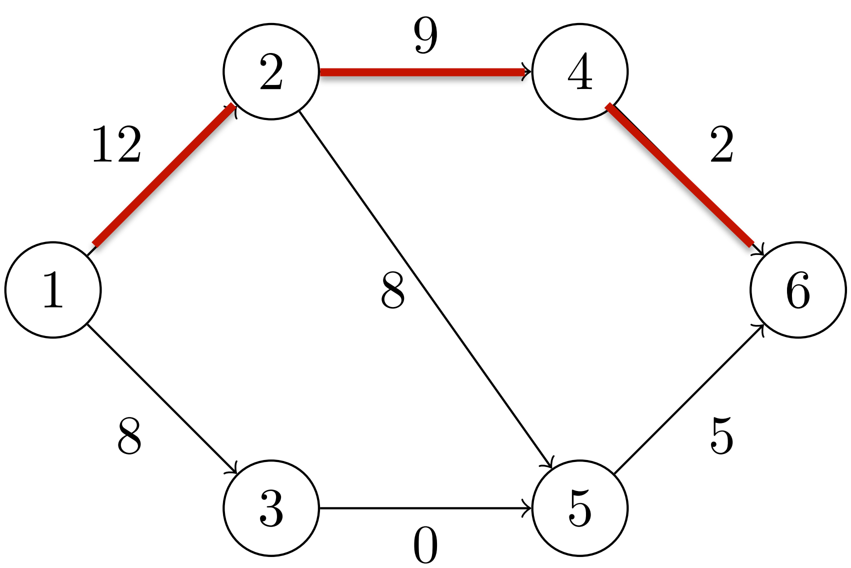
Ford Fulkerson Algorithmus: Aufgabe 3.6

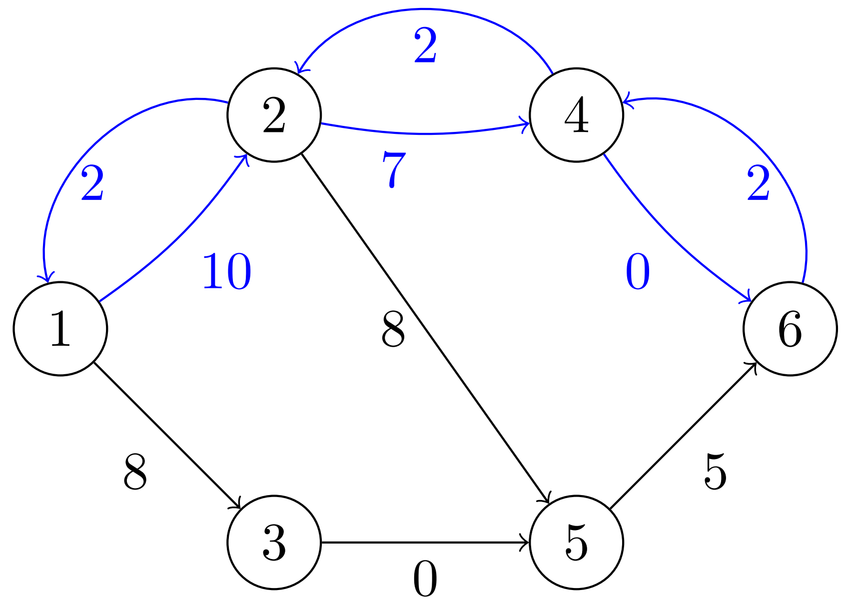




Kapazität Pfad 1 = 2

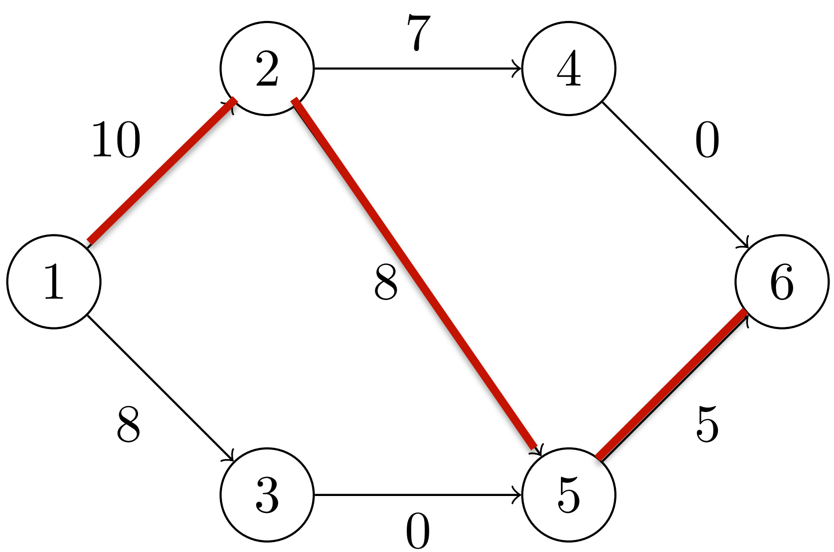
Ford Fulkerson Algorithmus: Aufgabe 3.6

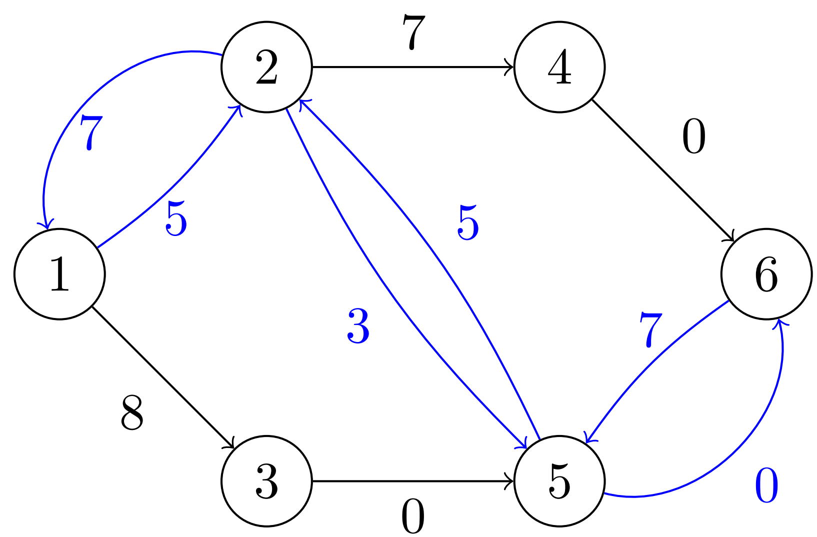




