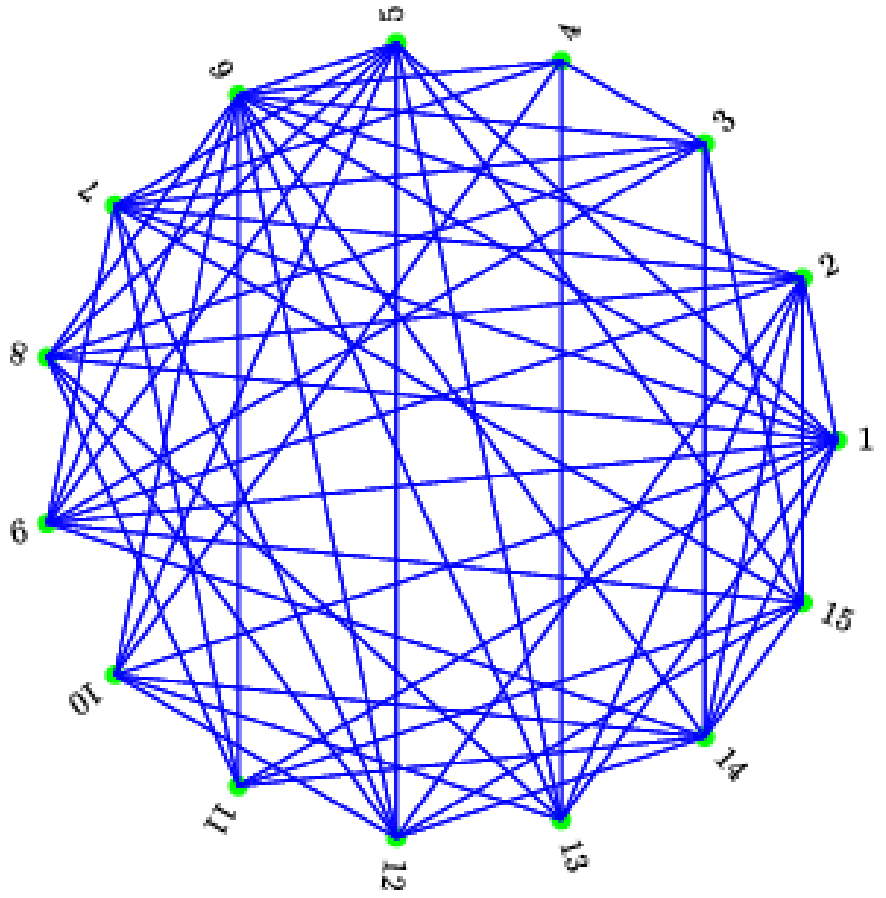
Dijkstra

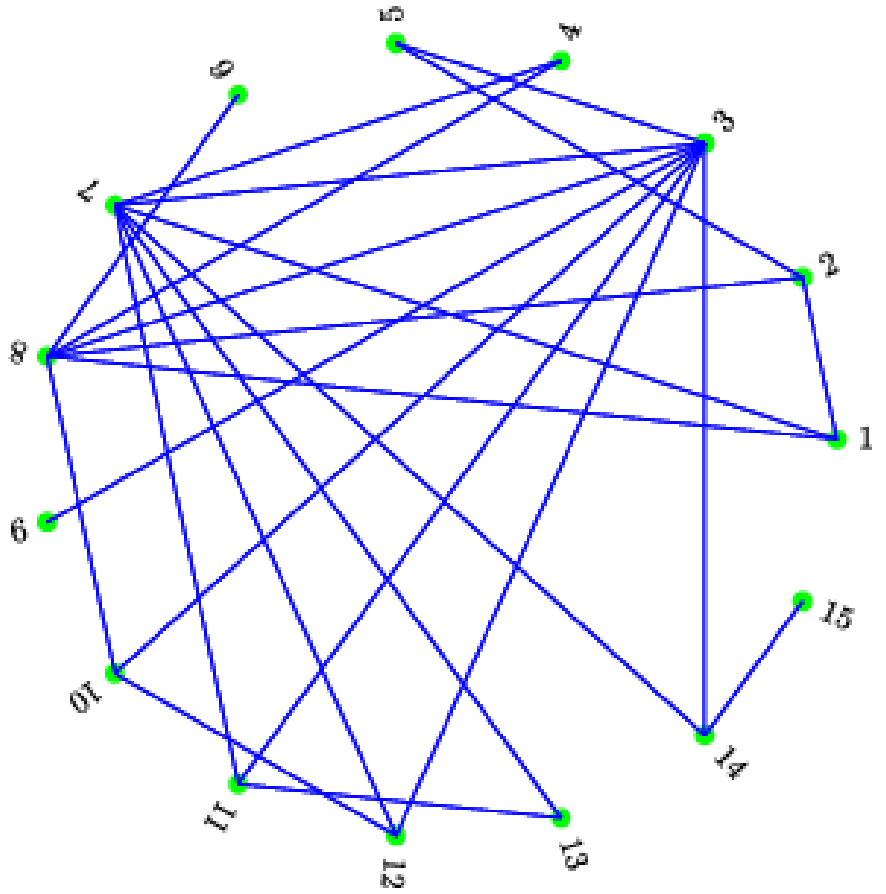
Взвешенный кратчайший путь

Граф	Веса	Алгоритм	Сложность O(·)
Любой	Unweighted	BFS	|V | + |E|
DAG	Any	DAG Relaxation	|V | + |E|
Любой	Any	Bellman-Ford	|V | · |E|
Любой	Non-negative	Dijkstra	|V | log |V | + |E|

