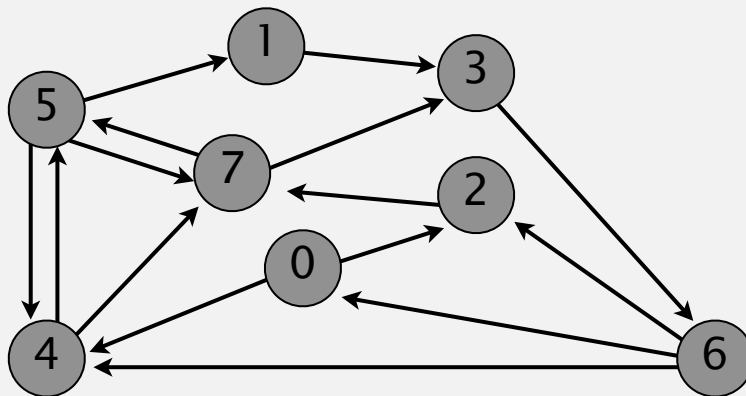


Dijkstra's Algorithm Demo

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93

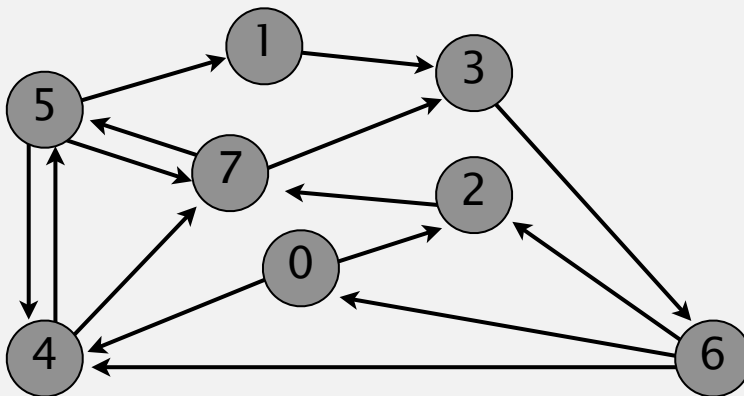


v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2		
3		
4		
5		
6		
7		

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```
4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
```



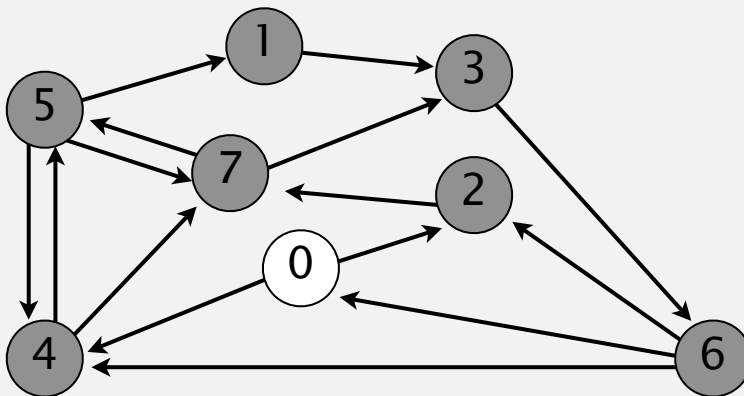
<code>v</code>	<code>distTo[v]</code>	<code>edgeTo[v]</code>
0	0.00	-
1		
2		
3		
4		
5		
6		
7		

choose vertex 0

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest $\text{distTo}[]$ value).
- Add vertex to tree and relax all edges incident from that vertex.