Kapazität Pfad 2 = 2

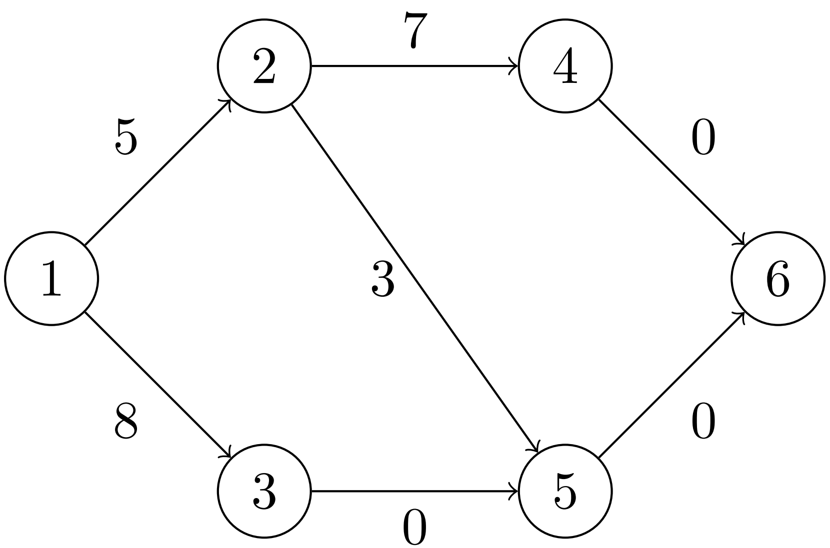
Ford Fulkerson Algorithmus: Aufgabe 3.6





Kapazität Pfad 3 = 5

Ford Fulkerson Algorithmus: Aufgabe 3.6



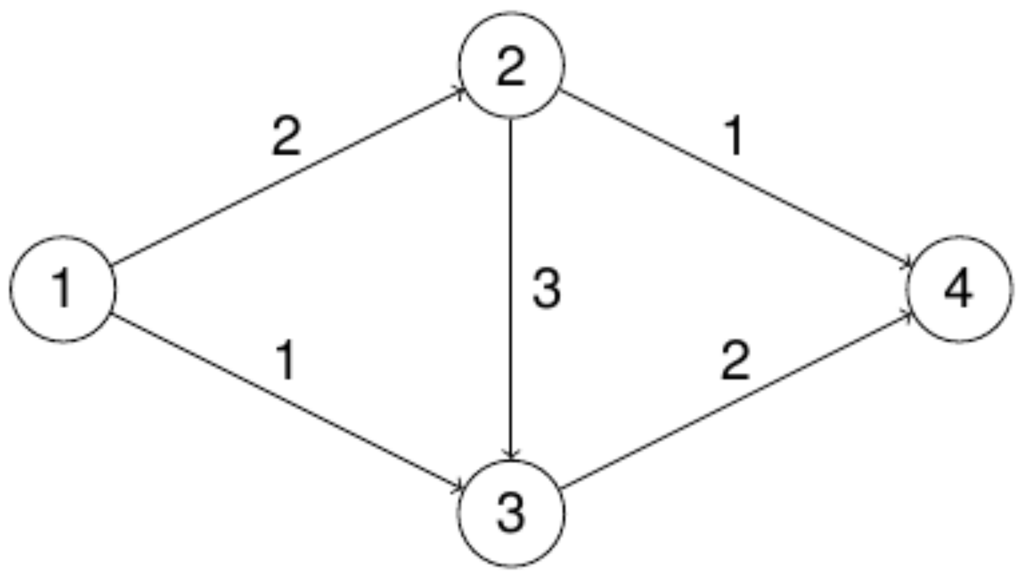
Abbruch, da kein möglicher Pfad mehr Kapazität hat.