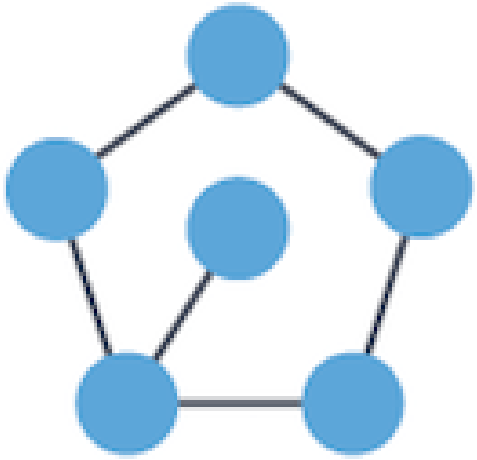
Взвешенные графы. Dense graph

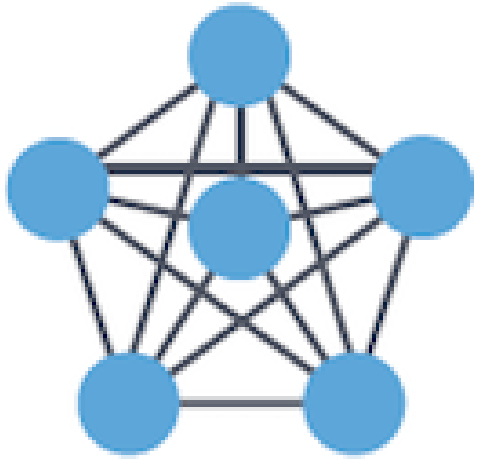


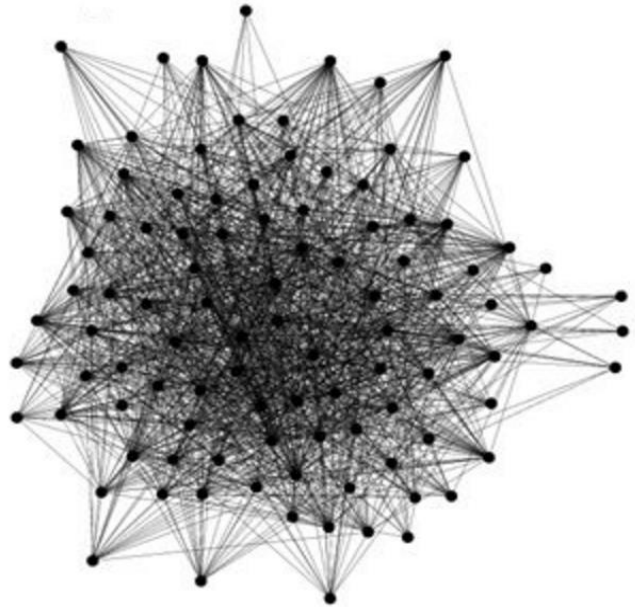
Взвешенные графы. Sparse graph

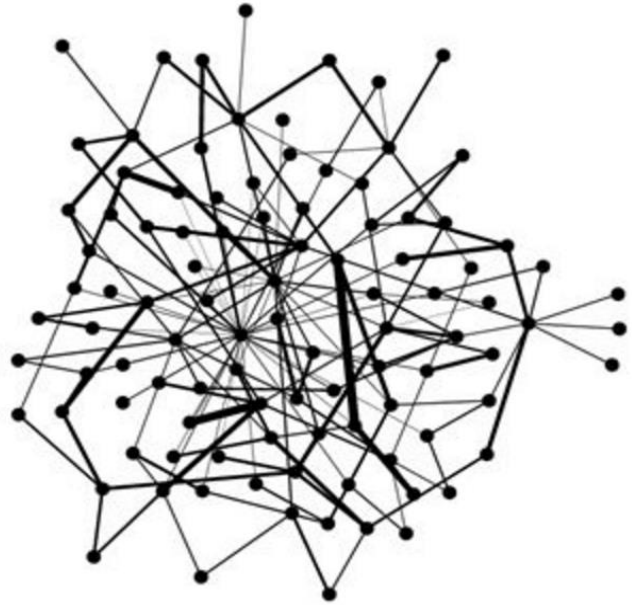


Взвешенные графы. Sparse vs Dense graph









Взвешенные графы. Dijkstra

Дорога с отрицательной длиной?