```
4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
```



v	$\text{distTo}[v]$	$\text{edgeTo}[v]$
0	0.00	-
1		
2		
3		
4		
5		
6		
7		

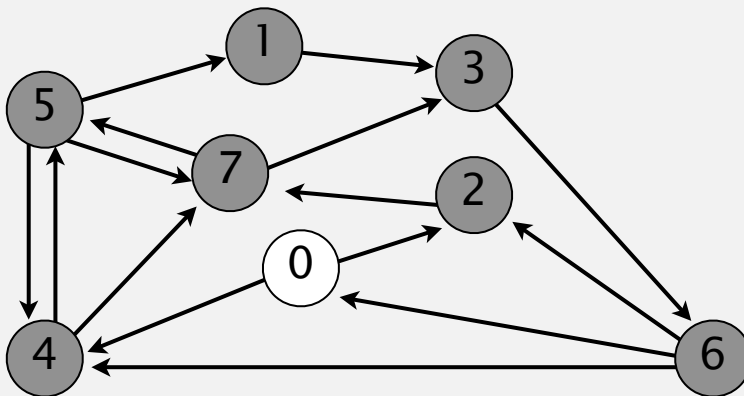
relax all edges pointing from 0: $0 \rightarrow 2$ (0.26) and $0 \rightarrow 4$ (0.38)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



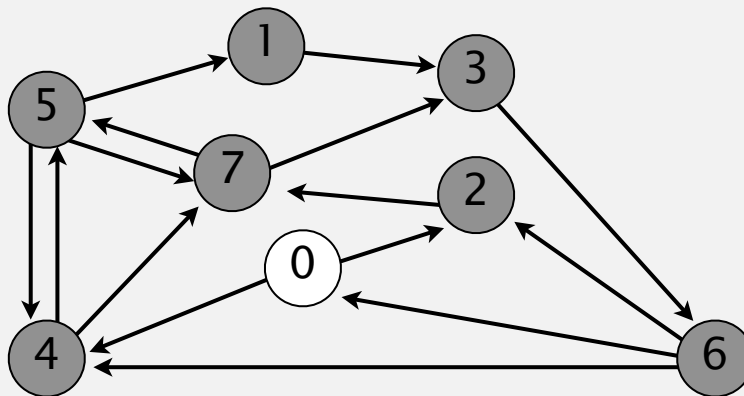
v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7		

relax all edges pointing from 0: 0→2 (0.26) and 0→4 (0.38)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5	0.35
5→4	0.35
4→7	0.37
5→7	0.28
7→5	0.28
5→1	0.32
0→4	0.38
0→2	0.26
7→3	0.39
1→3	0.29
2→7	0.34
6→2	0.40
3→6	0.52
6→0	0.58
6→4	0.93



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7		

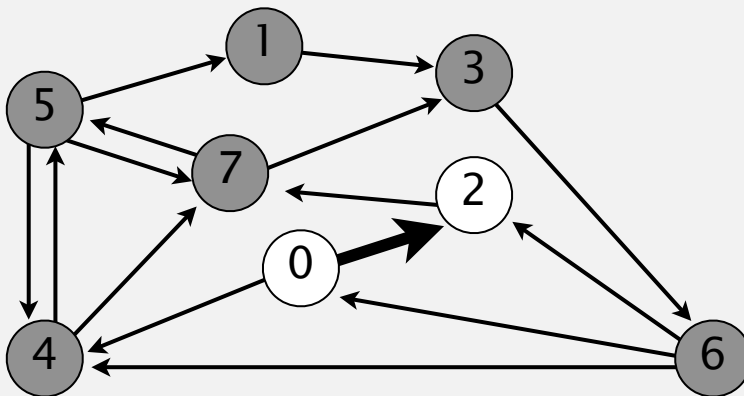
choose vertex 2

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7		

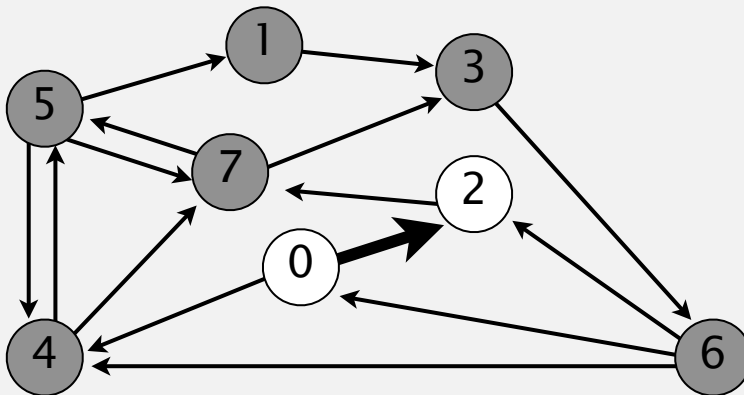
relax all edges pointing from 2: 2→7 (0.34)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7	0.60	2→7

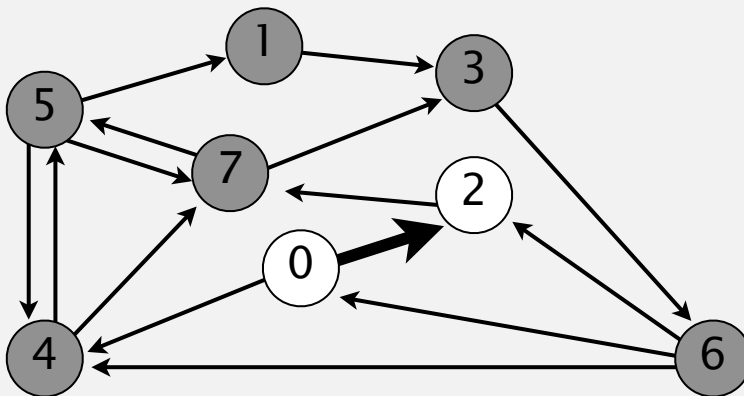
0.26 + 0.34

relax all edges pointing from 2: 2→7 (0.34)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest $\text{distTo}[]$ value).
- Add vertex to tree and relax all edges incident from that vertex.

