Networks and flows
Ford-Fulkerson algorithm

Let $G = (V, E)$ be a directed graph. For vertex $v \in V$ we can define its *indegree (sisendaste)* $\overrightarrow{\deg}(v)$ and *outdegree (väljundaste)* $\overleftarrow{\deg}(v)$.

If $\overrightarrow{\deg}(v) = 0$ or $\overleftarrow{\deg}(v) = 0$, the vertex v is called *source (lähe)* or *sink (suue)* of graph G, respectively.

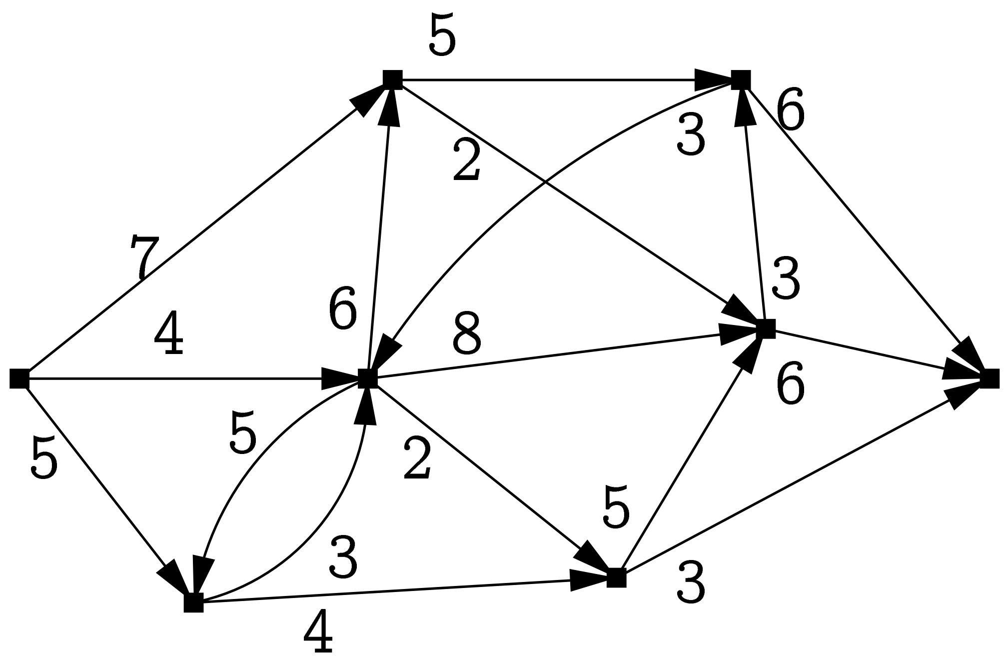
Capacity of graph G is a function $\psi : E \longrightarrow \mathbb{R}_+$.

The quantities

$$\overrightarrow{\deg}_\psi(v) = \sum_{e \in E \, \mathcal{E}(e)=(u,v)} \psi(e) \text{ and } \overleftarrow{\deg}_\psi(v) = \sum_{e \in E \, \mathcal{E}(e)=(v,u)} \psi(e)$$

are called *ψ-indegree* and *ψ-outdegree* of vertex $v \in V$, respectively.

Network (võrk) is a pair (G, ψ) where G is a directed graph and ψ its capacity.


5
6
3
2
7
3
6
4
8
5
5
2
6
5
3
3
4

Proposition. The sums of all ψ-indegrees and ψ-outdegrees of graph G are equal.

Proof.

$$\sum_{v\in V}\overrightarrow{\deg_\psi}(v)=\sum_{v\in V}\sum_{\substack{e\in E\\ \mathcal{E}(e)=(u,v)}}\psi(e)=\sum_{e\in E}\psi(e)=$$
$$\sum_{v\in V}\sum_{\substack{e\in E\\ \mathcal{E}(e)=(v,u)}}\psi(e)=\sum_{v\in V}\overleftarrow{\deg_\psi}(v)\ .$$

□

Let (G, ψ) be a network. We assume that G has exactly one source s and exactly one sink t.

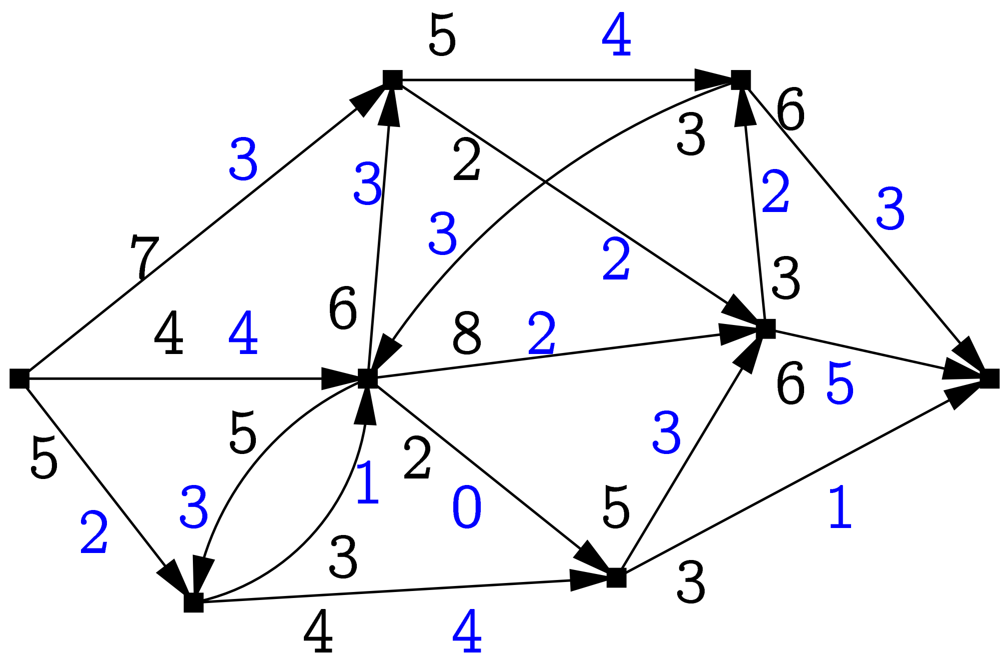
Flow (voog) on the network (G, ψ) is a function $\varphi : E \longrightarrow \mathbb{R}_+$, such that

- $\varphi(e) \leq \psi(e)$ for every $e \in E$.
- $\overrightarrow{\deg_\varphi}(v) = \overleftarrow{\deg_\varphi}(v)$ for every $v \in V \backslash \{s, t\}$.

The previous proposition implies $\overleftarrow{\deg_\varphi}(s) = \overrightarrow{\deg_\varphi}(t)$. This quantity is called *value (väärtus)* of the flow φ and denoted $|\varphi|$.

The flow is *maximal* if its value is the largest possible.

We assume that $G = (V, E)$ has no loops nor multiple directed arcs, i.e. $E \subseteq V \times V$.



