Dijkstra's Algorithm

The Problem

Often when we traverse from one-point to another or from one place to another place, we want to know the shortest path.

Given a path $G = (V, E)$ which models the shortest path problem, we may want to determine -

a) given nodes u, v what is the shortest u-v path.

b) given a start node s, what is the shortest path from s to each other node.

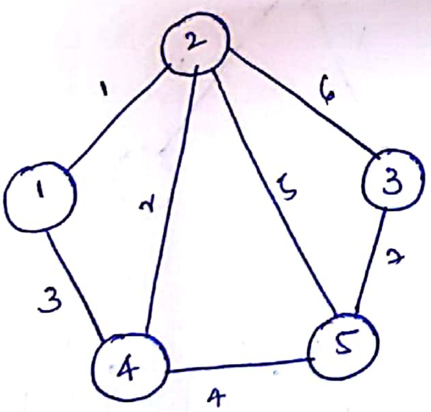
Set-up or Input Instance setup

The edges in the graph has a length $l_e \geq 0$.

The edges in the graph has a length le ≥ 0 .

l_e → indicate time it takes to traverse e to travel.

→ usually associated as cost



Take $u = 1, V = 2 \quad l_e = 1$

$u = 1 \quad V = 5$

$l_e = \{1, 4\} + \{4, 5\} = 3 + 4$
$= 7.$

$= \{1, 2\} + \{2, 5\} = 1 + 5 = 6$

$= \{1, 2\} + \{2, 4\} + \{4, 5\}$
$= 1 + 2 + 4 = 7.$

<u>Algorithm</u> :

Dijkstra's Algorithm (G, l)

Let S be the set of explored nodes.

for each $u \in S$, we store distance $d(u)$

Initially $S = \{s\}$ $d(s) = 0$.

While $S \neq V$

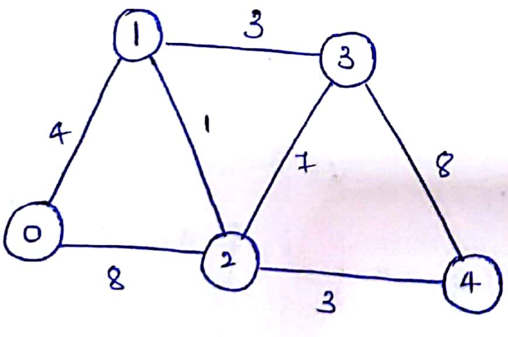
Select a node $v \notin S$ with atleast one edge from S for which

$$d'(v) = \min_{\substack{e=(u,v):\\ u \in S}} d(u) + l_e$$

add v to S and define $d(v) = d'(v)$

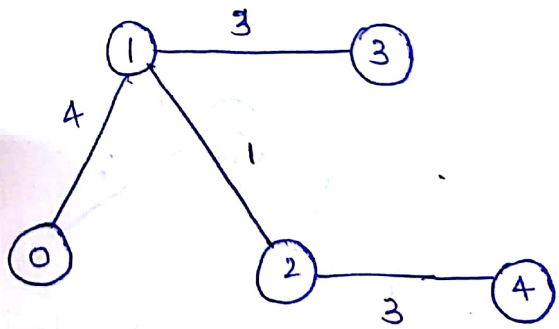
endwhile

<u>Example</u> :



$\Longrightarrow$

source = 0



d

0	1	2	3	4
0	4	5	7	8

S

distance[ ]
d[ ]

u	v

min (d[u] + l_e)
e = (u, v) : u ∈ S

{0}

0	1	2	3	4
0	∞	∞	∞	∞

d.

–

$\boxed{u = 0, v = 1}$ or u = 0, v = 2.

{0, ~~#~~ 1}

0	1
0	4

d.

min (0 + 4) = 4

$\cancel{\boxed{u = 0, v = 2}}$

{0, 1, 2}

0	4	5.

d

u = 0, v = 2 | $\boxed{u = 1, v = 2.}$

min (0 + 8) = 8.
min (4 + 1) = 5 ✓

{0, 1, 2, 3}.

0	1	2	3
0	4	5	7

d

$\boxed{u = 1. v = 3.}$ | u = 2, v = 3 | u =

min (4 + 3) = 7 ✓
min (4 + 7) = 11

{0, 1, 2, 3, 4}.

0	1	2	3	4
0	4	5	7	8

d

u = 3, v = 4 | $\boxed{u = 2, v = 4.}$

min (3 + 8) = 11
min (5 + 3) = 8.

S = V.

STOP