Gesamtkapazität

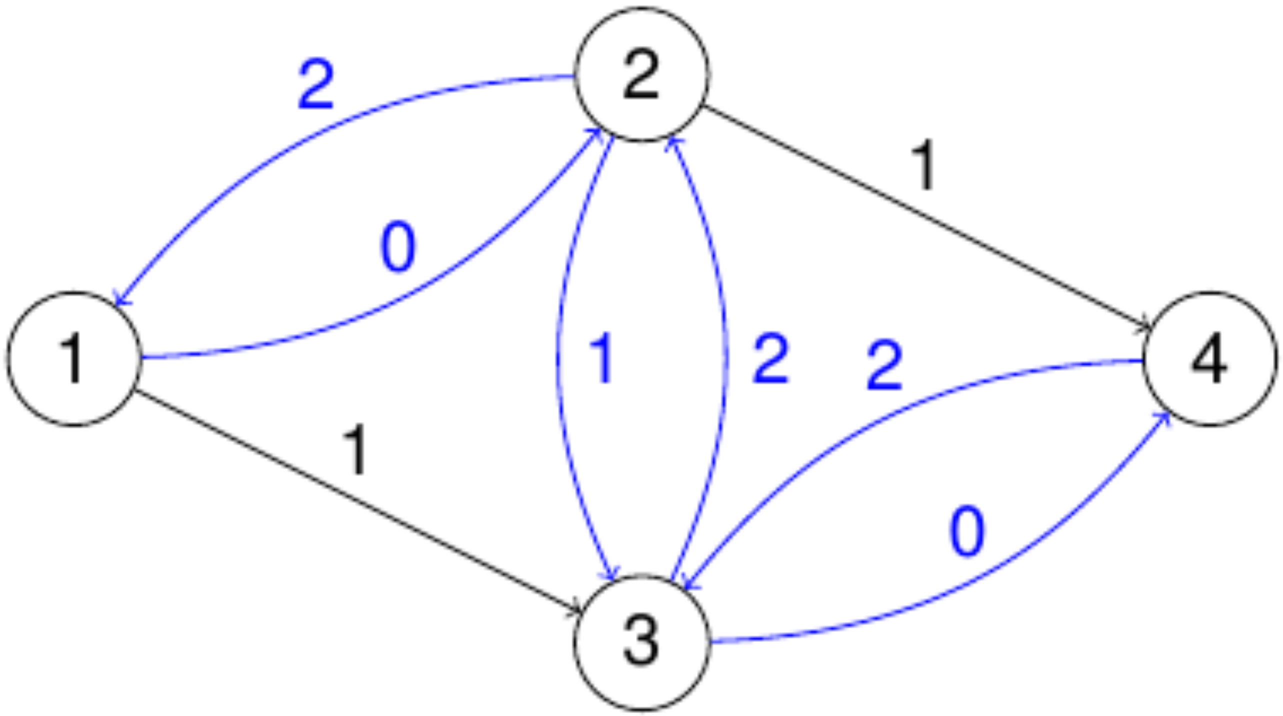
= 2 (P1) + 2 (P2) + 5 (P3)
= 9

Ford Fulkerson Algorithmus: Aufgabe 3.7 a)

Wenden Sie den **Ford Fulkerson Algorithmus** an, um die **Gesamtkapazität** des Netzwerkes zu ermitteln (Quelle in Knoten 1 und Senke in Knoten 4).

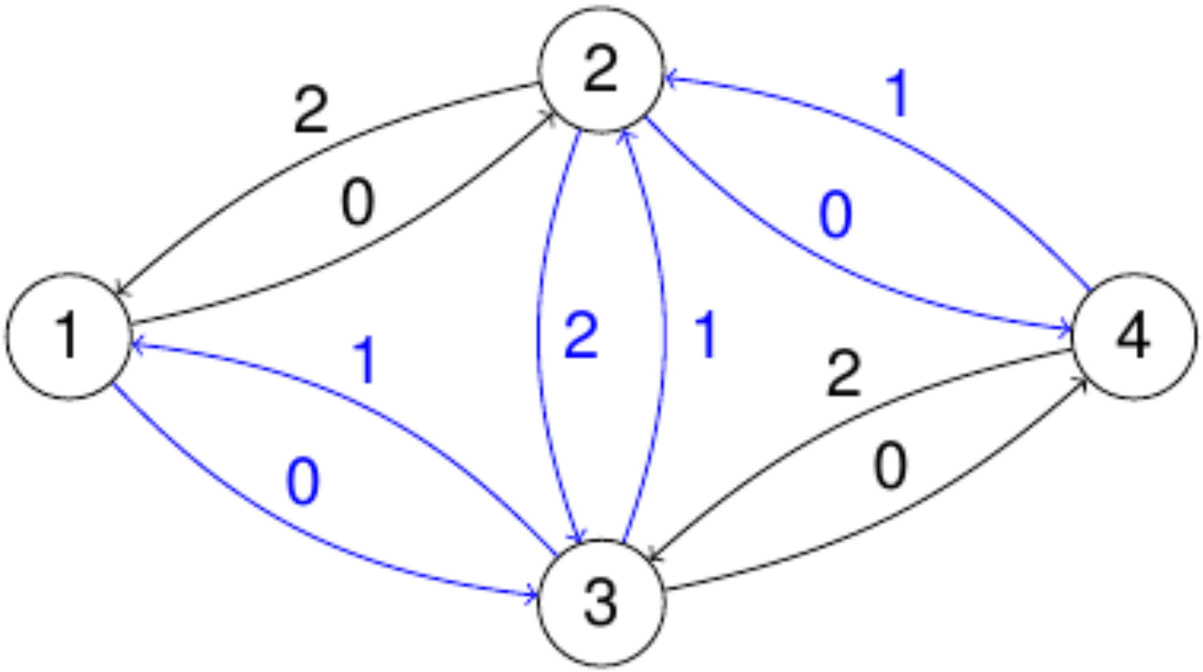


Ford Fulkerson Algorithmus: Aufgabe 3.7 a)



Kapazität Pfad 1 = 2

Ford Fulkerson Algorithmus: Aufgabe 3.7 a)

