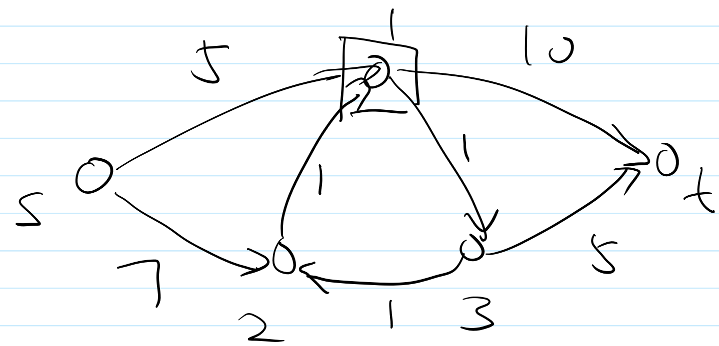
Lecture 13 Shortest Paths

Thursday, October 19, 2017 2:23 PM



- Dijkstra Algorithm

want: compute shortest paths from s to other vertices in ascending order of distance

initially only know $dis(s)=0$

first step: try to find a vertex closest to s.

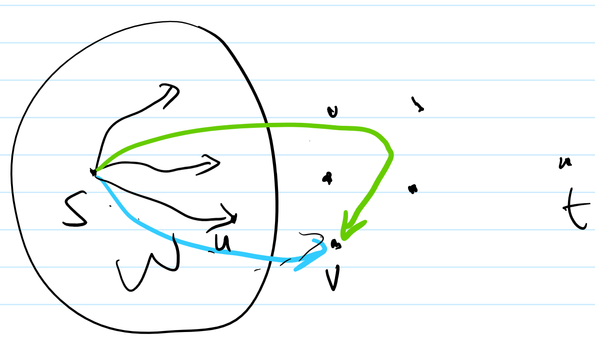
observation: the closest point must be a neighbor of s.

next step: neighbors of all vertices that we have computed before.

- maintain a set W

property: 1. know the shortest path from s to any $u \in W$

2. distance to any $u \in W$ no larger than distance to any $v \notin W$



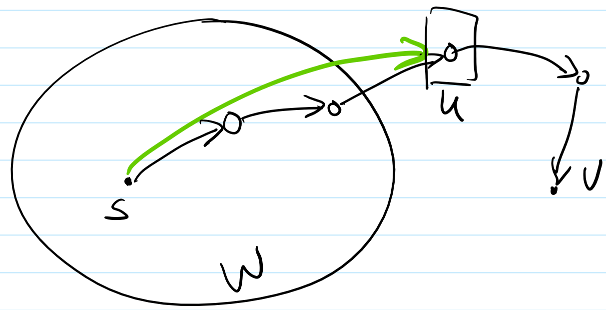
- ALG: find a vertex v ($v \notin W$), add v to W.

v should be the one with min distance to s

among all $v \notin W$.

Claim: if v is the one with min distance to S for $v \notin W$ then the shortest path from S to v only uses points in W.

Proof: assume the shortest path is not entirely in W

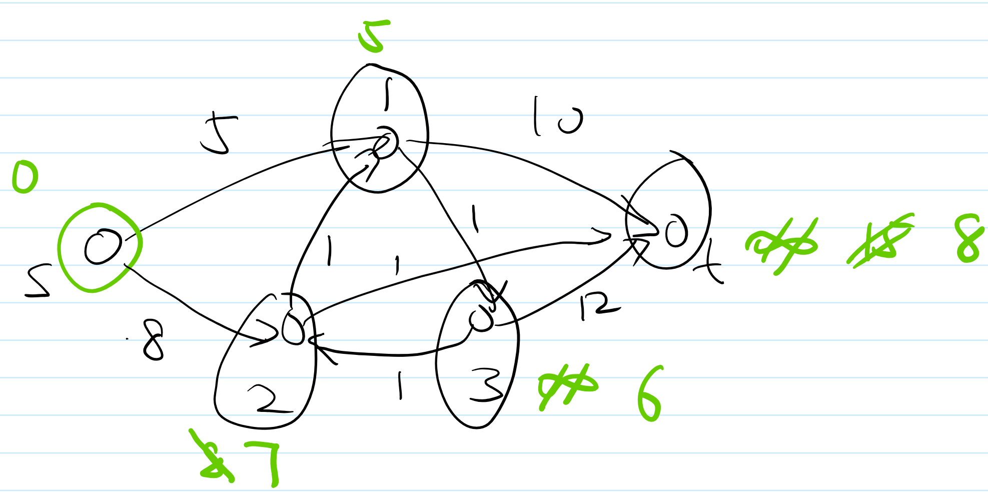


contradicts with the assumption that v is closest to S.

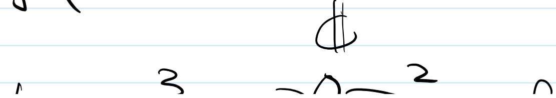
- Implementing Dijkstra's algorithm
  - maintain W (set of vertices with known shortest path)
  - maintain dis[v]

$$\begin{cases} \text{for } v \in W & \text{dis}[v] = \text{length of shortest path} \\ \text{for } v \notin W & \text{dis}[v] = \boxed{\text{length of shortest path to } v \text{ where all vertices are in } W} \end{cases}$$

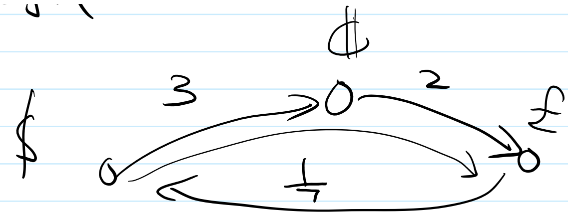
  - every iteration: find $v \notin W$ with smallest dis[v]
    add v to W, update dis[u]



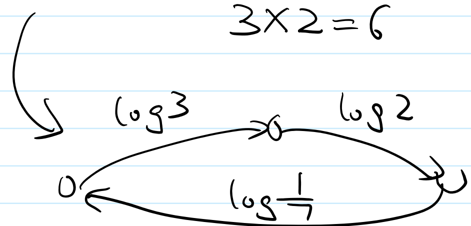
- negative edge length



- Negative edge weight



$3 \times 2 = 6$



$\log 3 + \log 2 = \log 6$

$\log 3 + \log 2 + \log\frac{1}{7} = \log\frac{6}{7} < 0$