```
4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7	0.60	2→7

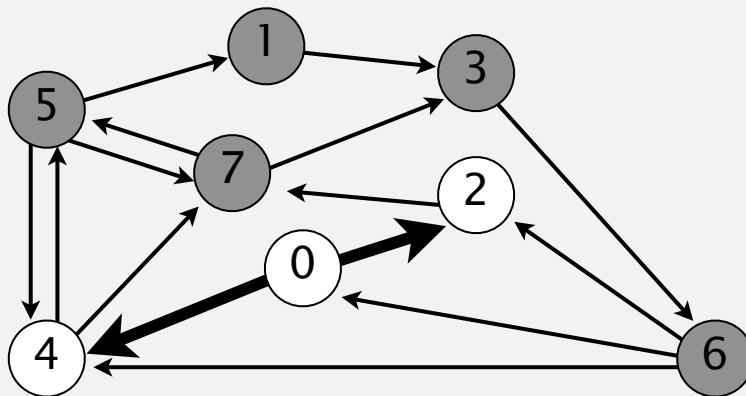
choose vertex 4

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7	0.60	2→7

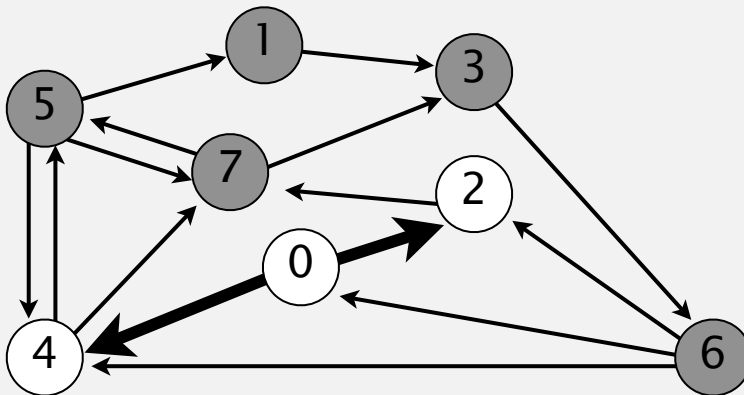
choose vertex 4

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



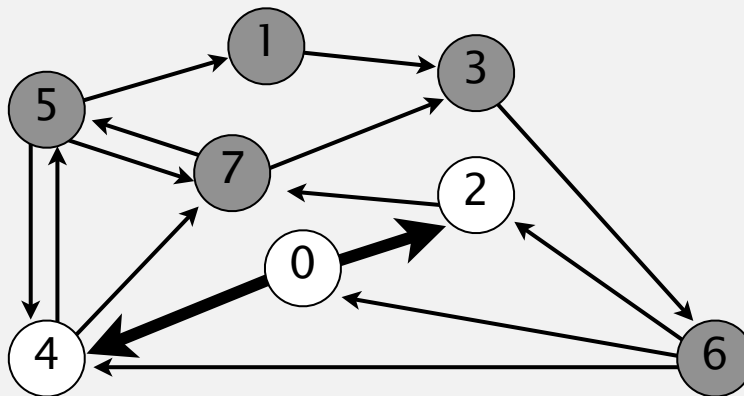
v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5		
6		
7	0.60	2→7

relax all edges pointing from 4: 4→5 (0.35) and 4→7 (0.37)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5	0.35
5→4	0.35
4→7	0.37
5→7	0.28
7→5	0.28
5→1	0.32
0→4	0.38
0→2	0.26
7→3	0.39
1→3	0.29
2→7	0.34
6→2	0.40
3→6	0.52
6→0	0.58
6→4	0.93