5
4
6
3
3
2
3
2
3
2
3
7
6
3
4
4
8
2
6
5
5
5
2
1
3
0
5
2
3
1
3
4
4

Proposition. Consider a network (G, ψ) with $G = (V, E)$. Let $V = V_s \dot{\cup} V_t$, such that $s \in V_s$ and $t \in V_t$. Let

$$\Phi(V_s, V_t) = \sum_{e \in E \cap (V_s \times V_t)} \varphi(e) - \sum_{e \in E \cap (V_t \times V_s)} \varphi(e) \ .$$

Then $\Phi(V_s, V_t)$ is equal to the value of φ.

Proof. Induction over $|V_s|$.

Base. If $|V_s| = 1$ then $V_s = \{s\}$. The set $V_s \times V_t$ contains all the arcs originating from s and $V_t \times V_s = \emptyset$.

Step. Let the claim hold for some sets V_s and V_t. Let $x \in V_t \backslash \{t\}$, $V_s' = V_s \cup \{x\}$ and $V_t' = V_t \backslash \{x\}$. It is enough to prove $\Phi(V_s, V_t) = \Phi(V_s', V_t')$.

$\Phi(V_s, V_t)$:

$V \times V$	V_s	x	V_t'
V_s		$+\varphi$	$+\varphi$
x	$-\varphi$		
V_t'	$-\varphi$		

$\Phi(V_s', V_t')$:

$V \times V$	V_s	x	V_t'
V_s			$+\varphi$
x			$+\varphi$
V_t'	$-\varphi$	$-\varphi$	

$$\Phi(V_s, V_t) - \Phi(V_s', V_t') =$$

$V \times V$	V_s	x	V_t'
V_s		$+\varphi$	
x	$-\varphi$		$-\varphi$
V_t'		$+\varphi$	

$$= \overrightarrow{\deg}_{\varphi}(x) - \overleftarrow{\deg}_{\varphi}(x) = 0 \quad . \qquad \square$$

A *cut (lõige)* in the network (G, ψ) (where $G = (V, E)$) is such an arc set $L \subseteq E$ that every directed path from source to sink uses some arc from the set L.

Alternatively: $L \subseteq E$ is a cut, if there are no directed paths from s to t in the graph $(V, E \backslash L)$.

Capacity (läbisaskevõime) of L is the quantity $\psi(L) = \sum_{e \in L} \psi(e)$.

The cut is *minimal* if its capacity is the smallest possible.

Theorem (Ford and Fulkerson). The value of all maximal flows in a network is equal to the capacity of all the minimal cuts.

Proof. Let (G, ψ) be a network with $G = (V, E)$, source s and sink t. We will show that

I. The value of no flow is larger than the capacity of any cut.

II. For any maximal flow φ there exists a cut with capacity $|\varphi|$.

Part I Let φ be a flow and L a cut.

Let $V_s \subseteq V$ be the set of such nodes v that there exists a directed path from s to v without using any arc from L. Let $V_t = V \backslash V_s$. Since $E \cap (V_s \times V_t) \subseteq L$, we have

$$\psi(L) \geq \sum_{e \in E \cap (V_s \times V_t)} \psi(e) \geq \sum_{e \in E \cap (V_s \times V_t)} \varphi(e) \geq \Phi(V_s, V_t) = |\varphi| \ .$$

Part II Let φ be a maximal flow.

Let $V_s \subseteq V$ be the set of all vertices v such that:
There exists an *undirected* path $s = v_0 \overset{e_1}{\text{———}} v_1 \overset{e_2}{\text{———}} \cdots \overset{e_m}{\text{———}} v_m = v$, such that

- If $e_i = (v_{i-1}, v_i)$ then $\varphi(e_i) < \psi(e_i)$.
- If $e_i = (v_i, v_{i-1})$ then $\varphi(e_i) > 0$.

We say that the flow between v_{i-1} and v_i is *unsaturated (küllastamata)*.

Such a path is called *augmenting (suurendav)*.

Let $V_t = V \backslash V_s$. We will show that $t \in V_t$. Indeed, if $t \in V_s$ then φ is not maximal:

Let $s = v_0 \xrightarrow{e_1} v_1 \xrightarrow{e_2} \cdots \xrightarrow{e_m} v_m = t$ be some augmenting path. Define positive real numbers δ_i as follows:

$$\delta_i = \begin{cases} \psi(e_i) - \varphi(e_i), & \text{if } e_i = (v_{i-1}, v_i) \\ \varphi(e_i), & \text{if } e_i = (v_i, v_{i-1}) \end{cases} .$$

Let $\varepsilon = \min_i \delta_i$ and let φ' be the following flow:

$$\varphi'(e) = \begin{cases} \varphi(e), & \text{if } e \notin \{e_1, \ldots, e_m\} \\ \varphi(e) + \varepsilon, & \text{if } e = e_i = (v_{i-1}, v_i) \\ \varphi(e) - \varepsilon, & \text{if } e = e_i = (v_i, v_{i-1}) \end{cases} .$$

Then φ' is a flow and $|\varphi'| = |\varphi| + \varepsilon$.

Construction of the sets V_s and V_t gives:

- If $e \in E \cap (V_s \times V_t)$ then $\varphi(e) = \psi(e)$.
- If $e \in E \cap (V_t \times V_s)$ then $\varphi(e) = 0$.

Let $L = E \cap (V_s \times V_t)$. Then L is a cut and $\psi(L) = |\varphi|$. $\square$

Algorithm for finding a maximal flow (Ford-Fulkerson).
Let (G, ψ) be a network with $G = (V, E)$.

Let φ be some initial flow on the network (G, ψ), say, $\forall e :$ $\varphi(e) = 0$.

Repeat:

1. Find an augmenting path $s = v_0 \overset{e_1}{\text{——}} v_1 \overset{e_2}{\text{——}} \cdots \overset{e_m}{\text{——}} v_m = t$.
   If there is no such path then stop and output φ.
2. Construct φ' as described 2 slides ago.
3. Assign $\varphi := \varphi'$.

The augmenting path is found traversing the graph in some manner.

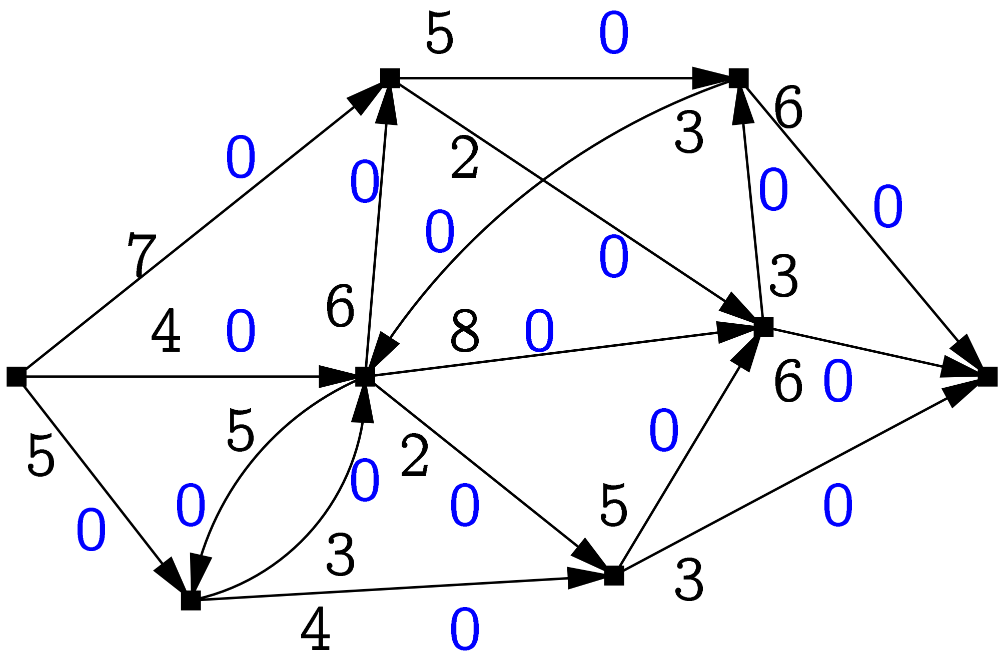
Theorem. Ford-Fulkerson algorithm finds a maximal flow.