Взвешенные графы. Dijkstra

Взвешенные графы. Dijkstra

~~Дорога с отрицательной длиной?~~

Взвешенные графы. Dijkstra

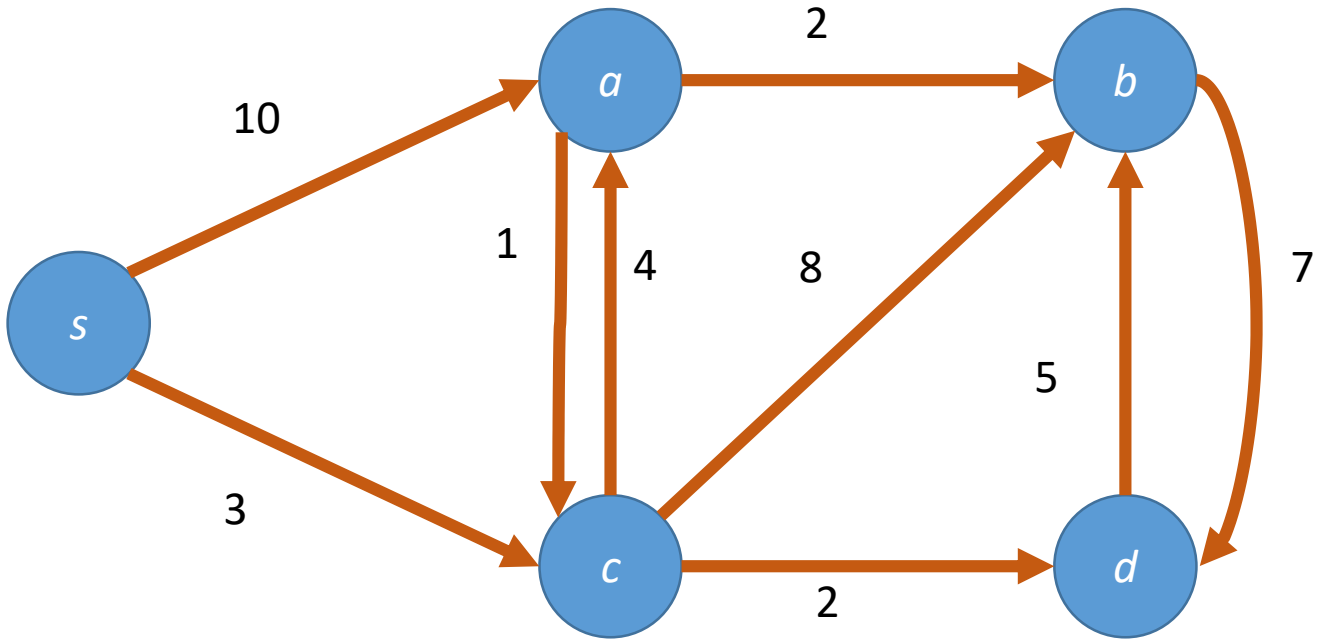
Идея! Обобщить подход BFS для взвешенных графов:

- Вырастить «сферу» с центром в s;
- Циклично обходить вершины начиная с ближайших;