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

relax all edges pointing from 4: 4→5 (0.35) and 4→7 (0.37)

$$0.38 + 0.35$$

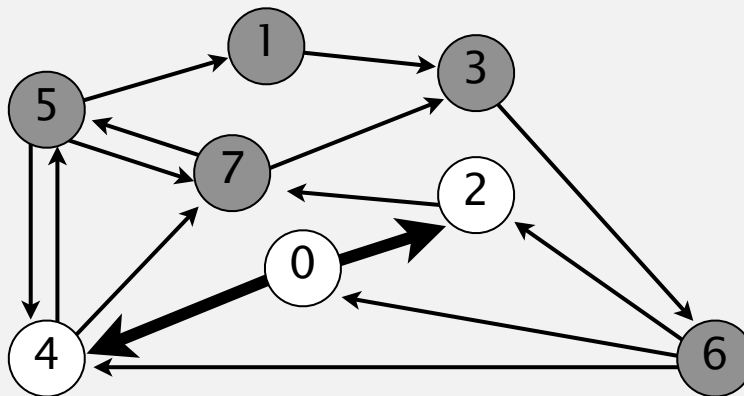
$$0.38 + 0.37$$

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



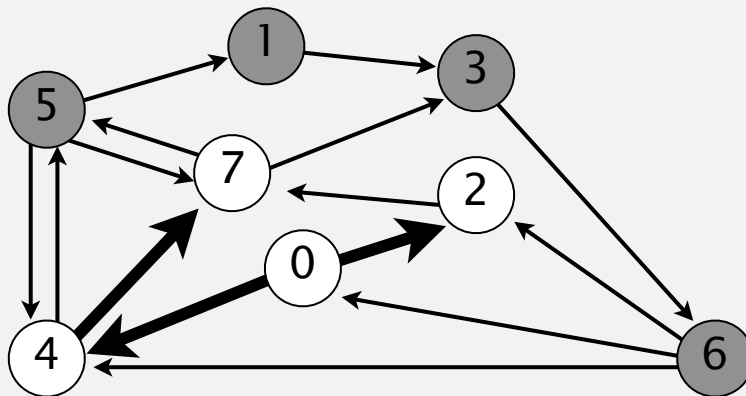
v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

choose vertex 7

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5	0.35
5→4	0.35
4→7	0.37
5→7	0.28
7→5	0.28
5→1	0.32
0→4	0.38
0→2	0.26
7→3	0.39
1→3	0.29
2→7	0.34
6→2	0.40
3→6	0.52
6→0	0.58
6→4	0.93



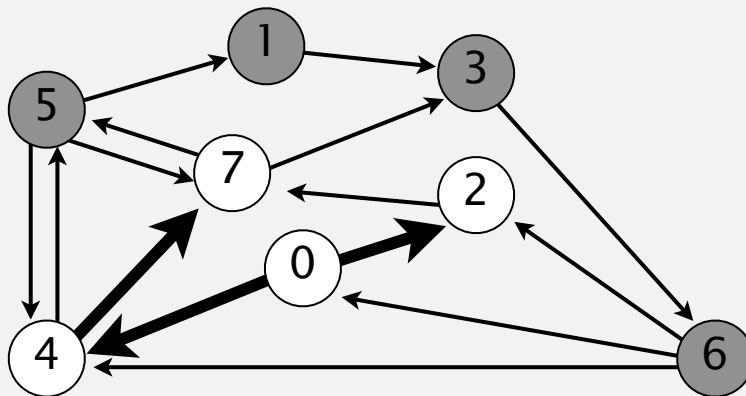
v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

choose vertex 7

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5	0.35
5→4	0.35
4→7	0.37
5→7	0.28
7→5	0.28
5→1	0.32
0→4	0.38
0→2	0.26
7→3	0.39
1→3	0.29
2→7	0.34
6→2	0.40
3→6	0.52
6→0	0.58
6→4	0.93



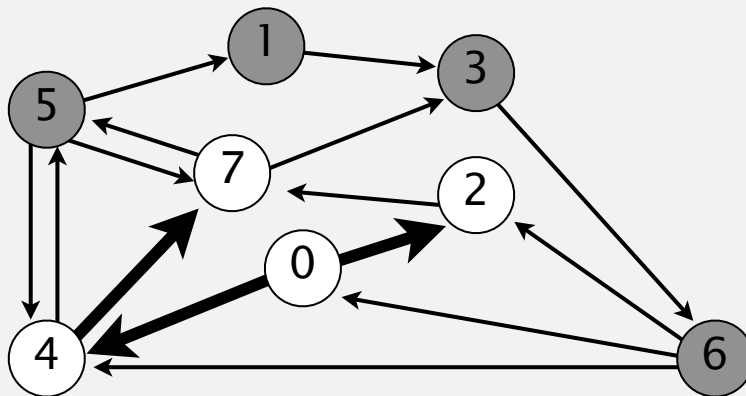
v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3		
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

relax all edges pointing from 7: 7→3 (0.39) and 7→5 (0.28)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5	0.35
5→4	0.35
4→7	0.37
5→7	0.28
7→5	0.28
5→1	0.32
0→4	0.38
0→2	0.26
7→3	0.39
1→3	0.29
2→7	0.34
6→2	0.40
3→6	0.52
6→0	0.58
6→4	0.93



v	<code>distTo[v]</code>	<code>edgeTo[v]</code>
0	0.00	-
1		
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