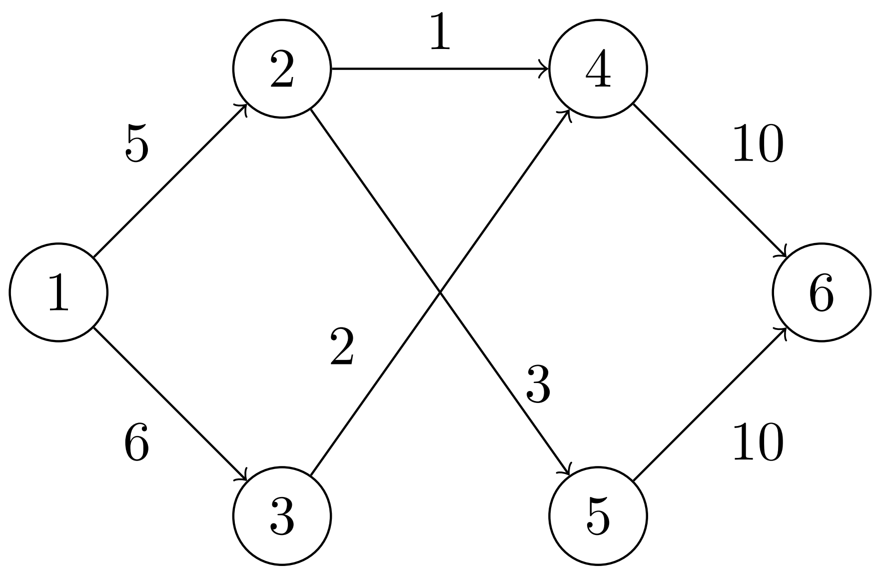


Kapazität Pfad 2 = 1

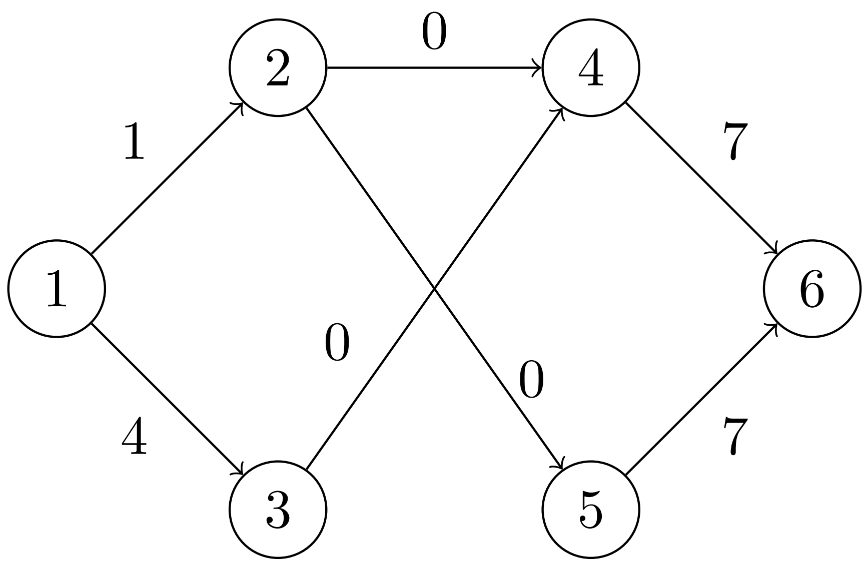
=> Gesamtkapazität = 2 + 1 = 3

Ford Fulkerson Algorithmus: Aufgabe 3.7 b)

Wenden Sie den **Ford Fulkerson Algorithmus** an, um die **Gesamtkapazität** des Netzwerkes zu ermitteln (Quelle in Knoten 1 und Senke in Knoten 6).



Ford Fulkerson Algorithmus: Aufgabe 3.7 b)



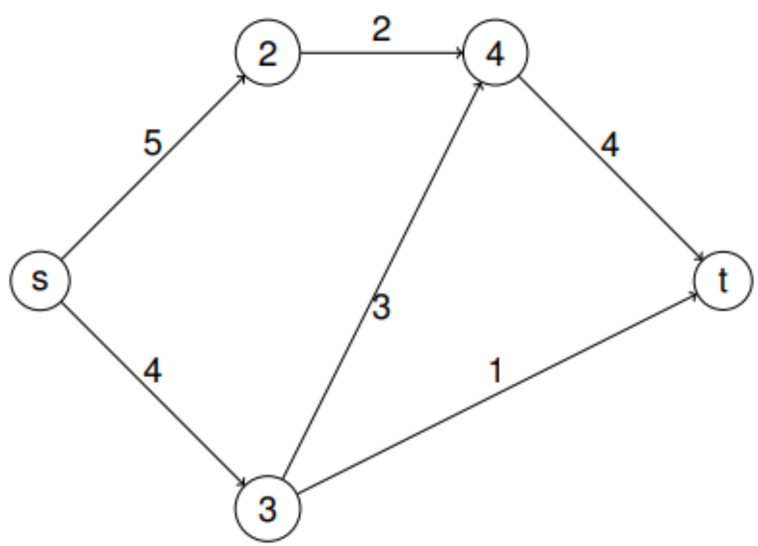
Gesamtkapazität
= 1 (P1) + 3 (P2) + 2 (P3)
= 6

Flussmaximierungsproblem 3.8

Sei V die Menge aller Knoten eines Graphen und i, j, s, t ∈ V Knoten, wobei s die Quelle und t die Senke bezeichnet. Des Weiteren sei g(i, j) ≥ 0 und beschreibt, wie viel von Knoten i nach Knoten j fließt.

a) Nun stellen sie das spezielle Problem für den gegebenen Graphen auf. Die Lösung des Problems soll dabei den maximalen Fluss von **s** nach **t** ergeben.

b) Stellen Sie das Flussmaximierungsproblem allgemein als Optimierungsproblem auf.



