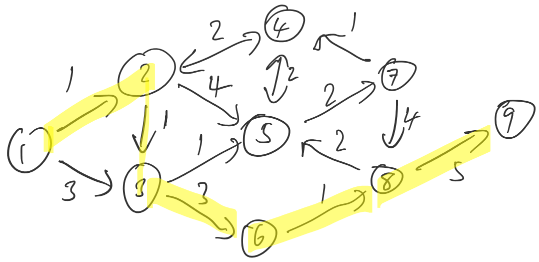
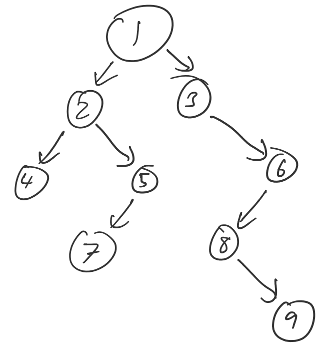
Lec09-Dijkstra (1959) KT 4.4

Thursday, September 14, 2023 3:33 PM

Shortest path in weighted, directed graph from starting node

(HW problem is about counting shortest paths in unweighted graph)





BFS works for unweighted graph.

Applications:
- Online maps & GPS navigation
- Routing systems (packets on networks)
- Drug-Pathway interaction
- Epidemiology (path of disease through social network)

Dijkstra (1959)

Use tree-growing paradigm (start w/ node, keep adding frontier edges)

Recall: Prim's MST alg: nextedge = frontier edge of min. weight.

Shortest path:
- Let s be the starting node.
- Maintain an array $d[u]$ = length of currently known shortest $s \to u$ path. $(d[s] = 0)$
- nextedge = frontier edge (u,v) that minimizes $\underbrace{d[u]}_{\text{modification from Prim's}} + \text{length}(u,v)$.

Pseudocode:

for $u \in V$, $d[u] = \infty$ (tentative distance), $p[u] = \emptyset$ (tentative parent), $F = V$ (frontier)

F = makeHeap$(V; d)$

$d[s] = 0$,

while $F \neq \emptyset$:

  $u \leftarrow$ vertex in F with min. $d[u]$ } deletemin(u) in heap — always explore next nearest node

  remove u from F

  for each neighbor v of u in F: } reduce tentative distances

    if $d[u] + \text{length}(u,v) < d[v]$:

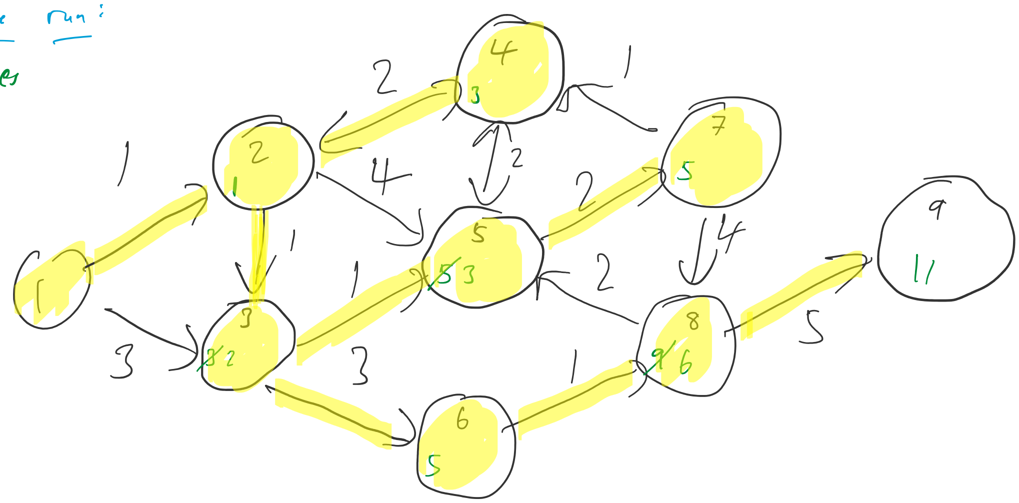
      $p[v] = u$

for each neighbor v of u:
    if d[u] + length(u,v) < d[v]:
        p[v] = u
        d[v] = d[u] + length(u,v)

} reduce tentative distances
(reducekey of heap)

return d[u], p[u]

Example run:
distances



Correctness: Let T be the set of nodes explored at some point during the alg.
$\forall u \in T$, the path found by Dijkstra is the shortest.

proof. By induction on size of T.

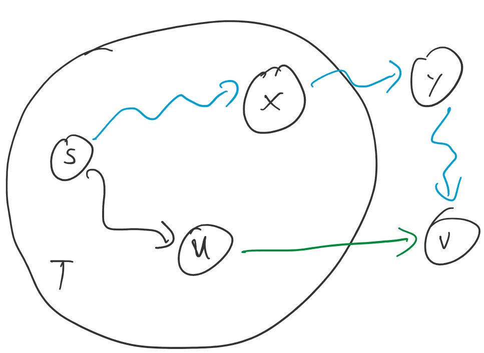
Base case: $|T| = 1$, so $T = \{s\}$, $d[s] = 0$, so correct.

Induction hypo: Assume theorem is true when $|T| \le k$.

Let v be the $(k+1)^{st}$ node added using edge (u,v).

Let P_v be the path chosen by Dijkstra.

Let P' be any other $s \to v$ path. (blue)



length($s \to u \to v$) $\le$ length (blue) by design of alg.

Thm $\exists$ optimal set of shortest paths from s and their union is a tree

Thm $\exists$ optimal set of shortest paths from s and their union is a tree

proof. Dijkstra works.

Runtime: Same as Prim's MST.

- Every edge is processed once.
  - ↳ Either:
    - (1) do nothing ; $O(1)$
    - (2) reduce key of heap item ; $O(\log |V|)$

- total time: $O(|E| \log |V|)$