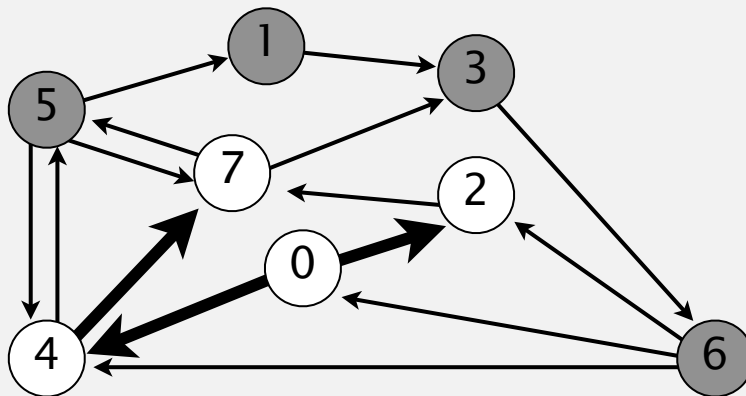
$0.60 + 0.39$

relax all edges pointing from 7: 7→3 (0.39) and 7→5 (0.28)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5 0.35
 5→4 0.35
 4→7 0.37
 5→7 0.28
 7→5 0.28
 5→1 0.32
 0→4 0.38
 0→2 0.26
 7→3 0.39
 1→3 0.29
 2→7 0.34
 6→2 0.40
 3→6 0.52
 6→0 0.58
 6→4 0.93



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

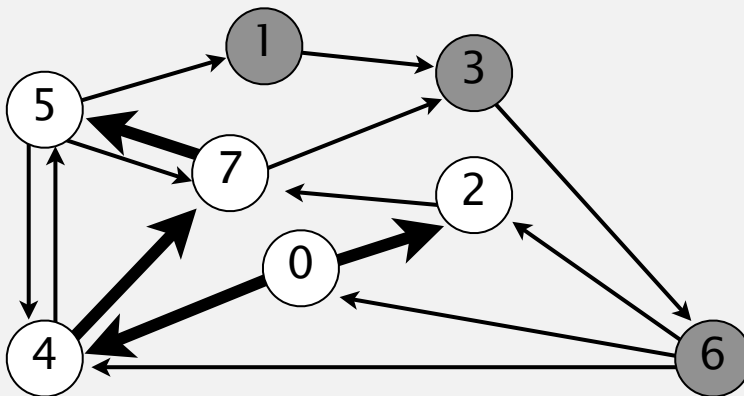
choose vertex 5

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

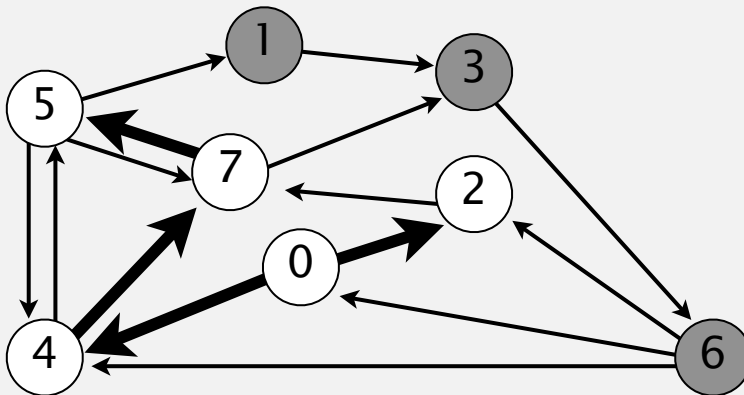
choose vertex 5

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



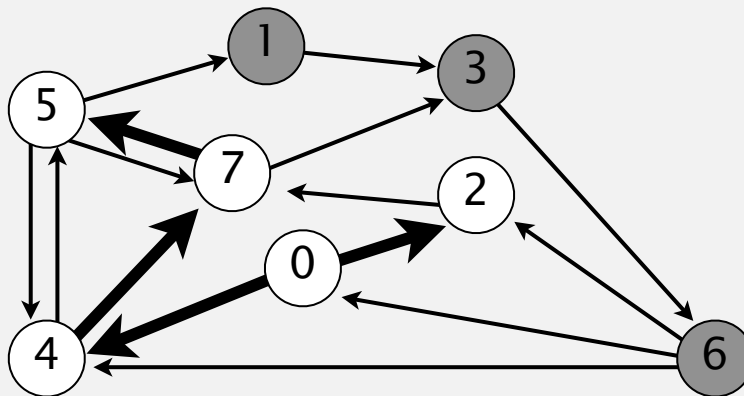
v	distTo[v]	edgeTo[v]
0	0.00	-
1		
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