Proof. The algorithm obviously outputs a flow. We need to prove that it does not stop before a maximal flow is found.

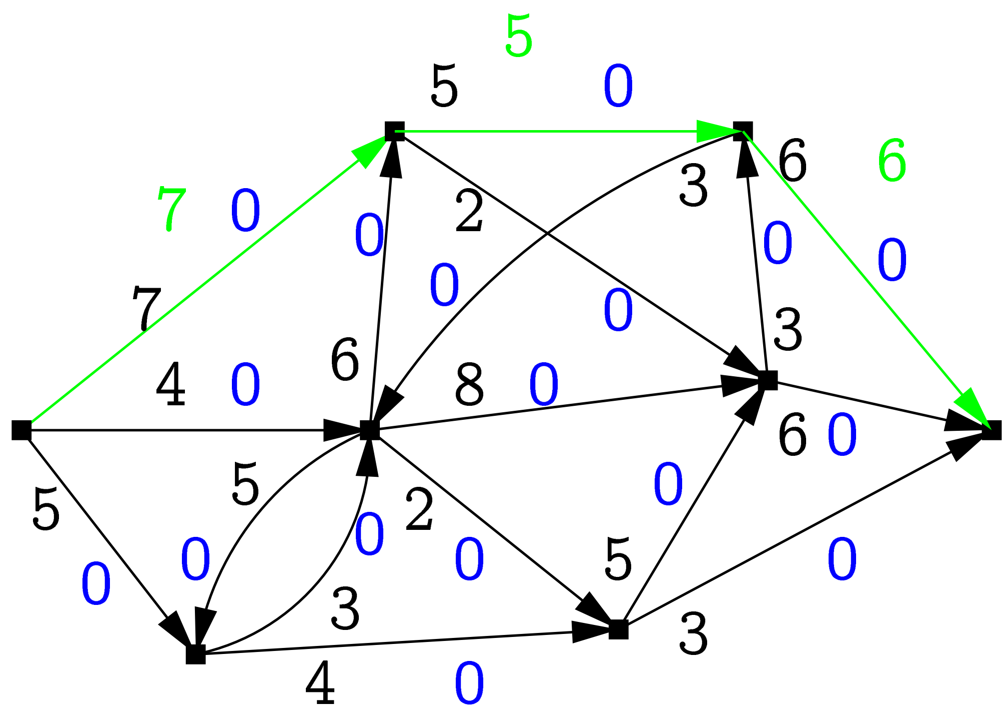
We will show that if φ is not a maximal flow then there exists an augmenting path $s \rightsquigarrow t$ for it.

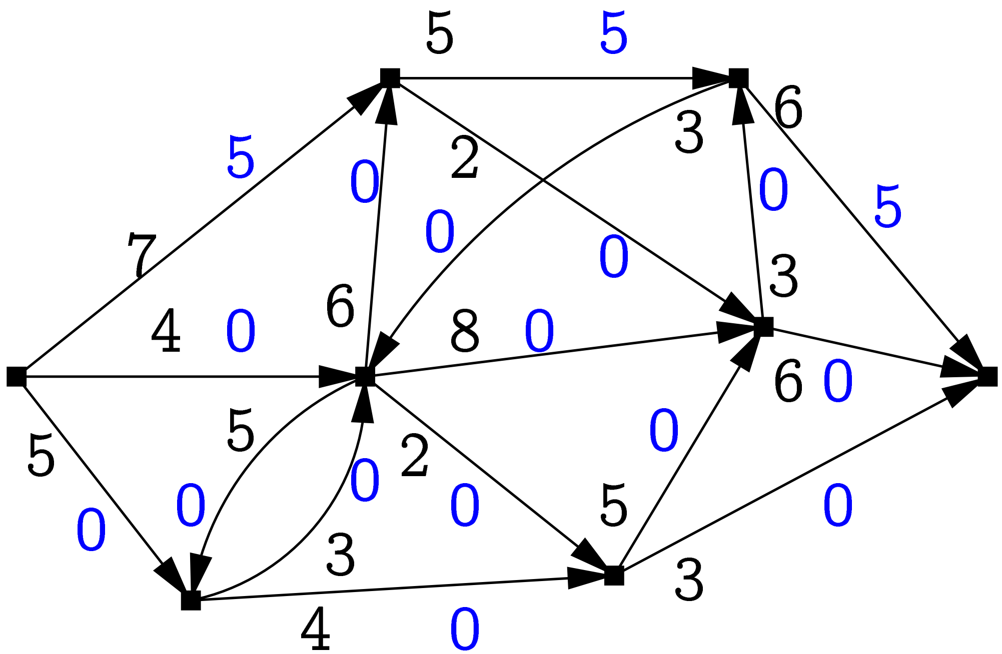
Let V_s be the set of vertices v such that there exists an augmenting path from s to v and let $V_t = V \setminus V_s$. Assume that $t \in V_t$.

Similarly to the proof of the previous theorem we get that $L = E \cap (V_s \times V_t)$ is a cut and $\psi(L) = |\varphi|$. Thus φ must be maximal. $\square$

