

29.

(1) distance - from Source

i	1	2	3	4	5	6	7	8	9	10
dis	0	2	∞	20	∞	∞	∞	∞	10	∞

predecessor

i	1	2	3	4	5	6	7	8	9	10
pre	0	1	-1	1	-1	-1	-1	-1	-1	-1

(2) loop

num	1	2	3	4	5	6	7	8	9	10
dis	0	2	∞	20	3	∞	∞	∞	10	∞
pre	0	1	-1	1	2	-1	-1	-1	-1	-1

2	dis	0	2	10	20	3	∞	∞	6	10	∞
pre	0	1	5	1	2	-1	-1	5	-1	-1	

3	dis	0	2	10	20	3	8	∞	6	10	8
pre	0	1	5	1	2	8	-1	5	-1	8	

5	dis	0	2	9	20	3	8	∞	6	10	8
pre	0	1	6	1	2	8	-1	5	-1	8	

6	dis	0	2	9	20	3	8	∞	6	9	8
pre	0	1	6	1	2	8	-1	5	10	8	

7	dis	0	2	9	20	3	8	∞	6	9	8
pre	0	1	6	1	2	8	-1	5	10	8	

8	dis	0	2	9	20	3	8	11	6	9	8
pre	0	1	6	1	2	8	9	5	10	8	

9	dis	0	2	9	20	3	8	11	6	9	8
pre	0	1	6	1	2	8	9	5	10	8	

27. 对图 17-17 的加权有向图，列出所有从源点 1 到顶点 3 的简单路径。每一条路径的长度是多少？哪一条是最短路径？

28. 对图 17-17，按长度递增顺序，列出从源点 1 到达每一个顶点的最短路径。观察除第一条路径之外的每条路径，每一条路径都是在第一条路径上扩充了一条边。



图 17-17 一个加权有向图

29. 1) 对图 17-17，以顶点 1 为源点，显示 Dijkstra 方法（图 17-10）的初始时的数组 distance-FromSource 和 predecessor.

2) 显示这两个数组在步骤 2)、3) 和 4) 每一次循环之后的值。

3) 使用最后的数组 predecessor 确定从源点 1 开始的最短路径。

(3)

to 2: 1 → 2

to 3: 1 → 2 → 5 → 8 → 6 → 3

to 4: 1 → 4

to 5: 1 → 2 → 5

to 6: 1 → 2 → 5 → 8 → 6

to 7: 1 → 2 → 5 → 8 → 10 → 7

to 8: 1 → 2 → 5 → 8

to 9: 1 → 2 → 5 → 8 → 10 → 9

to 10: 1 → 2 → 5 → 8 → 10

快速排列

10, 9, 6, 7, 5, 14, 2, 8

1° 10 9 6 7 5 8 2 14

2° 2 9 6 7 5 8 10 14

3° 2 9 6 7 5 8 10 14

4° 2 8 6 7 5 9 10 14

5° 2 5 6 7 8 9 10 14