relax all edges pointing from 5: 5→1 (0.32), 5→4 (0.35), and 5→7 (0.28)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5	0.35
5→4	0.35
4→7	0.37
5→7	0.28
7→5	0.28
5→1	0.32
0→4	0.38
0→2	0.26
7→3	0.39
1→3	0.29
2→7	0.34
6→2	0.40
3→6	0.52
6→0	0.58
6→4	0.93



v	<code>distTo[v]</code>	<code>edgeTo[v]</code>
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

0.73 + 0.32

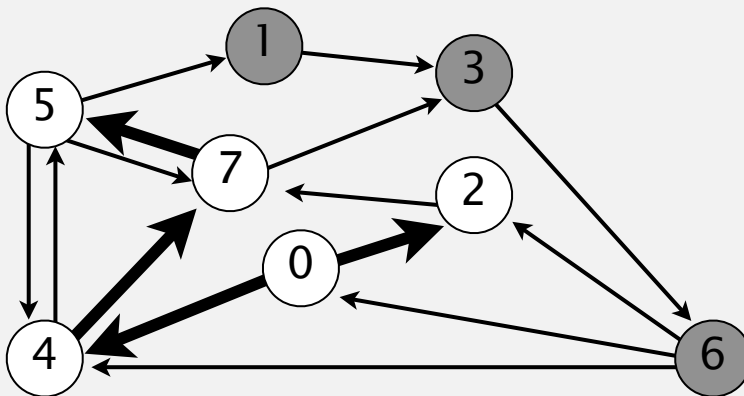
relax all edges pointing from 5: 5→1 (0.32), 5→4 (0.35), and 5→7 (0.28)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



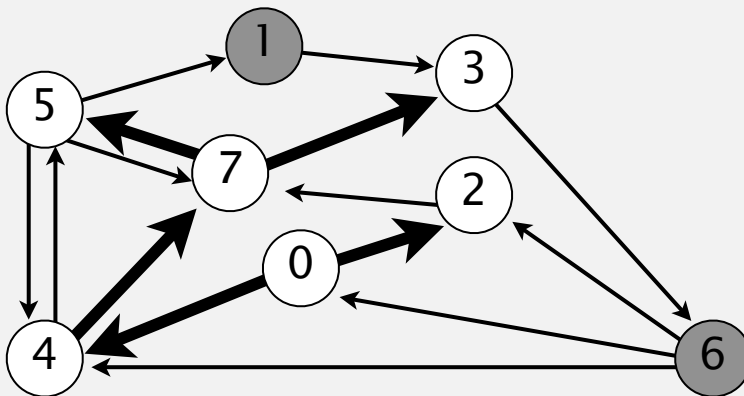
v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

select vertex 3

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5 0.35
 5→4 0.35
 4→7 0.37
 5→7 0.28
 7→5 0.28
 5→1 0.32
 0→4 0.38
 0→2 0.26
 7→3 0.39
 1→3 0.29
 2→7 0.34
 6→2 0.40
 3→6 0.52
 6→0 0.58
 6→4 0.93



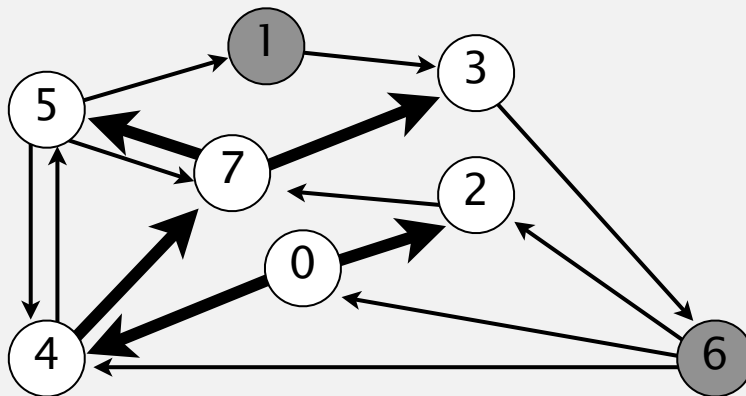
v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

select vertex 3

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5 0.35
 5→4 0.35
 4→7 0.37
 5→7 0.28
 7→5 0.28
 5→1 0.32
 0→4 0.38
 0→2 0.26
 7→3 0.39
 1→3 0.29
 2→7 0.34
 6→2 0.40
 3→6 0.52
 6→0 0.58
 6→4 0.93



