

Dijkstra 算法 (最短路径)

1. 算法 (1) 在还未求出到选取点最近的点的集合 ($V_i - U$) 中选取距离最小的点加入集合 U ; 点记为 V_{min}

(2) 修正。如果 $V_{min}.length + arcs[min][i] \leq V_i.length$, 则把 $V_i.length$ 改为较小的, 且 V_i 的前趋顶点改为 V_{min} 。(其中 V_i 是 $V - U$ 中剩下的点)。

2. 内核:

```

void dijkstra (graph, dist[])
{
    // dist[] 初始化
    for (i=1; i<graph.n; i++) {
        minw=MAX; min=0;
        for (j=1; j<graph.n; j++)
            if ((graph.arcs[i][j]==0) && (dist[j].length<minw)) {
                minw=dist[j].length; min=j;
            }
        if (min==0) break;
        graph.arcs[min][min]=1; // 加入 U
        for (j=1; j<graph.n; j++)
            if ((graph.arcs[min][j] != 1) && (dist[min].length+graph.arcs[min][j]) {
                dist[j].length=dist[min].length+graph.arcs[min][j];
                dist[j].prever=min;
            }
    }
}

```

3. 图示

① ② ③ ④ ⑤ ⑥

有向图 G 边矩阵

$\text{arcs}[j][i] =$

0	50	10	∞	45	∞	0
∞	0	15	∞	5	∞	0
20	∞	0	15	∞	∞	0
∞	20	∞	0	35	∞	0
∞	∞	∞	30	0	∞	0
∞	∞	∞	3	∞	0	0

① $\text{dist}[n] = \{ \{10, 0\}, \{50, 0\}, \{10, 0\}, \{\text{MAX}, -1\}, \{45, 0\}, \{\text{MAX}, -1\} \}$

$\uparrow v_1$
length process

\uparrow
元边

② $\{10, 0\}, \{30, 0\}, \{10, 0\}, \{25, 2\}, \{45, 0\}, \{\text{MAX}, -1\}$

\checkmark

\checkmark 新

③ $\{10, 0\}, \{45, 3\}, \{10, 0\}, \{25, 2\}, \{45, 0\}, \{\text{MAX}, -1\}$

\checkmark

新

④ $\{10, 0\}, \{45, 3\}, \{10, 0\}, \{25, 2\}, \{45, 0\}, \{\text{MAX}, -1\}$

\checkmark

\checkmark

\checkmark

\checkmark

\uparrow

结束.

没有从 v_5 到

v_5 的路径.

$v_0 \rightarrow v_1$ 的路径通过得: $v_1 \leftarrow v_2 \leftarrow v_3 \leftarrow v_0$

时间 $O(n^2)$ 空间 $O(n)$