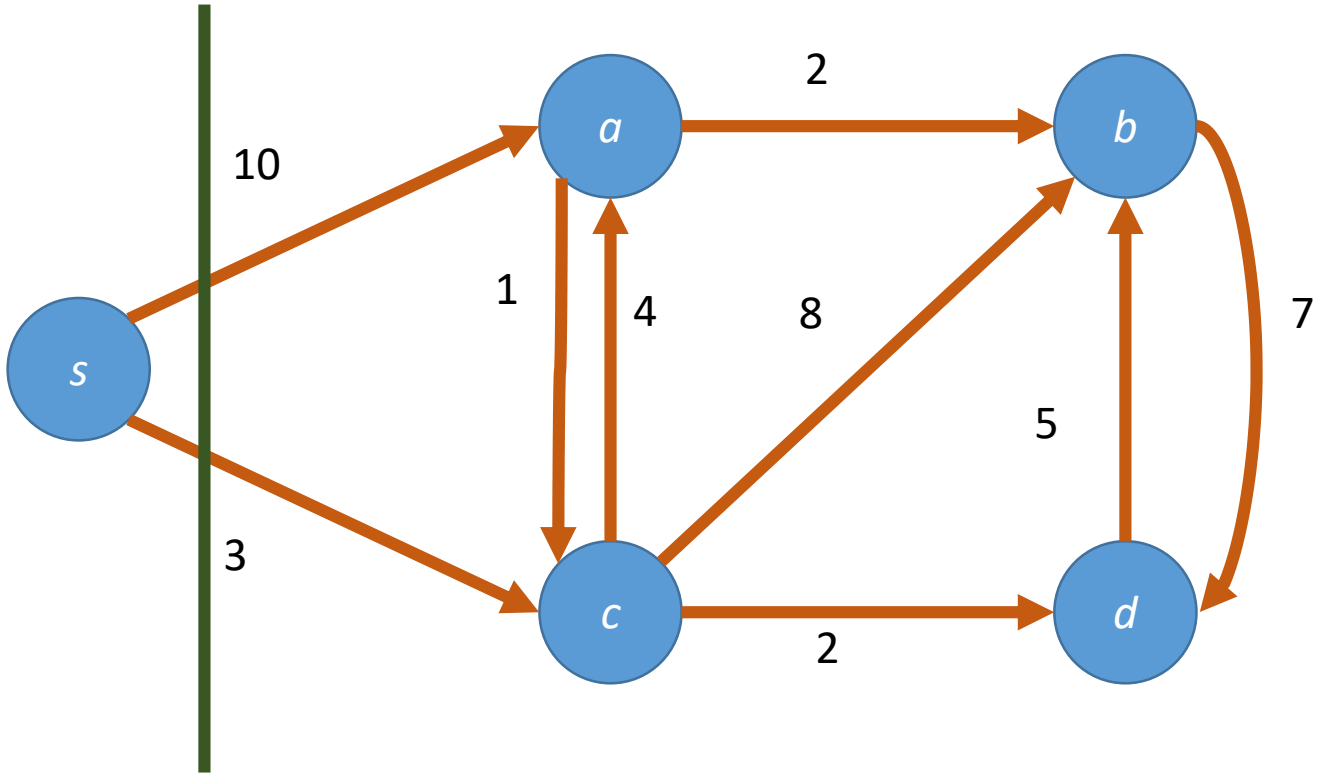
Взвешенные графы. Dijkstra

Задача: Как обойти ближайшие вершины, если заранее не известно кратчайшее расстояние до них?