v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6		
7	0.60	2→7

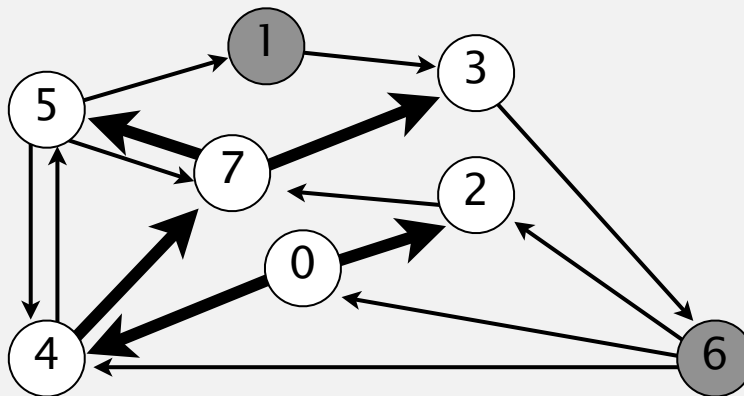
relax all edges pointing from 3: 3→6 (0.52)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

0.99 + 0.52

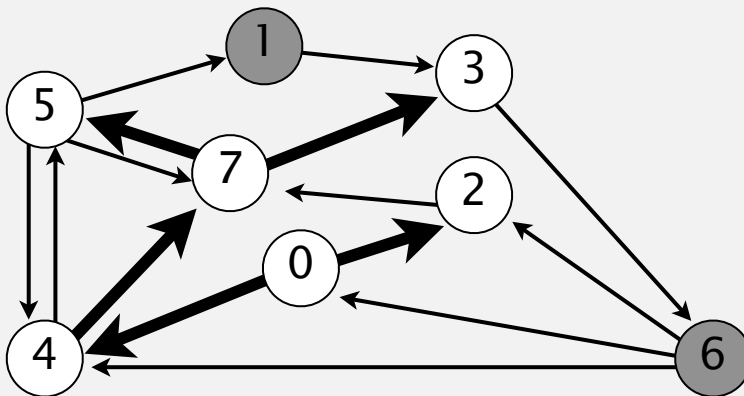
relax all edges pointing from 3: 3→6 (0.52)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