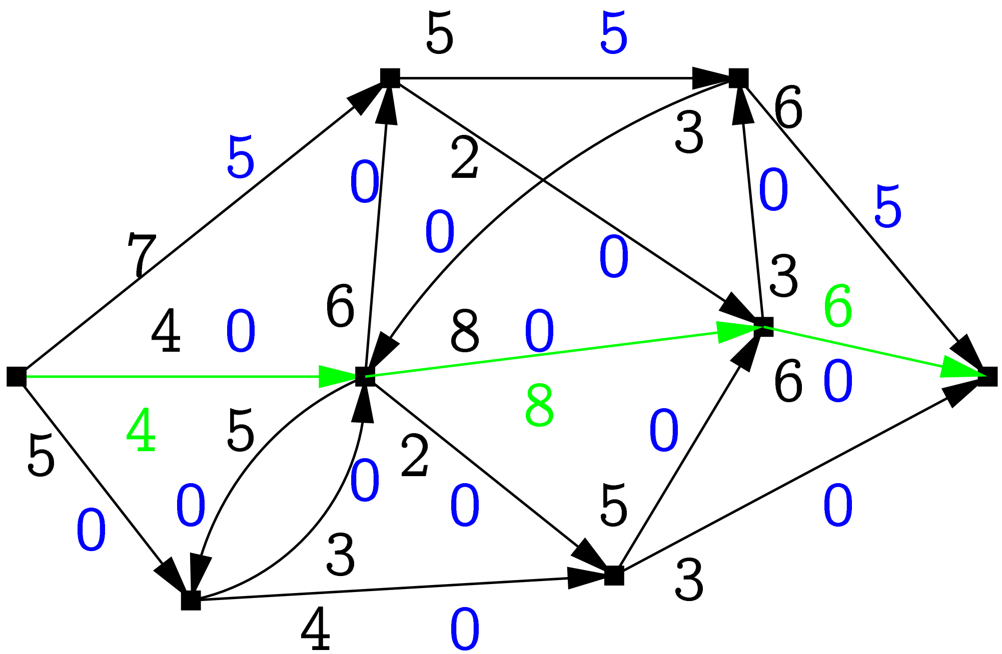


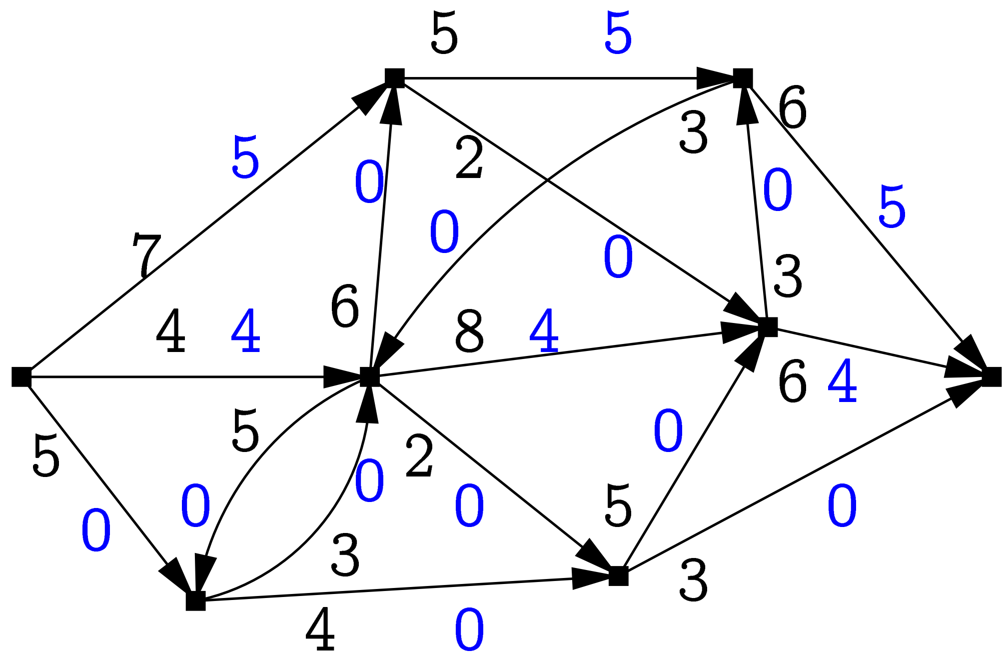
5
0
0
6
7
0
3
2
0
0
0
0
3
6
4
0
8
0
5
6
0
5
2
0
0
0
0
5
0
3
3
4
0