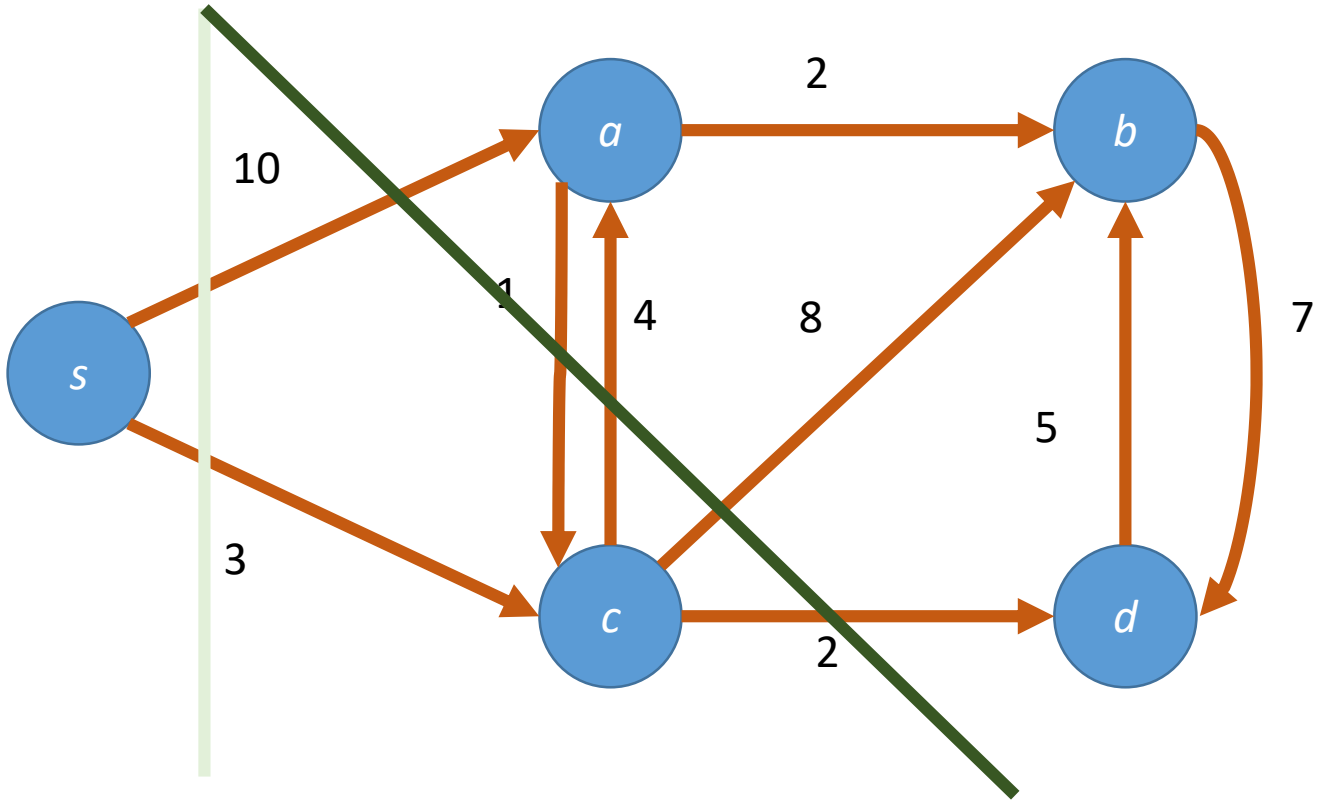
Взвешенные графы. Dijkstra



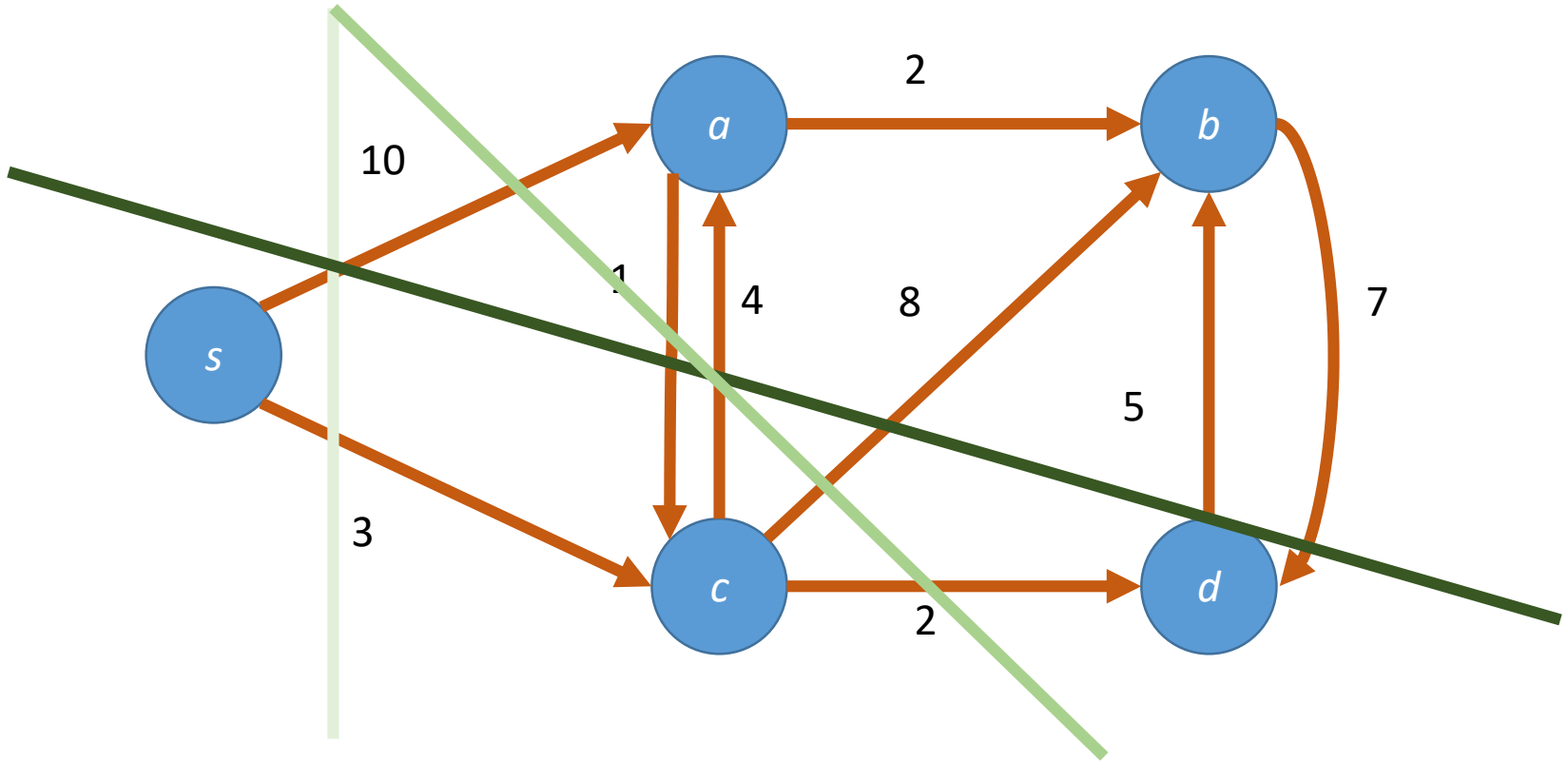
Взвешенные графы. Dijkstra



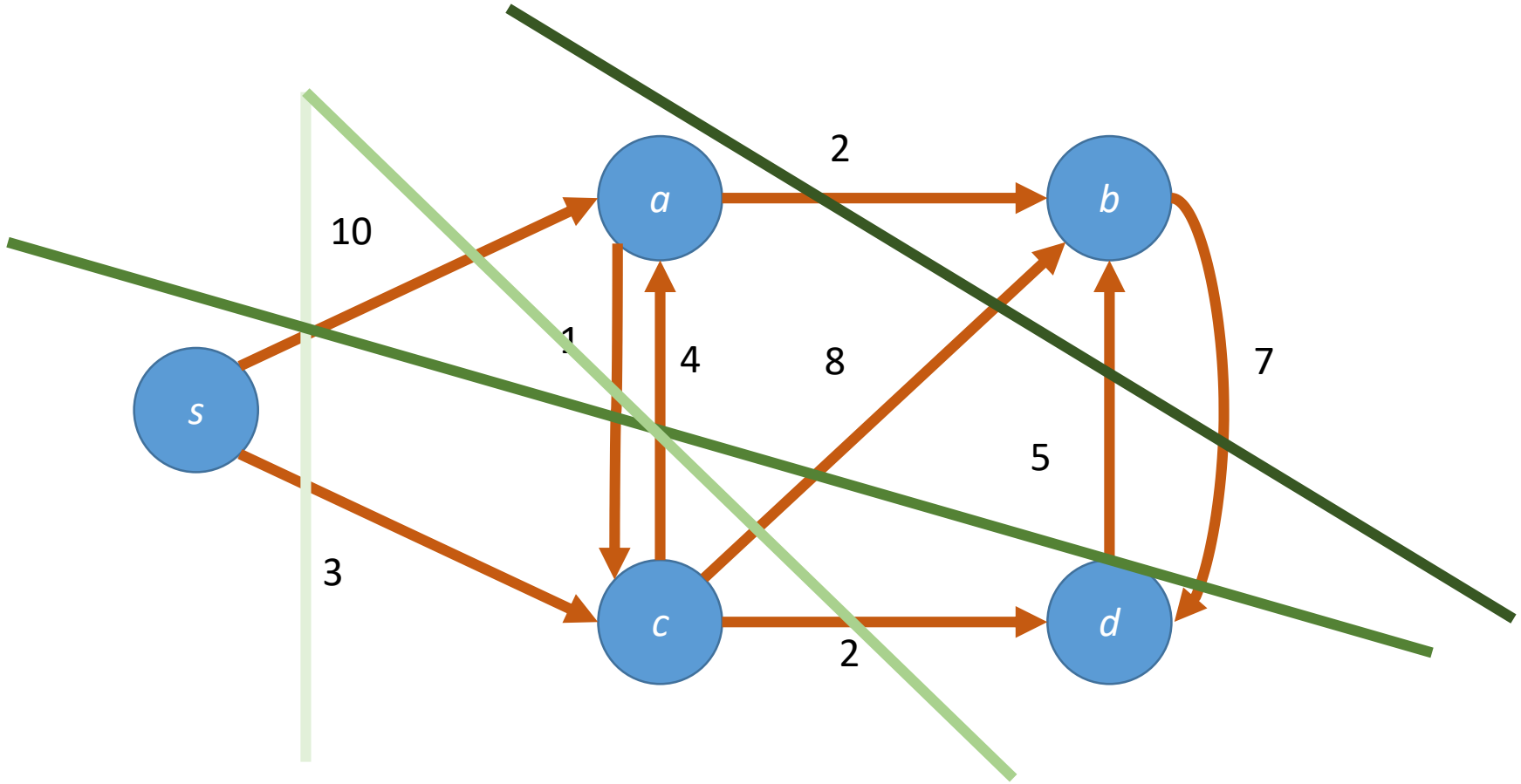
Взвешенные графы. Dijkstra



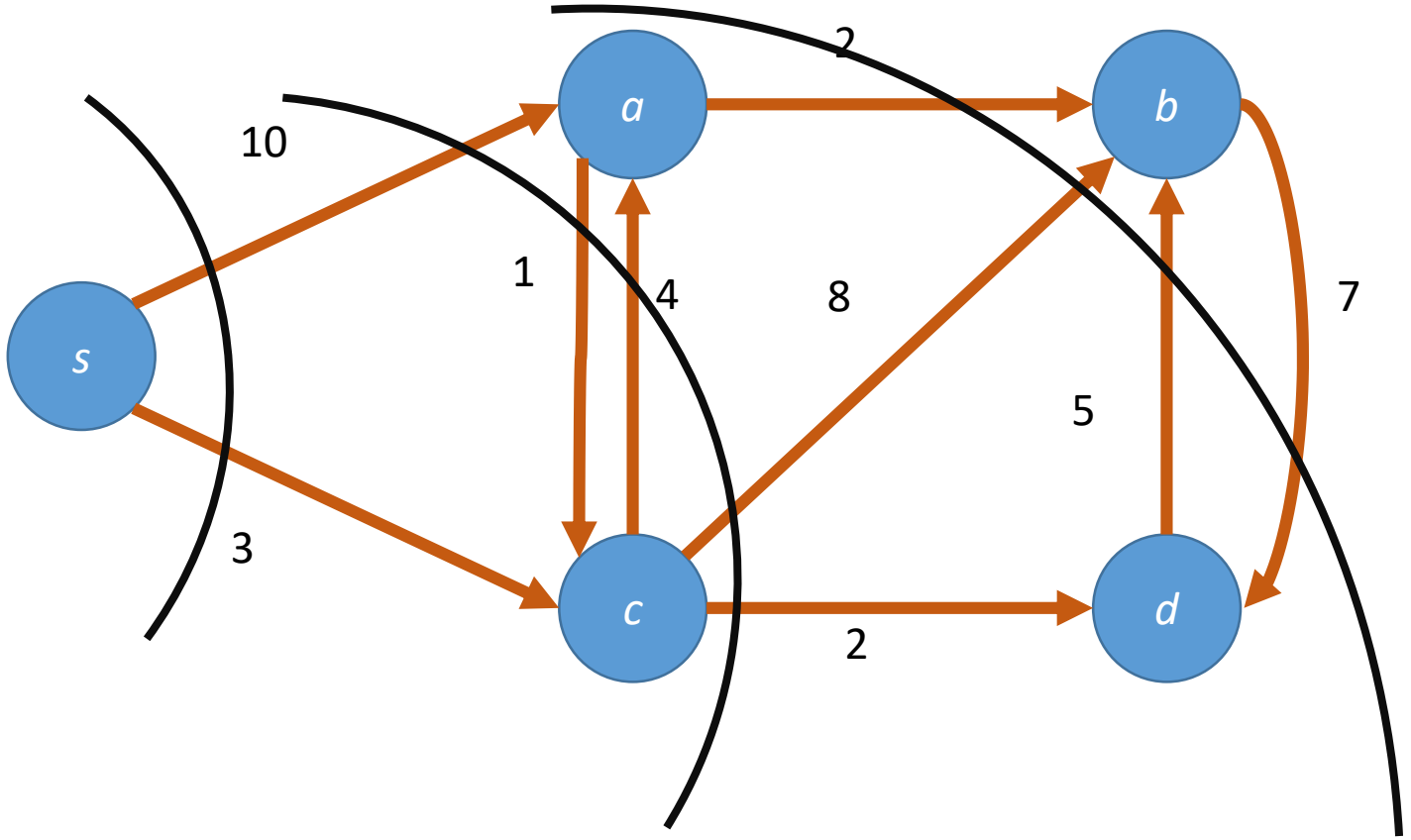
Взвешенные графы. Dijkstra



Взвешенные графы. Dijkstra



Взвешенные графы. Dijkstra



Взвешенные графы. Dijkstra

Идея! Если веса неотрицательны, то расстояние от s монотонно увеличивается вдоль кратчайших путей

Взвешенные графы. Dijkstra

Идея! Можно быстро решить SSSP, если задан порядок вершин по возрастанию расстояния от s

Взвешенные графы. Dijkstra

