



## 实验四：图的操作

实验内容：

- 1、输入图的顶点数 $n$ （不超过10个）、边数 $m$ ，顶点分别用0到 $n-1$ 表示；
- 2、采用“起始顶点，终止顶点，权值”输入图的 $m$ 条边，创建图；
- 3、输出从顶点0开始的BFS遍历、DFS遍历的顶点遍历顺序；
- 4、输出从顶点0到其余顶点最短路径及最短路径的长度，如果没有路径，输出0。



## Dijkstra的算法7.15

- (1) 带权的邻接矩阵arcs表示带权有向图, arcs[i][j]表示弧 $\langle v_i, v_j \rangle$ 上的权值。若 $\langle v_i, v_j \rangle$ 不存在, 则arcs[i][j]为 $\infty$ 。S为已找到从v出发的最短路径的终点的集合, 初始状态为空集。那么从v出发到图上其余各顶点  $v_i$ 可能达到的最短路径长度初始值为:
$$D[i] = \text{arcs}[\text{Locate Vex}(G, v)][i] \quad v_i \in V$$
- (2) 选择 $v_j$ , 使得  $D[j] = \text{Min}\{D[i] \mid v_i \in V-S\}$ ,  $v_j$ 即当前求得的一条从V出发的最短路径的终点。令 $S = S \cup \{j\}$
- (3) 修改从v出发到集合 $V - S$ 上任一顶点 $v_k$ 可达的最短路径长度。如果 $D[j] + \text{arcs}[j][k] < D[k]$ , 则修改 $D[k] = D[j] + \text{arcs}[j][k]$
- (4) 重复步骤(2)(3)共 $n-1$ 次, 由此求得从v到图上其余各顶点的最短路径是依路径长度递增的序列。



## Dijkstra的算法7.15

```
void ShortestPath_DIJ(MGraph G, int v0, PathMatrix &P, ShortPathTable &D){
```

```
    //用Dijkstar算法求有向网G的v0顶点到其余顶点v的最短路径P[v]及其带权长度D[v].
```

```
    //若P[v][w]为TRUE,则w是从v0到v当前求得最短路径上的顶点
```

```
    //final[v]为TRUE当且仅当 $v \in S$ , 即已经求得从v0到v最短路径。
```

```
    for(v=0; v<G.vexnum; ++v)    {
```

```
        final[v] = FALSE;        D[v] = G.arcs[v0][v];
```

```
        for(w=0; w<G.vexnum; ++w)    P[v][w] = FALSE;                //设空路径
```

```
        if(D[v]<INFINITY)    {P[v][v0] = TRUE;        P[v][v] = TRUE;}
```

```
    }//for
```

```
    D[v0] = 0; final[v0] = TRUE;                //初始化, v0顶点属于S集
```



## Dijkstra的算法7.15

//开始主循环，每次求得v0到某个v顶点的最短路径，并加v到S集

for(i=1; i<G.vexnum; ++i) { //其余G1.vexnum-1个顶点

min = INFINITY; //当前所知离v0顶点的最近距离

for(w=0; w<G.vexnum; ++w){

if(!final[w]) //w顶点在V-S中

if(D[w]<min){v = w; min = D[w];} //w顶点离v0顶点更近

final[v] = TRUE; //离v0顶点最近的v加入S集

for(w=0; w<G.vexnum; ++w) //更新当前最短路径及距离

if(!final[w] && (min + G.arcs[v][w] < D[w])){ //修改D[w]和P[w], w ∈ V-S

D[w] = min + G.arcs[v][w];

P[w] = P[v]; P[w][w] = TRUE; //P[w] = P[v] + P[w]

}//if }//for }//ShortestPath\_DIJ

## Dijkstra的算法实例

- 例如，图7.34所示有向网G6的带权邻接矩阵

$$\begin{bmatrix} \infty & \infty & 10 & \infty & 30 & 100 \\ \infty & \infty & 5 & \infty & \infty & \infty \\ \infty & \infty & \infty & 50 & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & 10 \\ \infty & \infty & \infty & 20 & \infty & 60 \\ \infty & \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

图7.34

若对G6实行Dijkstra算法，则所得从V0到其余各顶点的最短路径，以及运算过程中D向量的变化情况，如下所示：



## Dijkstra的算法实例

终点	从 V0 到各终点的 D 值和最短路径的求解过程				
	i=1	i=2	i=3	i=4	i=5
V1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$ 无
V2	10 (v0, v2)				
V3	$\infty$	60 (v0, v2, v3)	50 (v0, v4, v3)		
V4	30 (v0, v4)	30 (v0, v4)			
V5	100 (v0, v5)	100 (v0, v5)	90 (v0, v4, v5)	60 (v0, v4, v3, v5)	
Vj	V2	V4	V3	V5	
S	{v0, v2}	{v0, v2, v4}	{v0, v2, v3, v4}	{v0, v2, v3, v4, v5}	