Flussmaximierungsproblem 3.8

V: Knoten

E: Kanten

$i,j,s,t \in V$

s: Quelle

t: Senke

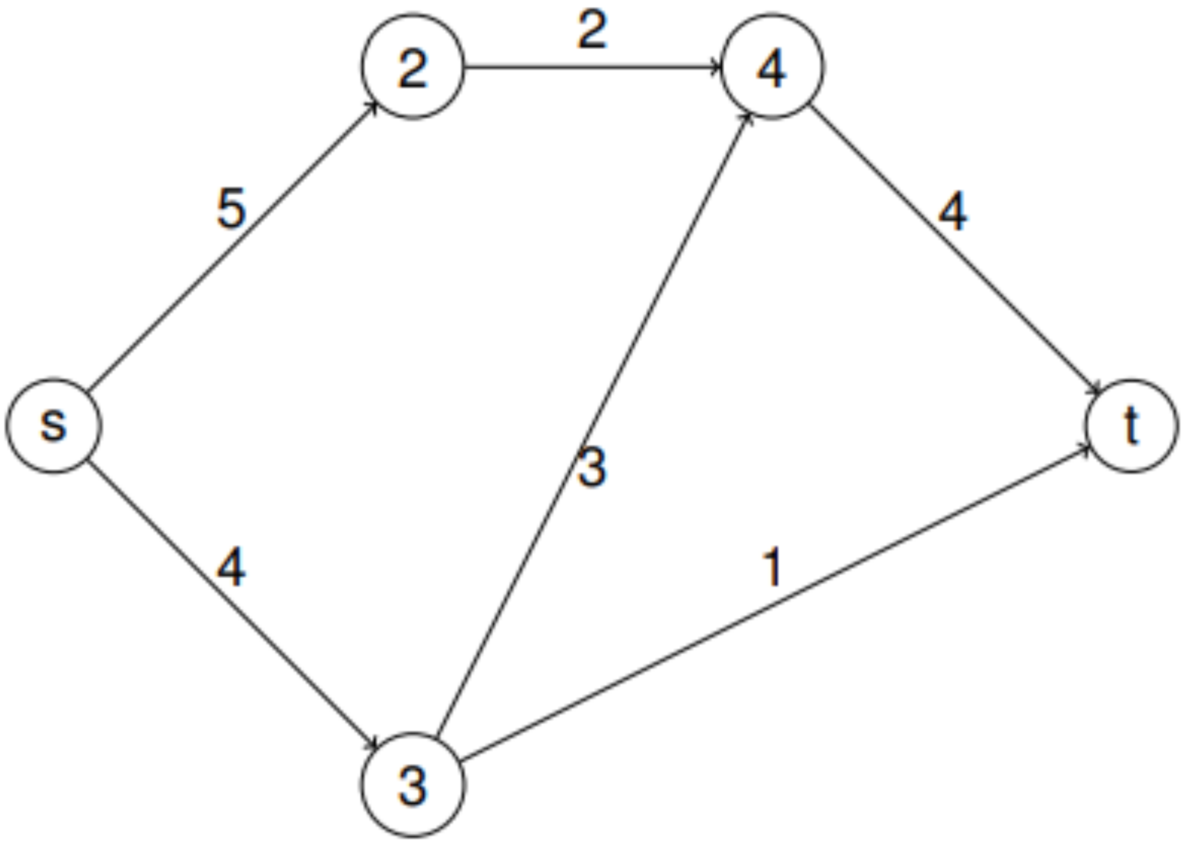
Kapazitätsfunktion

$u(i, j) \geq 0$

Flussfunktion

$g(i, j) \geq 0$

Flussmaximierungsproblem 3.8

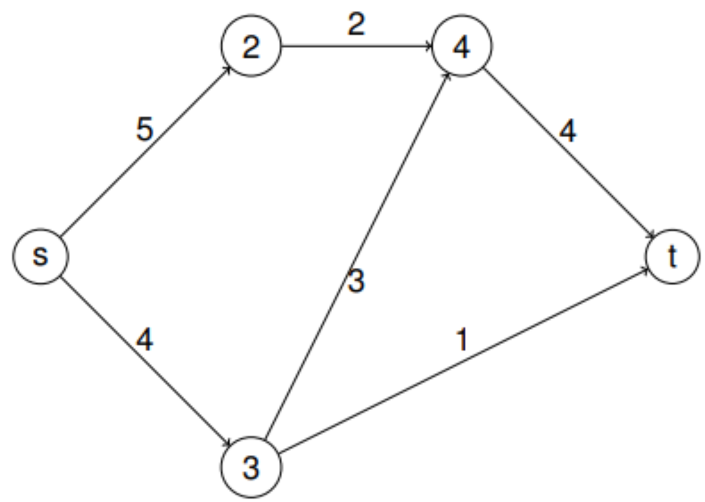


Flussmaximierungsproblem 3.8

Zielfunktion

$$\max z = \sum_{j \in V:(s,j)} g(s,j)$$

$$\max z = g_{s,2} + g_{s,3}$$

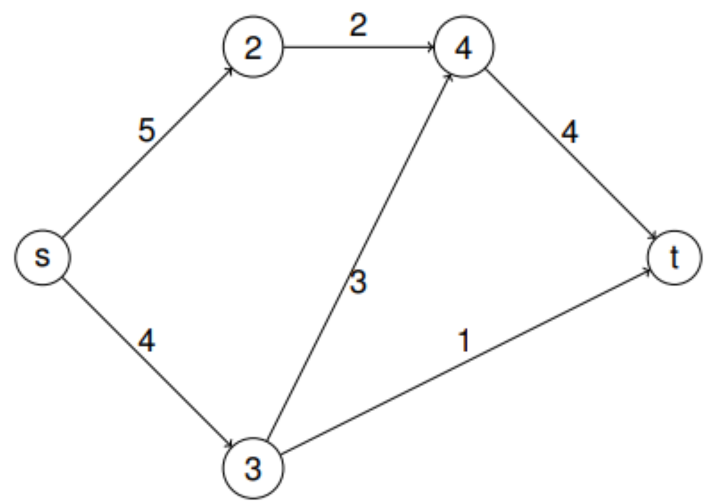


Flussmaximierungsproblem 3.8

Nebenbedingungen – Kapazitätsrestriktionen

$$g(i,j) \leq u(i,j) \quad \forall (i,j) \in E$$

$$
\begin{aligned}
g_{s,2} &\leq 5 \\
g_{s,3} &\leq 4 \\
g_{2,4} &\leq 2 \\
g_{3,4} &\leq 3 \\
g_{3,t} &\leq 1 \\
g_{4,t} &\leq 4
\end{aligned}
$$



Flussmaximierungsproblem 3.8

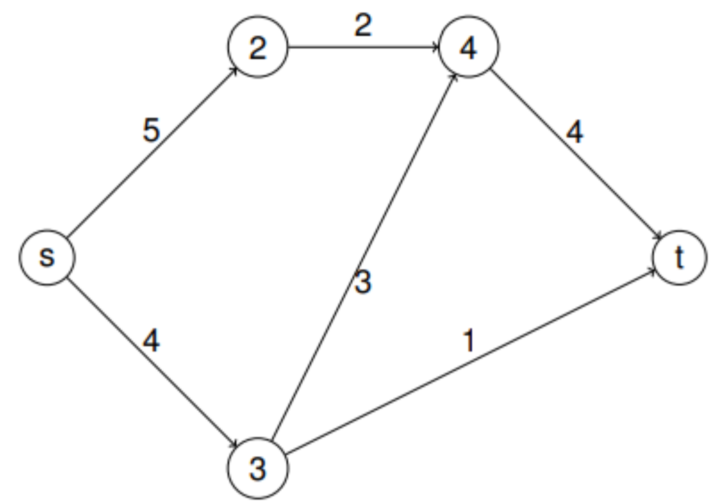
Nebenbedingungen – Massenbilanz

$$\sum_{j \in V:(i,j) \in E} g(i,j) - \sum_{j \in V:(j,i) \in E} g(j,i) = 0 \quad \forall\, i \in V \backslash \{s,t\}$$

$$-g_{s,2} + g_{2,4} = 0$$

$$-g_{s,3} + g_{3,4} + g_{3,t} = 0$$

$$-g_{2,4} - g_{3,4} + g_{4,t} = 0$$



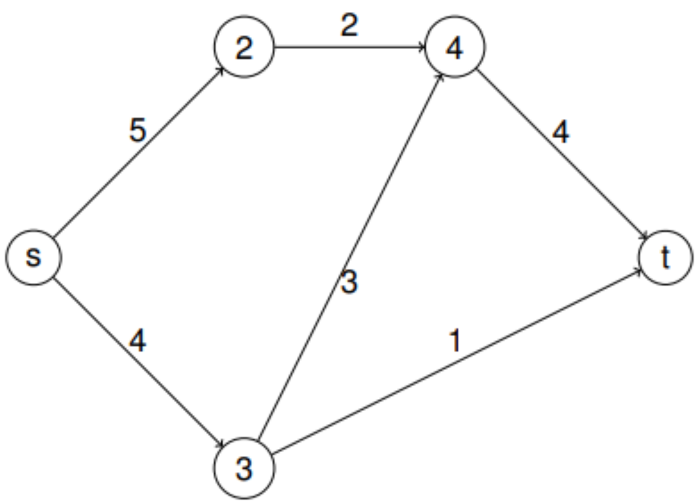
Flussmaximierungsproblem 3.8

$$\max z = g_{s,2} + g_{s,3}$$

s.t.