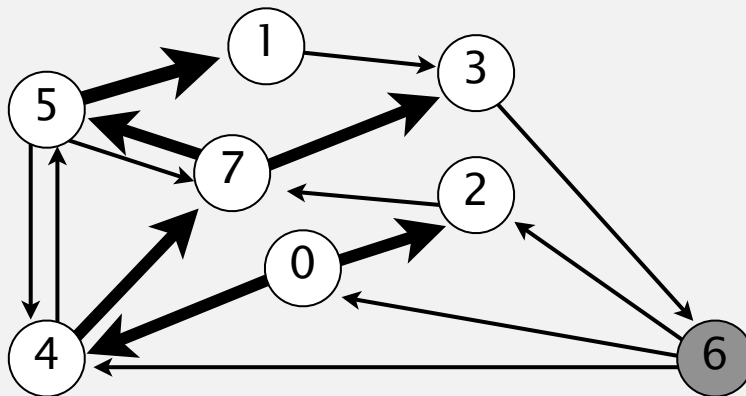
select vertex 1

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

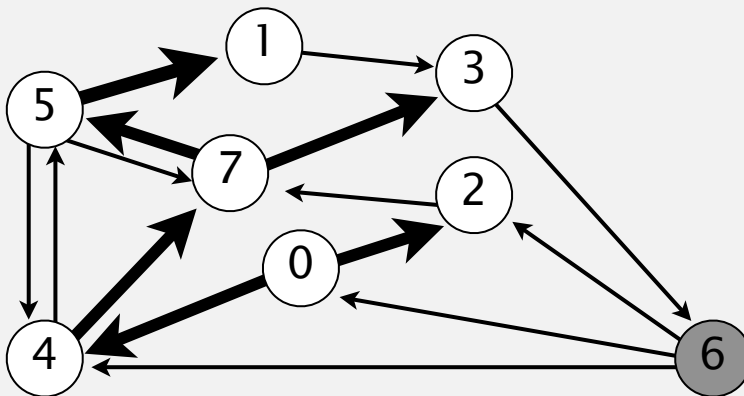
select vertex 1

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



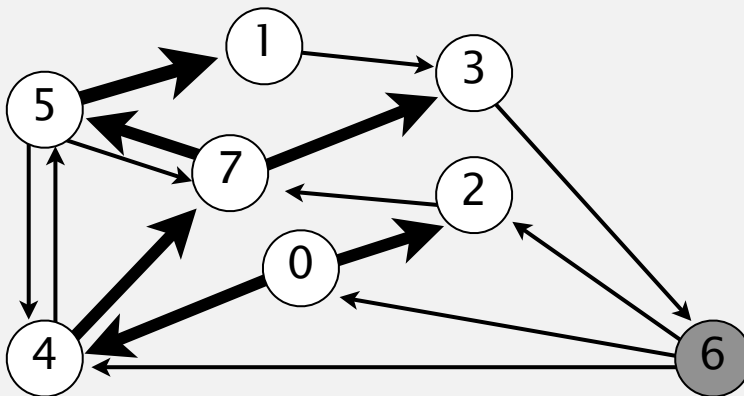
v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

relax all edges pointing from 1: 1→3 (0.29)

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest $\text{distTo}[]$ value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5 0.35
 5→4 0.35
 4→7 0.37
 5→7 0.28
 7→5 0.28
 5→1 0.32
 0→4 0.38
 0→2 0.26
 7→3 0.39
 1→3 0.29
 2→7 0.34
 6→2 0.40
 3→6 0.52
 6→0 0.58
 6→4 0.93



v	$\text{distTo}[v]$	$\text{edgeTo}[v]$
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

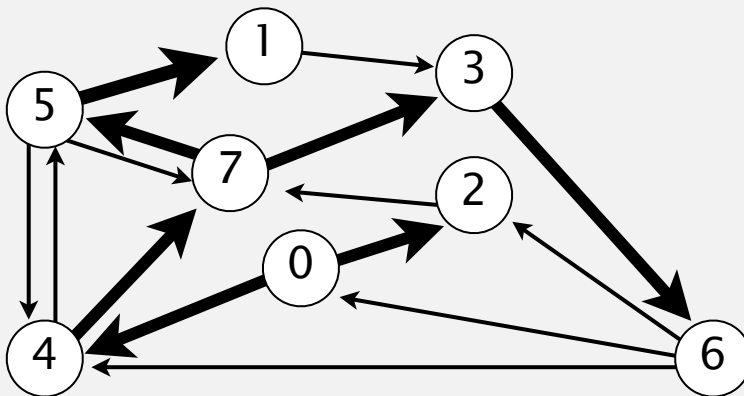
select vertex 6

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

```

4->5 0.35
5->4 0.35
4->7 0.37
5->7 0.28
7->5 0.28
5->1 0.32
0->4 0.38
0->2 0.26
7->3 0.39
1->3 0.29
2->7 0.34
6->2 0.40
3->6 0.52
6->0 0.58
6->4 0.93
    
```



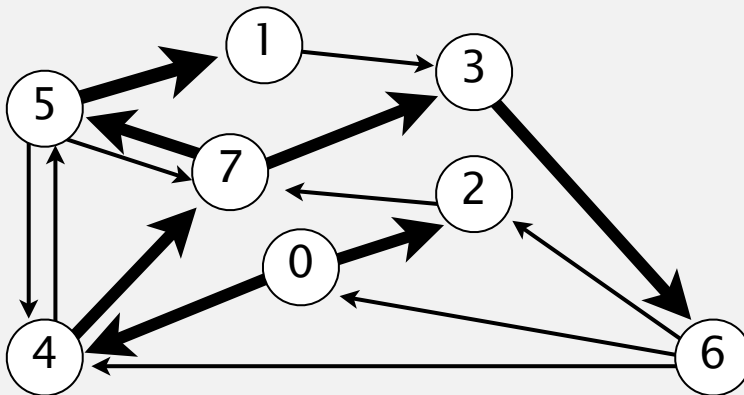
v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

select vertex 6

Dijkstra's algorithm

- Consider vertices in increasing order of distance from s (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges incident from that vertex.

4→5 0.35
 5→4 0.35
 4→7 0.37
 5→7 0.28
 7→5 0.28
 5→1 0.32
 0→4 0.38
 0→2 0.26
 7→3 0.39
 1→3 0.29
 2→7 0.34
 6→2 0.40
 3→6 0.52
 6→0 0.58
 6→4 0.93



v	distTo[v]	edgeTo[v]
0	0.00	-
1	1.05	5→1
2	0.26	0→2
3	0.99	7→3
4	0.38	0→4
5	0.73	4→5
6	1.51	3→6
7	0.60	2→7

relax all edges pointing from 6: 6→0 (0.58), 6→2 (0.40), and 6→4 (0.93)