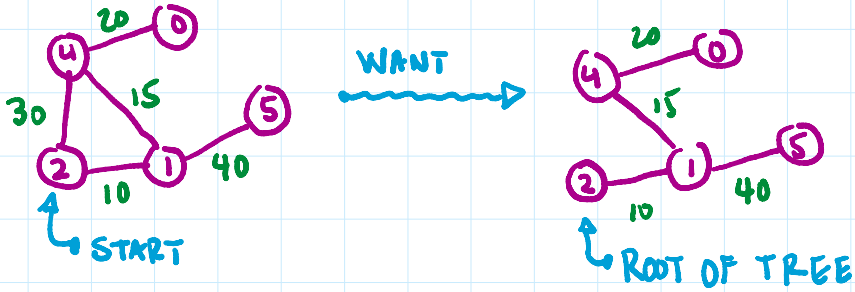
CMSC351-2023-11-08

Wednesday, November 8, 2023 8:18 AM

DIJKSTRA'S ALGORITHM

① Idea: we have a weighted, undirected simple, connected graph.
We choose a starting vertex s.
We'd like to know how to get from s to every other vertex using paths of min. total weight.

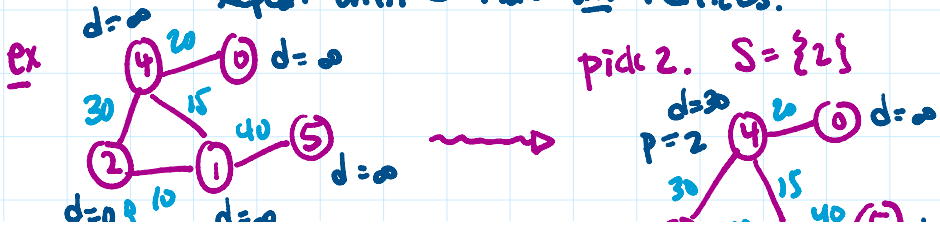
ex



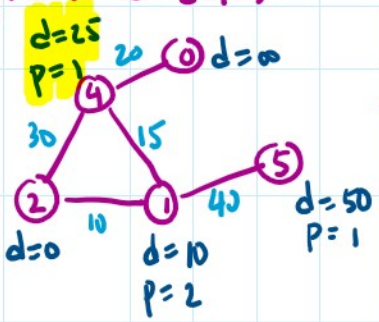
② Algorithm: Given a graph, and a starting vertex s.
Create a set $S = \{\}$ (will contain vertices)
Assign each vertex a distance of ∞ except $d[s] = 0$.
Create a list of predecessors, p, of size V all NULL.
Then:

Choose a vertex x of min dist. which is not in S.
Put it in S
For all y adjacent to x:
if $d[x] + w(x,y) < d[y]$: $w(x,y)$ = edge weight
$d[y] = d[x] + w(x,y)$
$p[y] = x$
end if
end for
Repeat until S has all vertices.

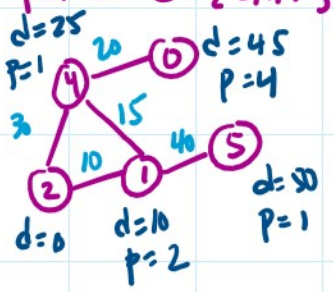
ex



pick 1: S = {2,1}



pick 4. S = {2,1,4}

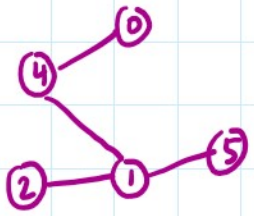


pick 0: S = {2,1,4,0}
no change

pick 5: S = {2,1,4,0,5}
no change

DONE!

Note
P[0] = 4
P[1] = 2
P[2] = NULL
P[4] = 1
P[5] = 1



ta-da!

③ Pseudocode.

```
\\ PRE: G is a graph with V vertices.
\\ PRE: s is the starting vertex.
def dijkstra(G,start):
    dist = [inf,...,inf] of length V.
    pred = [NULL,...,NULL] of length V.      } Θ(V)
    S = []
    dist[start] = 0
    while len(S) != V <-- V times
        x = vertex in G-S with smallest distance <-- if S=list, this is Θ(V).
        for each edge in the edge set of x
            let y be the connected vertex
            if dist[x] + (Weight of Edge x,y) < dist[y]:
                dist[y] = dist[x] + (Weight of Edge x,y)
                pred[y] = x
            endif
        endfor
        append x to S
    endwhile
    return(pred)
end
```

w/o for loop it's $\Theta(V^2)$ ∴
body of for loop takes
$\Theta(2E)$ time (as before)

total $\Theta(V^2+2E)$
$= \Theta(V^2+E)$.

note: Can be mod. using min heap (+ other)
to make it $\Theta(E \lg V)$.

Note: can be imp. using min heap (7 times)
to make it $\Theta(E \lg V)$.