$$
\begin{aligned}
g_{s,2} &\leq 5 \\
g_{s,3} &\leq 4 \\
g_{2,4} &\leq 2 \\
g_{3,4} &\leq 3 \\
g_{3,t} &\leq 1 \\
g_{4,t} &\leq 4 \\
-g_{s,2} + g_{2,4} &= 0 \\
-g_{s,3} + g_{3,4} + g_{3,t} &= 0 \\
-g_{2,4} - g_{3,4} + g_{4,t} &= 0
\end{aligned}
$$

$$g(i,j) \geq 0 \quad \forall (i,j) \in E$$



Flussmaximierungsproblem 3.8

Zielfunktion

$$\max z = \sum_{j \in V:(s,j)} g(s,j)$$

V: Knoten

E: Kanten

i, j, s, t ∈ V

s: Quelle

t: Senke

- **Maximieren den Fluss aus der Quelle**
- **Alles was aus der Quelle rausfließt muss wieder in die Senke reinfließen!**

Flussmaximierungsproblem 3.8

Nebenbedingungen

V: Knoten

E: Kanten

i, j, s, t ∈ V

s: Quelle

t: Senke

Kapazitätsrestriktionen

$$g(i,j) \leq u(i,j) \quad \forall (i,j) \in E$$

- **Der Fluss entlang einer Kante darf die Kapazität dieser Kante nicht überschreiten!**

Flussmaximierungsproblem 3.8

Nebenbedingungen

V: Knoten

E: Kanten

i, j, s, t ∈ V

s: Quelle

t: Senke

Massenbilanz

$$\sum_{j \in V:(i,j) \in E} g(i,j) - \sum_{j \in V:(j,i) \in E} g(j,i) = 0 \quad \forall\, i \in V \setminus \{s,t\}$$

- **Die Menge die in einem Knoten reinfließt muss gleich der Menge sein die aus diesem Knoten rausfließt!**

Flussmaximierungsproblem 3.8

Definitionsbereich

$$g(i, j) \geq 0$$

V: Knoten

E: Kanten

i, j, s, t ∈ V

s: Startknoten

t: Endknoten

Flussmaximierungsproblem 3.8

Allgemeine Darstellung

V: Knoten

E: Kanten

i, j, s, t ∊ V

s: Startknoten

t: Endknoten

$$
\begin{array}{rrcll}
\max z = & \sum_{j \in V:(s,j)} g(s,j) & & & \\
\text{s.t.} & g(i,j) & \leq & u(i,j) & \forall (i,j) \in E \\
& \sum_{j \in V:(i,j) \in E} g(i,j) - \sum_{j \in V:(j,i) \in E} g(j,i) & = & 0 & \forall i \in V \setminus \{s,t\} \\
& g(i,j) & \geq & 0 & \forall (i,j) \in E
\end{array}
$$

Modellierung mit Max-Flow 3.9

9 Freunde aus Berlin möchten einen Ausflug nach Dresden planen. Der Zug von Berlin nach Dresden fährt 3 Mal pro Tag, um 6:00, 8:00 und 10:00 Uhr. Viele wollen noch eine anstrengende Arbeitswoche ausschlafen, deswegen kommen die ersten beiden Züge für 4 von den Freunden nicht in Frage. Da der Ausflug sehr kurzfristig geplant wird, sind die meisten Fahrkarten leider schon ausverkauft. Die Züge um 6:00 und 8:00 Uhr haben jeweils noch einen Platz frei, während der Zug um 10:00 Uhr nur noch 4 frei hat. Da die Gruppe sehr motiviert ist, den Ausflug zu organisieren, wird auch überlegt mit dem Auto zu fahren, dafür sind aber nur 3 Personen bereit. Das Auto kann maximal 5 Personen inklusive Fahrer aufnehmen.

Ist es möglich den Ausflug zu organisieren, so dass alle Wünsche und Anforderungen erfüllt werden?

Hinweis: Alle die nicht ausschlafen möchten, wollen auch nicht mit dem Auto fahren.

Modellierung mit Max-Flow 3.9

9位来自柏林的朋友想要计划一次去德累斯顿的郊游。柏林到德累斯顿的火车每天都有3班，分别是早上6点、8点和10点。由于许多人想要在紧张的工作周之后好好睡个懒觉，所以前两班火车对于其中4位朋友来说不合适。由于这次郊游是临时计划的，大多数车票都已经售罄。早上6点和8点的火车各还有一个空位，而10点的火车只剩下4个空位。考虑到团队非常有动力组织这次郊游，他们也在考虑开车前往，但只有3人愿意开车。汽车最多能搭载包括司机在内的5人。
是否可能组织这次郊游，以满足所有的愿望和要求？ => 是的，如果最大流量≥9。

