

$D=0, \text{end}=4$

Initialization

$\text{dist} = \{0:\infty, 1:\infty, 2:\infty, 3:\infty, 4:0\}$

$\text{parent} = \{4:\text{None}\}$

$\text{queue} = \{0, 1, 2, 3, 4\}$   
(unvisited)

Iteration 1:  $x=4$  (cost 0)

inbound edges:  $2 \rightarrow 4$  (cost 1),  $1 \rightarrow 4$  (cost 50),  $3 \rightarrow 4$  (cost 10)

Relaxation:  $\text{dist}[2] = 0+1=1, \text{parent}[2]=4$

$\text{dist}[1] = 0+50=50, \text{parent}[1]=4$

$\text{dist}[3] = 0+10=10, \text{parent}[3]=4$

Iteration 2:  $x=2$  (cost 1)

inbound edges:  $0 \rightarrow 2$  (cost 10),  $1 \rightarrow 2$  (cost 3)

Relaxation:  $\text{dist}[0] = 1+10=11, \text{parent}[0]=2$

$\text{dist}[1] = \min(50, 1+3) = 4, \text{parent}[1]=2$

Iteration 3:  $x=1$  (cost 4)

inbound edges:  $0 \rightarrow 1$  (cost 2)

Relaxation:  $\text{dist}[0] = \min(11, 4+2) = 6, \text{parent}[0]=1$

Iteration 4:  $x=0=\text{start} \Rightarrow \text{stop}$

Path reconstruction

$\text{parent}[0]=1, \text{parent}[1]=2, \text{parent}[2]=4$

$\Rightarrow \text{path} = [0, 1, 2, 4]$

cost: 6

$\text{queue} = \{0, 1, 2, 3\}$

$\text{dist} = \{0:\infty, 1:50, 2:1, 3:10, 4:0\}$

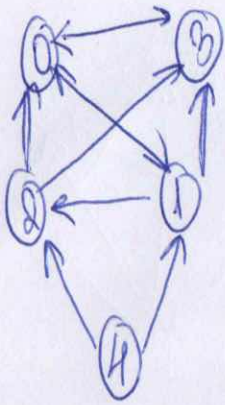
$\text{queue} = \{0, 1, 3\}$

$\text{dist} = \{0:11, 1:4, 2:1, 3:10, 4:0\}$

$\text{queue} = \{0, 3\}$

$\text{dist} = \{0:6, 1:4, 2:1, 3:10, 4:0\}$





Start = 0, end = 4

Initialization

dist = {0:∞, 1:∞, 2:∞, 3:∞, 4:0}

parent = {4:None}

queue = {0, 1, 2, 3, 4}

(unvisited)

Path reconstruction

dist = {0:∞, 1:5, 2:6, 3:8, 4:0}

parent = {4:None, 1:4, 2:4, 3:2}

dist[0] = ∞

→ raise ValueError("No path...")

Iteration 1: x = 4

inbound edges: ∅

queue = {0, 1, 2, 3}

Iteration 2: x = 0

inbound edges: [1, 2, 3]

dist[1] + 3 = ∞, dist[2] + 1 = ∞, dist[3] + 9 = ∞

queue = {1, 2, 3}

Iteration 3: x = 1

inbound edges: [0, 4]

dist[0] + 2 = ∞, dist[4] + 5 = 0 + 5 = 5 < ∞ → dist[1] = 5, parent[1] = 4

queue = {2, 3}

Iteration 4: x = 2

inbound edges: [1, 4]

dist[1] + 2 = 5 + 2 = 7 < ∞  
dist[4] + 6 = 0 + 6 = 6 < 7

→ dist[2] = 7, parent[2] = 1  
→ dist[4] = 6, parent[4] = 1

queue = {3}

Iteration 5: x = 3

inbound edges: [0, 1, 2]

dist[0] + 10 = ∞

dist[1] + 4 = 5 + 4 = 9 < ∞

→ dist[3] = 9, parent[3] = 1  
dist[2] + 2 = 6 + 2 = 8 < 9 → dist[3] = 8, parent[3] = 2

```
15. Given two vertices, find the lowest cost path between them (backwards Dijkstra)
0. Exit the program
Enter your option: 15
Enter the source vertex: 1
Enter the destination vertex: 100
Shortest distance: 141
Path: [1, 5, 487, 175, 714, 799, 222, 561, 100]
```

15. Given two vertices, find the lowest cost path between them (backwards Dijkstra)

0. Exit the program

Enter your option: 15

Enter the source vertex: 100

Enter the destination vertex: 1

Shortest distance: 196

Path: [100, 259, 229, 641, 538, 854, 1]

0. Exit the program

Enter your option: 15

Enter the source vertex: 1

Enter the destination vertex: 100

Shortest distance: 344

Path: [1, 7317, 460, 6010, 5295, 4560, 5513, 8467, 3517, 99, 9159, 6840, 5177, 7133, 288, 100]

0. Exit the program

Enter your option: 15

Enter the source vertex: 100

Enter the destination vertex: 1

Shortest distance: 238

Path: [100, 4442, 3980, 1974, 407, 4489, 5162, 2008, 3631, 2305, 8336, 1]

15. Given two vertices, find the lowest cost path between them (backward Dijkstra)

0. Exit the program

Enter your option: 15

Enter the source vertex: 100

Enter the destination vertex: 1

Shortest distance: 361

Path: [100, 85636, 77467, 52472, 38155, 40962, 34650, 29215, 35260, 1]