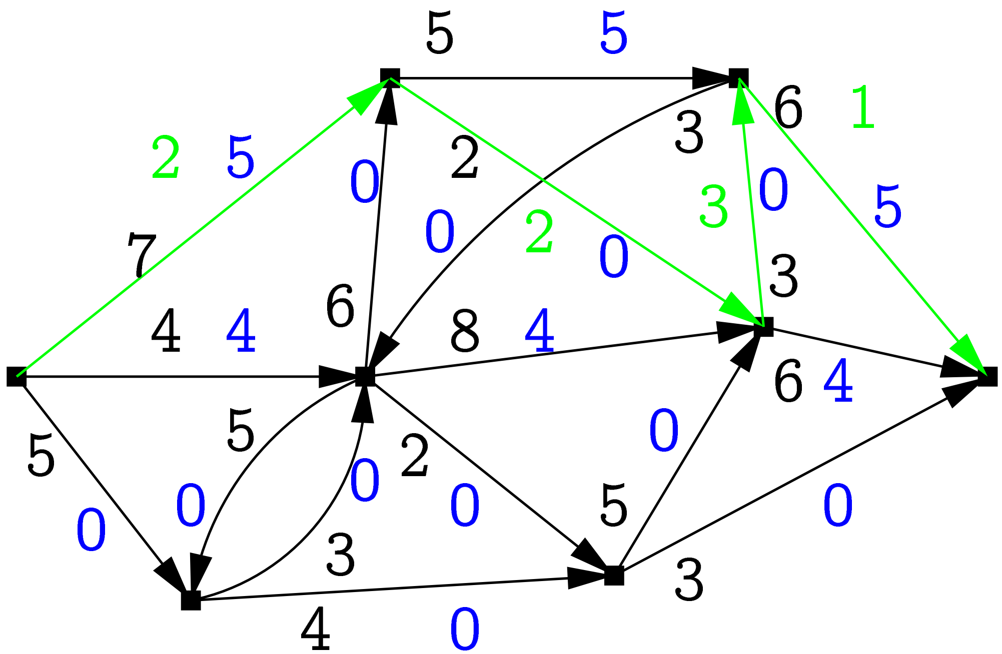

5
5
0
6
6
7
0
0
2
3
0
0
0
7
3
0
6
4
0
8
0
6
0
5
0
5
2
0
0
5
0
3
3
4
0

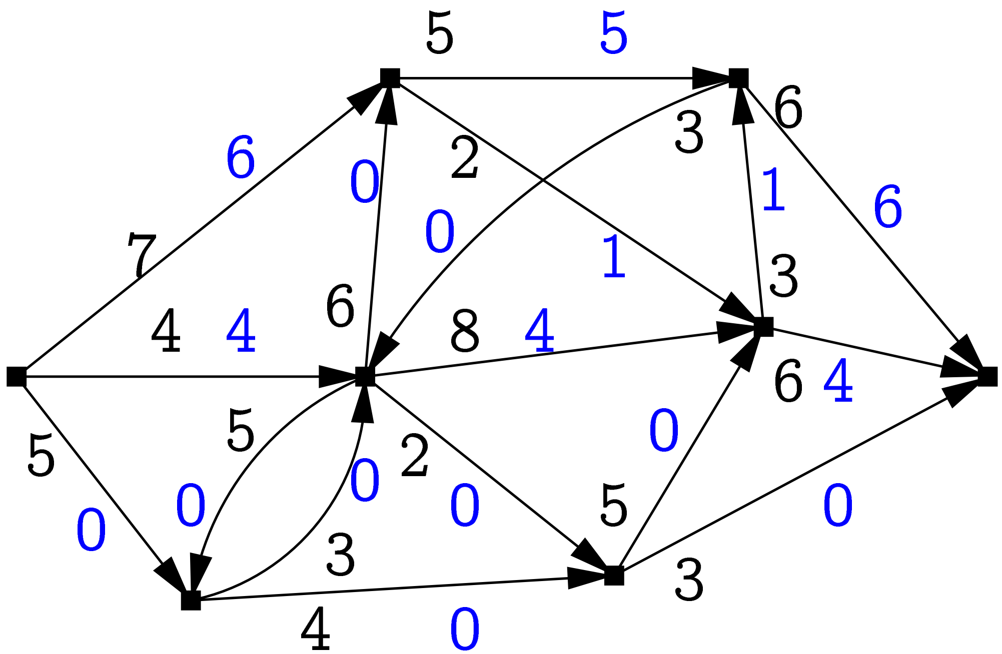


5
5
6
3
2
5
7
0
0
0
5
0
4
0
6
8
0
3
5
6
0
5
0
2
0
0
0
5
0
3
3
4
0

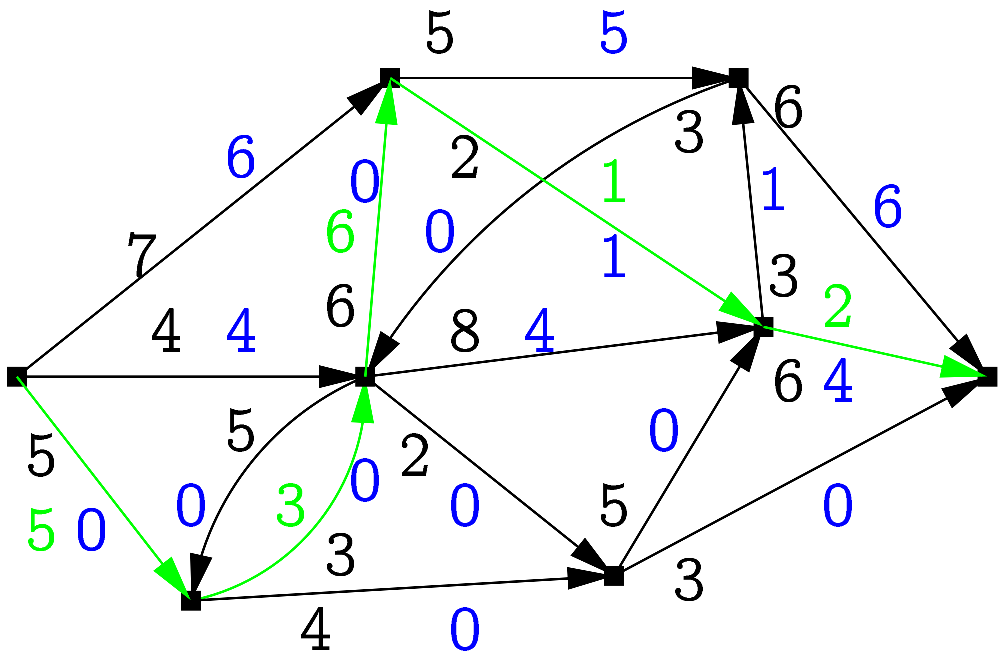

5
5
6
3
2
5
0
0
7
0
0
3
4
0
6
8
0
6
6
0
5
4
5
2
0
0
0
5
0
0
3
3
4
0

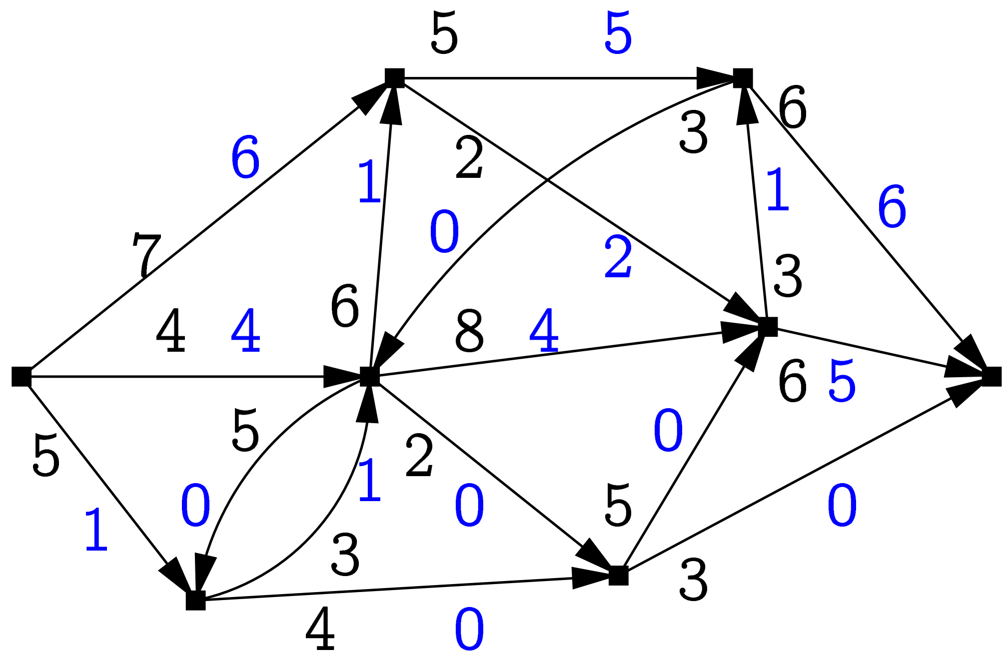