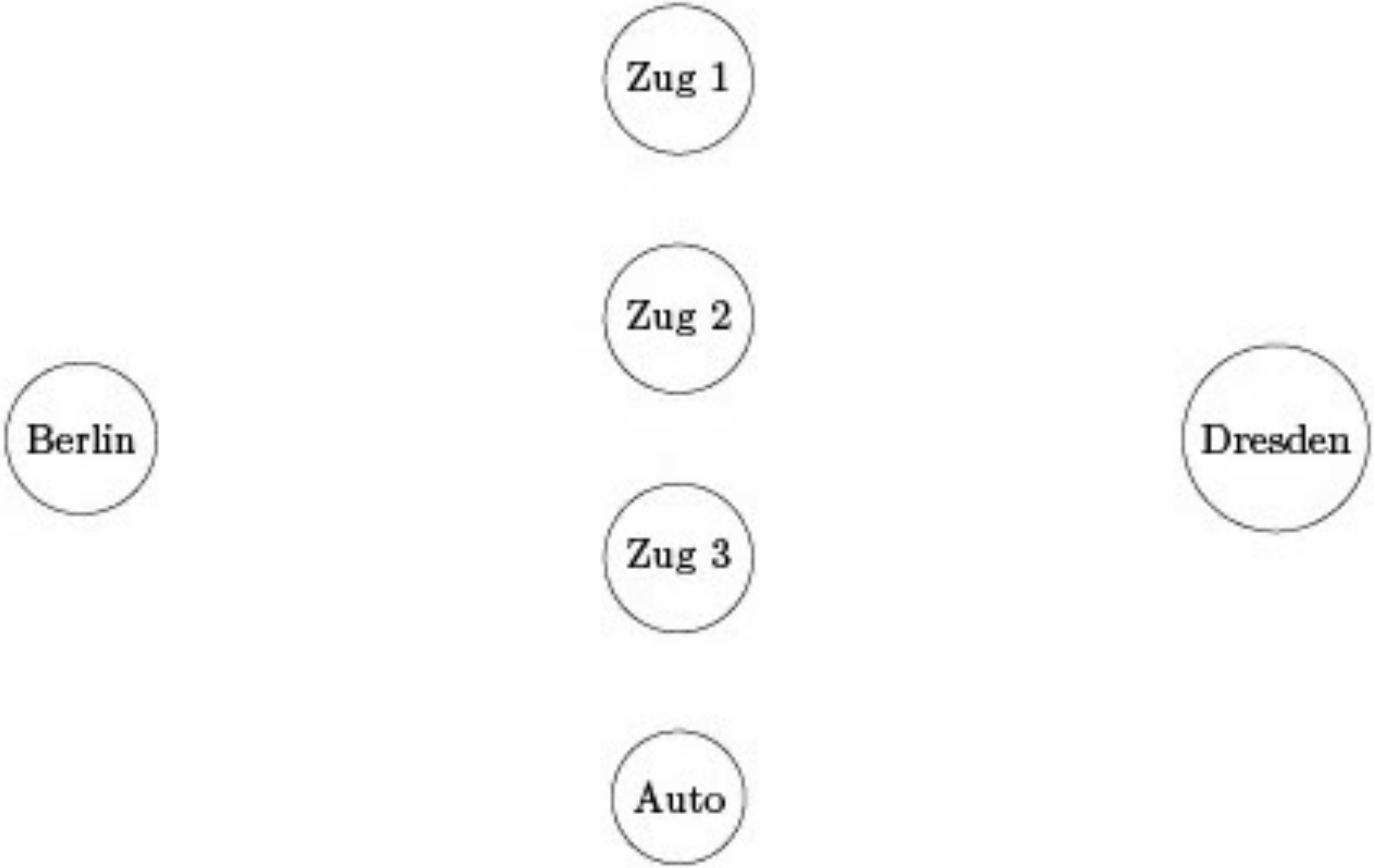
提示：所有不想睡懒觉的人也不想坐车。

9 Freunde aus Berlin möchten einen Ausflug nach Dresden planen. Der Zug von Berlin nach Dresden fährt 3 Mal pro Tag, um 6:00, 8:00 und 10:00 Uhr. Viele wollen noch eine anstrengende Arbeitswoche ausschlafen, deswegen kommen die ersten beiden Züge für 4 von den Freunden nicht in Frage. Da der Ausflug sehr kurzfristig geplant wird, sind die meisten Fahrkarten leider schon ausverkauft. Die Züge um 6:00 und 8:00 Uhr haben jeweils noch einen Platz frei, während der Zug um 10:00 Uhr nur noch 4 frei hat. Da die Gruppe sehr motiviert ist, den Ausflug zu organisieren, wird auch überlegt mit dem Auto zu fahren, dafür sind aber nur 3 Personen bereit. Das Auto kann maximal 5 Personen inklusive Fahrer aufnehmen.