Relax edge (u, v) : set $d(s, v) = d(s, u) + w(u, v)$

Взвешенные графы. Dijkstra

Задача:

- “relax edges” от каждой вершины в порядке увеличения расстояния от s.
- Эффективно искать следующую вершину в порядке.

Взвешенные графы. Changeable Priority Queue

Changeable Priority Queue Q:
- build(X);
- delete_min();
- decrease_key(id, k)

- Инициализировать Q
- Удалить элемент с минимальным ключом
- Найти сохраненный элемент с id и изменить ключ на k

Взвешенные графы. Changeable Priority Queue

Binary (min) heap

Type	binary tree/heap
Invented	1964
Invented by	J. W. J. Williams

Time complexity in big O notation

Operation	Average	Worst case
Insert	O(1)	O(log *n*)
Find-min	O(1)	O(1)
Delete-min	O(log *n*)	O(log *n*)
Decrease-key	O(log *n*)	O(log *n*)
Merge	O(*n*)	O(*n*)

Space complexity

Взвешенные графы. Changeable Priority Queue

Priority Queue Q' on n items	Q Operations $O(\cdot)$			Dijkstra $O(\cdot)$ $n = \|V\| = O(\|E\|)$
	`build(x)`	`delete_min()`	`decrease_key(id, k)`	
Array	n	n	1	$\|V\|^2$
Binary Heap	n	$\log n_{(a)}$	$\log n$	$\|E\| \log \|V\|$
Fibonacci Heap	n	$\log n_{(a)}$	$1_{(a)}$	$\|E\| + \|V\| \log \|V\|$

Взвешенные графы. Dijkstra

Set $d(s,v) = \infty$ for all $v \in V$, set $d(s,s) = 0$;