5
5
6
3
2
5
0
0
0
0
5
7
3
6
4
4
8
4
6
4
5
0
5
2
0
0
0
0
5
3
3
4
0

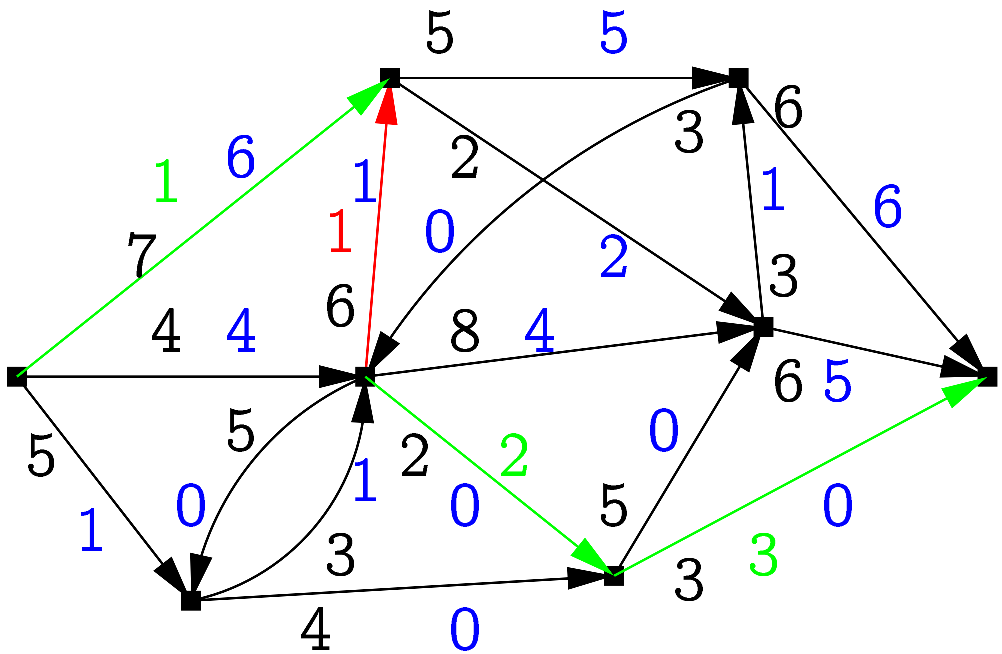

5
5
6
1
2
5
3
2
0
0
3
0
5
7
2
0
3
4
4
6
8
4
6
4
5
0
5
2
0
0
0
5
0
3
3
4
0

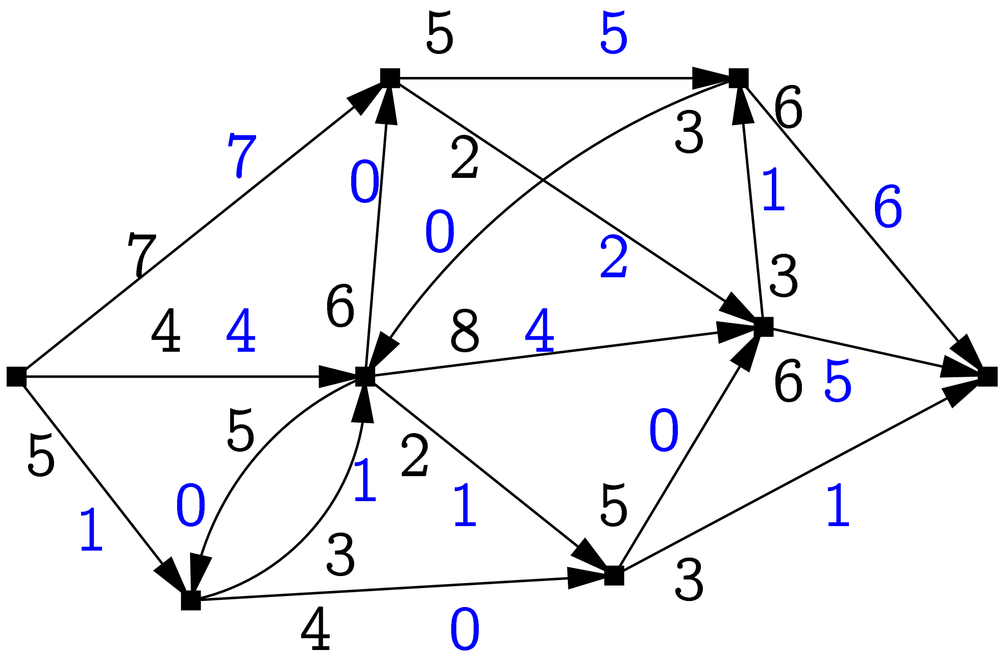


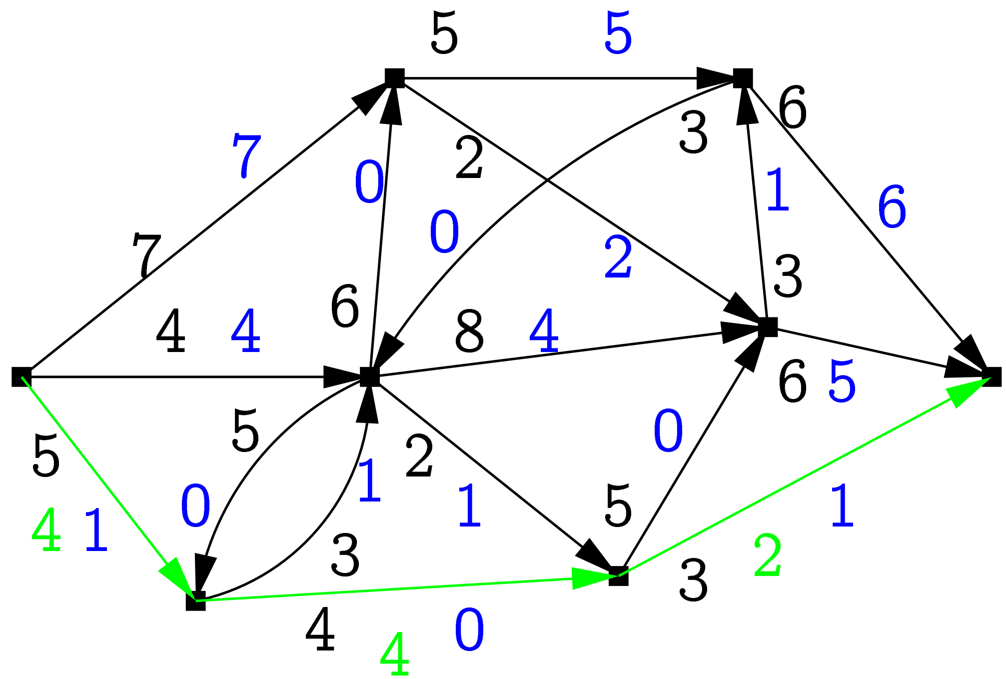
5
5
6
3
2
6
0
7
0
1
6
1
3
4
4
6
8
4
5
6
4
5
0
2
0
0
0
0
5
3
3
4
0