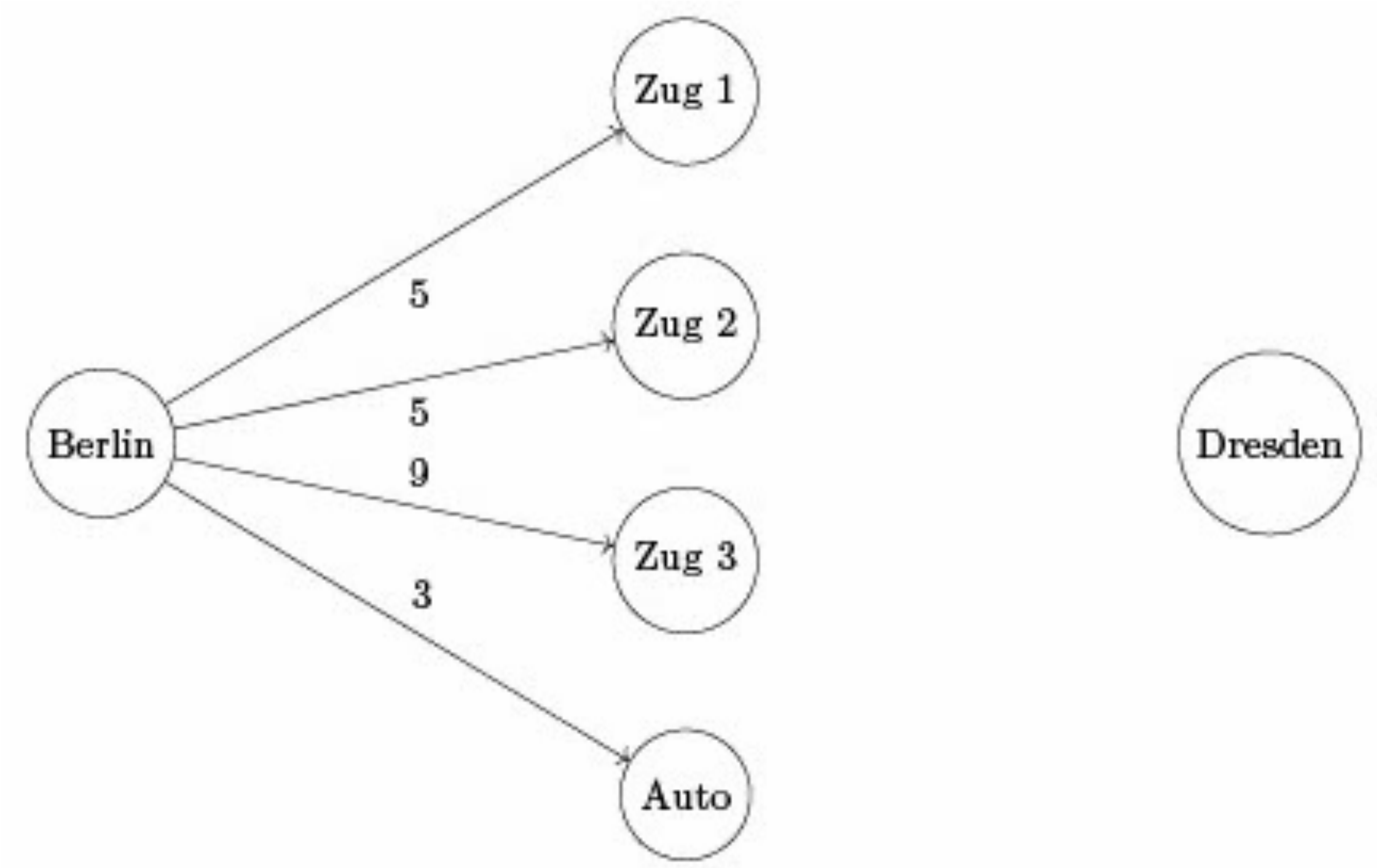
Ist es möglich den Ausflug zu organisieren, so dass alle Wünsche und Anforderungen erfüllt werden? **=> Ja, falls Maximum Flow $\geq$ 9**

Hinweis: Alle die nicht ausschlafen möchten, wollen auch nicht mit dem Auto fahren.

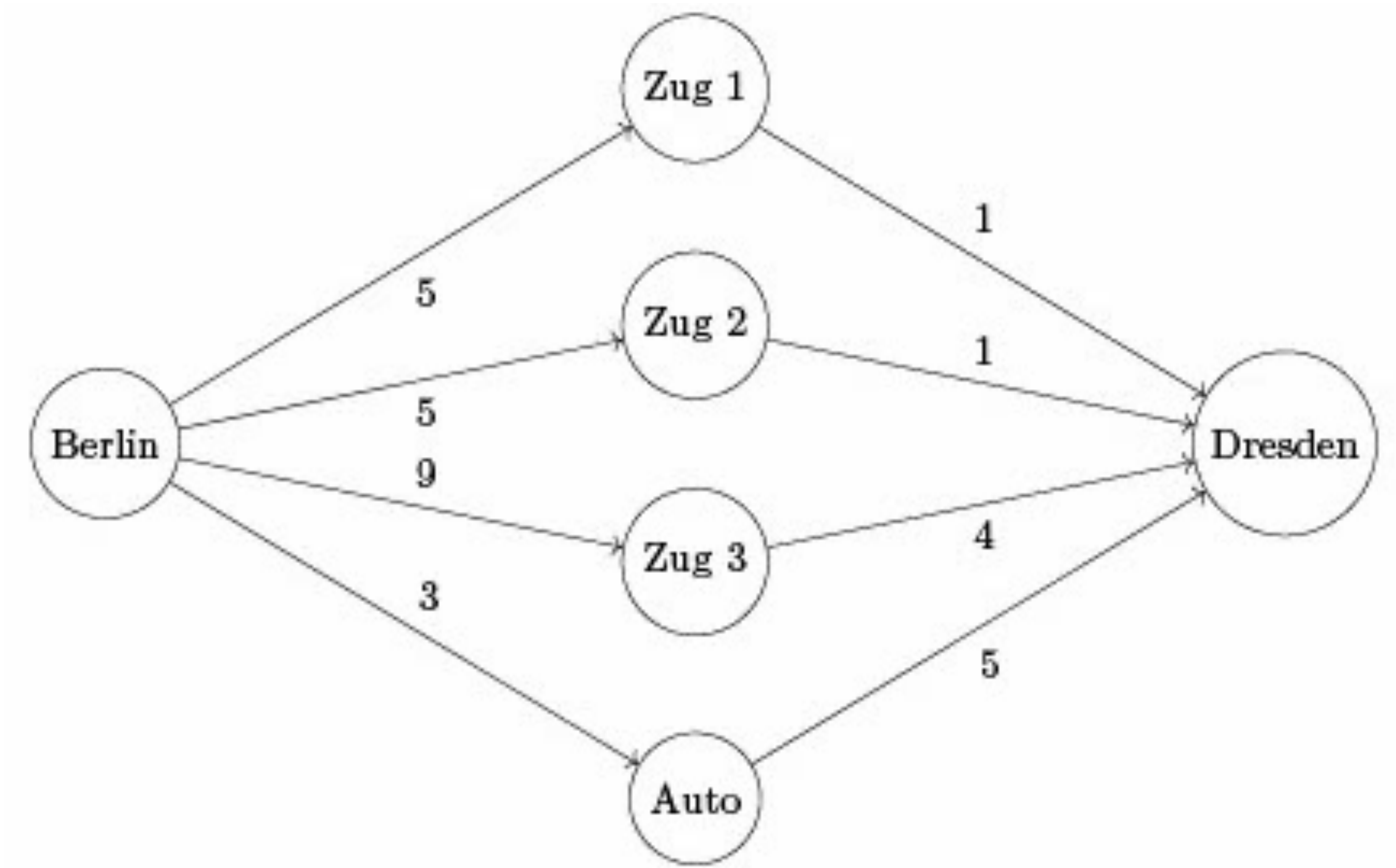
Modellierung mit Max-Flow 3.9



Moldellierung mit Max-Flow 3.9



Modellierung mit Max-Flow 3.9



Fragen zum Tutorium?