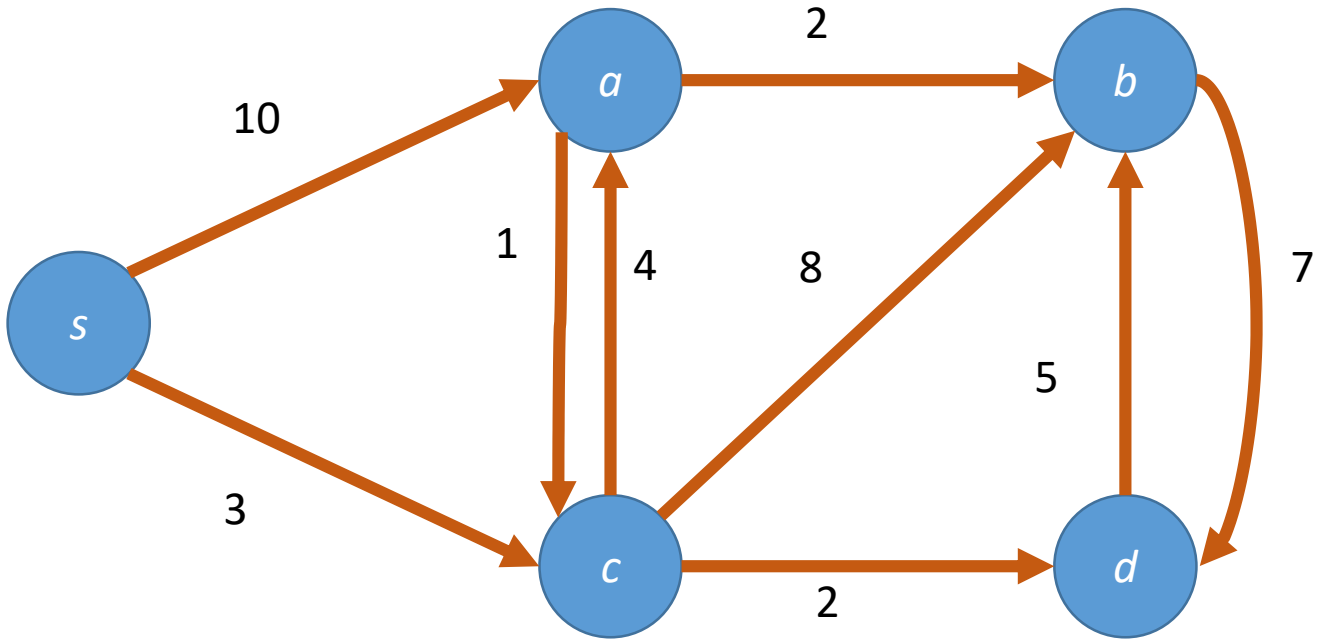
Build changeable-PQ Q with an item $\big(v, d(s,v)\big)$ for each vertex $v \in V$;

While Q not empty, delete an item $\big(u, d(s,u)\big)$ from Q that has minimum $d(s,u)$;

For vertex v in $Adj^{+}(u)$:

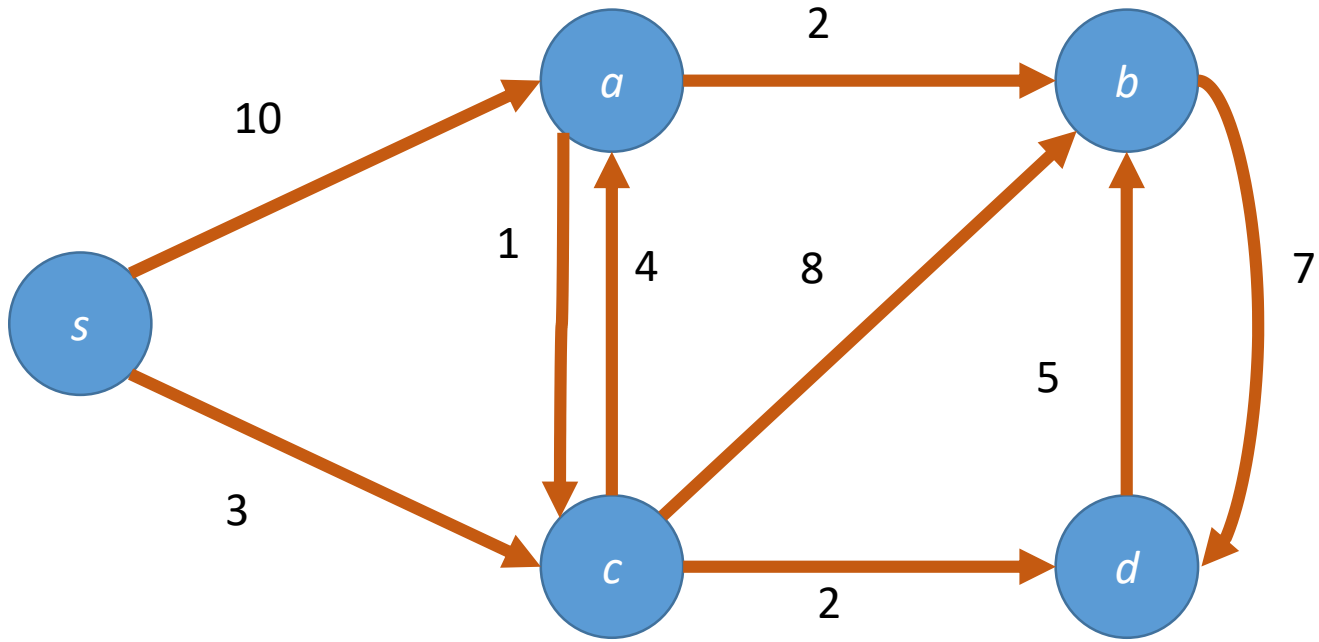
- If $d(s,v) > d(s,u) + w(u,v)$:
  - Relax edge(u,v);
  - Decrease the key of v in Q to new estimate $d(s,v)$;

Взвешенные графы. Dijkstra



Взвешенные графы. Dijkstra

Delete v from Q	$d(s,v)$ s	a	b	c	d
s	**0**	∞	∞	∞	∞
c		10	∞	**3**	∞
d		7	11		**5**
a		**7**	10		
b			**9**		
$\delta(s,v)$	0	7	9	3	5



Взвешенные графы. Dijkstra