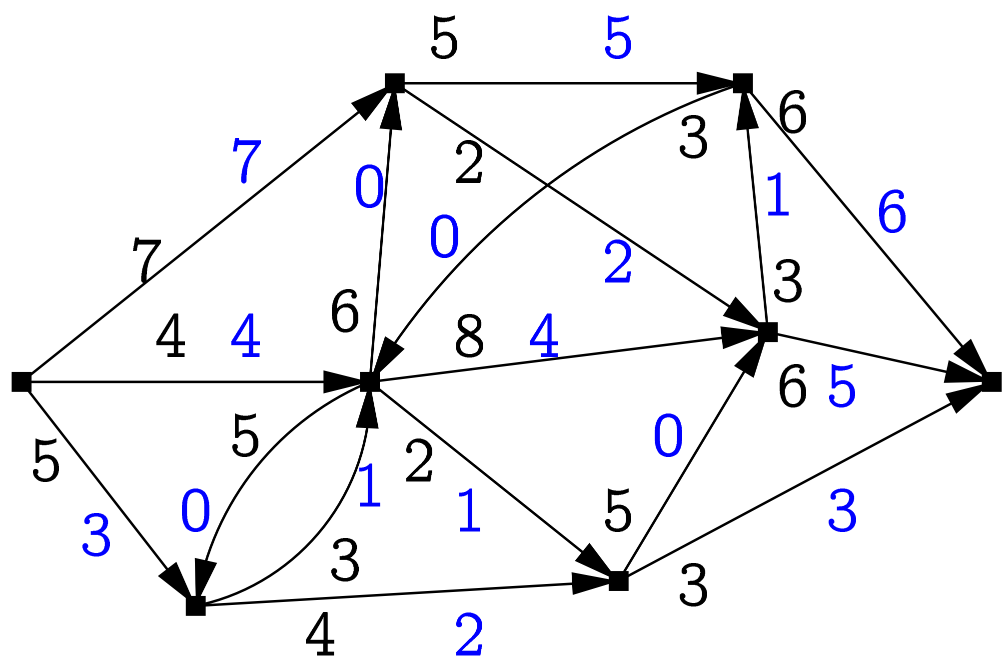

5
5
6
3
2
0
1
6
1
7
0
6
1
3
6
2
4
4
8
4
6
6
4
5
5
0
2
0
3
0
5
0
0
3
3
4
0


5
5
6
3
2
6
1
1
6
7
0
2
3
6
4
4
8
4
6
5
5
5
0
2
0
1
0
5
1
3
0
3
4
0

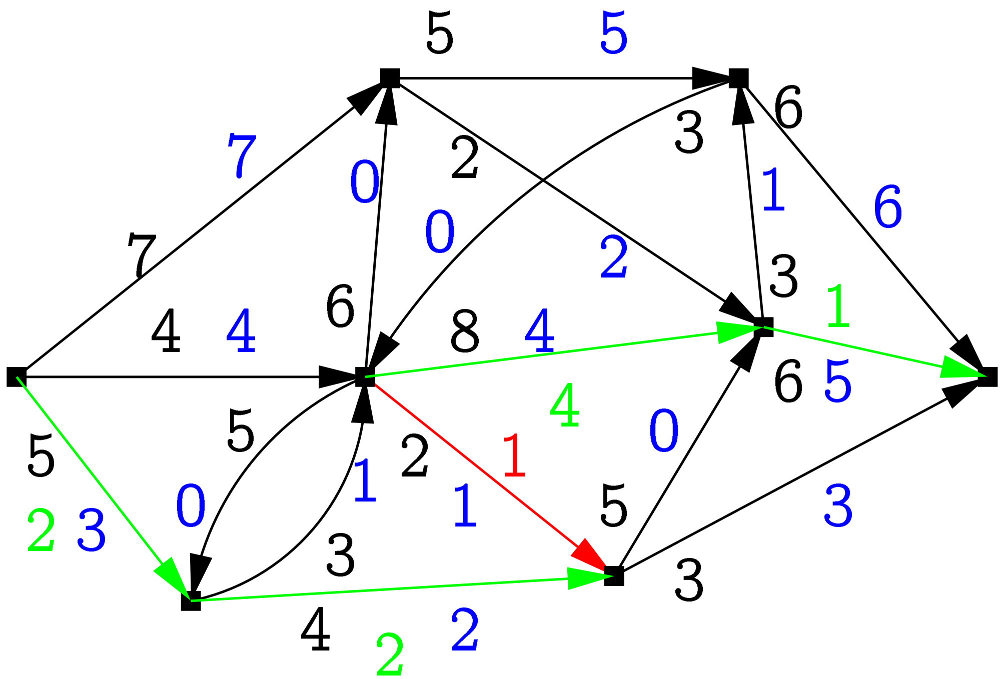

5
5
6
3
1
6
2
6
1
1
0
7
2
1
3
4
4
6
8
4
6
5
5
5
0
2
2
1
0
5
0
1
0
3
3
3
3
4
0


5
5
6
3
7
2
0
1
6
0
7
2
3
6
4
4
8
4
6
5
5
2
0
5
0
1
1
5
1
3
3
4
0

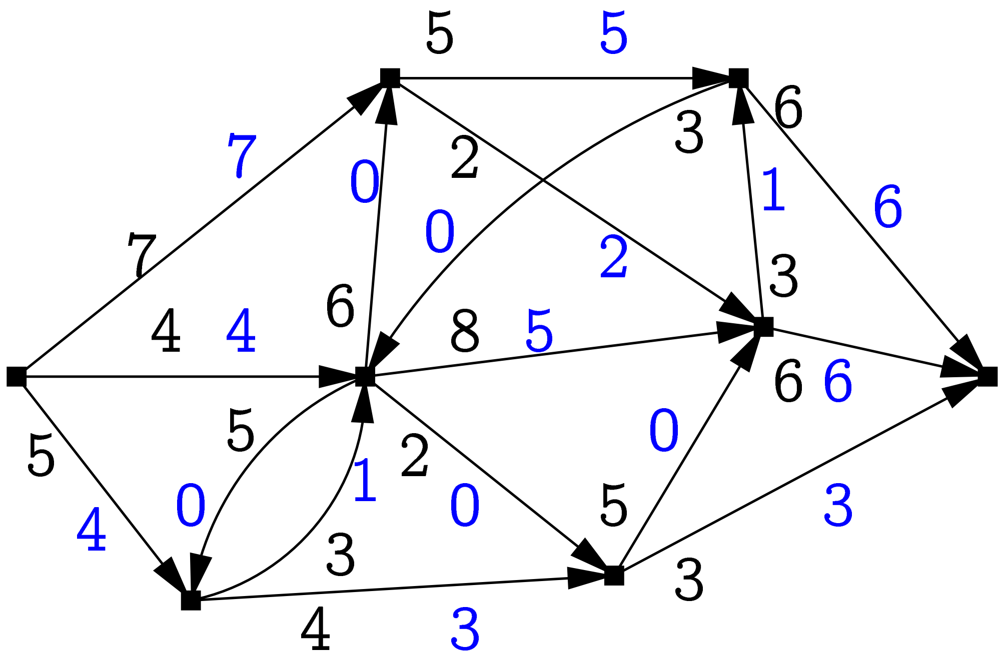

5
5
7
7
0
6
3
2
0
1
6
2
3
4
4
6
8
4
6
5
5
5
2
0
4
1
0
1
1
3
5
2
1
3
4
4
0



5
5
6
3
2
7
0
0
1
6
7
2
3
6
4
4
8
4
6
5
5
5
0
2
1
0
1
5
3
3
3
4
2
3

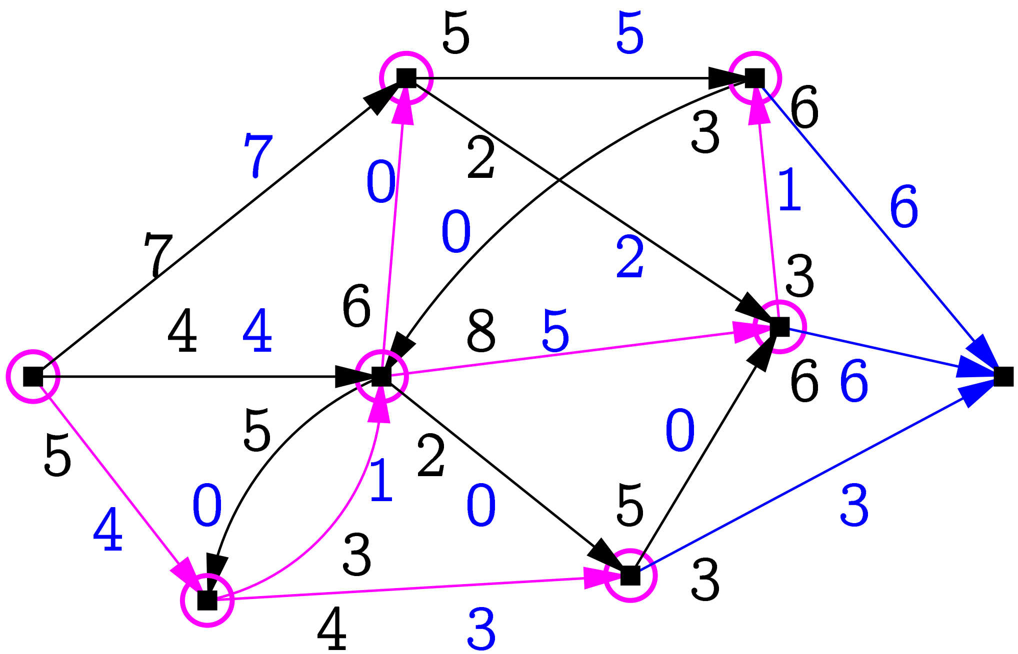


5
5
6
3
2
7
0
0
1
6
7
2
3
1
6
4
4
8
4
6
5
4
5
0
5
2
1
1
5
2
3
0
3
3
3
4
2
2



5
5
7
6
0
3
2
1
6
0
7
2
3
6
4
4
8
5
6
6
5
5
2
1
0
0
4
3
0
5
3
4
3
3

neisse tippudesse leidub suurendav tee



minimaalne lõige: ringiga $\rightarrow$ ringita

Finding the augmenting path:

Let $V_s = \{s\}$, $W = \{s\}$.

While $W \neq \emptyset$ and $t \notin V_s$ do:

1. Somehow choose $v \in W$. Remove it from the set W.
2. For each $e \in E$ and vertex v incident with it: if the flow between v and e's other endpoint w is unsaturated and $w \notin V_s$ then
   (a) Add w to sets V_s and W.
   (b) Remember that v is the vertex "preceding" w.

If $t \notin V_s$ then there is no augmenting path. If $t \in V_s$ one can construct an augmenting path moving from t by "preceding" vertices to s.

Proposition. If capacities of all the edges are integers then the main cycle of the algorithm is run at most $|\varphi|$ times where φ is a maximal flow.

Proof. Each iteration increases the value of the flow. Since our computations do not introduce non-integers, each increase has to be at least by 1. □

We will now assume that the augmenting path is found using breadth-first traversal of the graph (Edmonds-Karp algorithm).

The augmenting path $s = v_0 \xrightarrow{e_1} v_1 \xrightarrow{e_2} \cdots \xrightarrow{e_m} v_m = t$ found will have the following property:
For each i, the path $s = v_0 \xrightarrow{e_1} v_1 \xrightarrow{e_2} \cdots \xrightarrow{e_i} v_i$ is the shortest augmenting path from source to v_i.

Let (G, ψ) be a network with $G = (V, E)$ and let φ be a flow on it. Denote the length of the shortest path from source to $v \in V$ as $\delta_\varphi(v)$.

Proposition. Let $\varphi_0, \varphi_1, \varphi_2, \ldots$ be the sequence of flows generated during the maximal flow finding algorithm. Then for each $v \in V$ the sequence $\delta_{\varphi_i}(v)$ is non-decreasing.

Proof. Consider the flows φ_n and φ_{n+1} in this sequence and let $B = \{v \mid \delta_{\varphi_{n+1}}(v) < \delta_{\varphi_n}(v)\}$. Assume that B is not empty and let $v \in B$ be such that $\delta_{\varphi_{n+1}}(v)$ is the smallest possible.

Let P' be the shortest augmenting path from source to v w.r.t the flow φ_{n+1}. Let u be the vertex preceding v on this path. Since $\delta_{\varphi_{n+1}}(u) < \delta_{\varphi_{n+1}}(v)$, we have $u \notin B$.

Consider the flow φ_n between the vertices u and v.

If φ_n is unsaturated between the vertices u and v then

$$\delta_{\varphi_n}(v) \leq \delta_{\varphi_n}(u) + 1 \leq \delta_{\varphi_{n+1}}(u) + 1 = \delta_{\varphi_{n+1}}(v)$$

and $v \notin B$, a contradiction.

If φ_n is saturated between the vertices u and v then let P_n be the augmenting path from source to sink that was used to generate φ_{n+1} from φ_n.

In φ_{n+1}, the flow between u and v becomes unsaturated. Thus, in the path P_n there exists an edge $\cdots - v - u - \cdots$. According to the properties of P_n we get $\delta_{\varphi_n}(v) = \delta_{\varphi_n}(u) - 1$. Consequently,

$$\delta_{\varphi_n}(v) = \delta_{\varphi_n}(u) - 1 \leq \delta_{\varphi_{n+1}}(u) - 1 = \delta_{\varphi_{n+1}}(v) - 2 < \delta_{\varphi_{n+1}}(v)$$

and $v \notin B$, a contradiction. $\square$

Theorem. Edmonds-Karp algorithm makes at most $(|V|-2) \cdot |E|$ iterations.

Proof. Consider the nth iteration of the algorithm. On this iteration, the augmenting path $P_n : s = v_0 \overset{e_1}{\text{———}} v_1 \overset{e_2}{\text{———}} \cdots \overset{e_m}{\text{———}} v_m = t$ is constructed. Call the pair of vertices (v_{i-1}, v_i) *critical* if the respective quantity δ_i (showing how much the flow between v_{i-1} and v_i must be changed to make it saturated) is minimal (i.e. $\delta_i = \varepsilon$).

Each iteration has a critical pair of vertices. On the next iteration it becomes saturated.

Let's count the number of iterations where a pair (u, v) can be critical. If it is critical on the nth iteration, we have $\delta_{\varphi_n}(v) = \delta_{\varphi_n}(u) + 1$.

To make (u, v) again critical on iteration number $n' > n$, there must exist another augmenting path $P_{n'}$ containing the arc $\cdots — v — u — \cdots$. Then

$$\delta_{\varphi_{n'}}(u) = \delta_{\varphi_{n'}}(v) + 1 \geq \delta_{\varphi_n}(v) + 1 = \delta_{\varphi_n}(u) + 2,$$

thus every time when (u, v) is critical, $\delta_\varphi(u)$ has increased at least by 2.

The quantity $\delta_\varphi(u)$ can not exeed $|V| - 2$ (when (u, v) is critical). Thus (u, v) is critical at most $\frac{|V|-2}{2}$ times. The number of vertex pairs (u, v) is at most $2 \cdot |E|$. $\square$