

## HW10

教材 P183-184:

2. 应用题: (4)

3. 算法设计题: (3)、(5)

(4) 有向网如图 6.35 所示, 试用迪杰斯特拉算法求出从顶点  $a$  到其他各顶点间的最短路径, 完成表 6.9。

	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	1	0	1	0
2	0	0	1	0	0	0	1	0	0	0
3	0	0	0	1	0	0	0	1	0	0
4	0	0	0	0	1	0	0	0	1	0
5	0	0	0	0	0	1	0	0	0	1
6	1	1	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	1
8	1	0	0	1	0	0	0	0	1	0
9	0	0	0	0	1	0	1	0	0	1
10	1	0	0	0	0	1	0	0	0	0

图 6.34 邻接矩阵

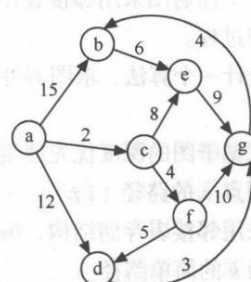


图 6.35 有向网

表 6.9

最短路径的求解

终点 \ D	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
b	15 (a,b)					
c	<u>2</u> (a,c)					
d	12 (a,d)					
e	$\infty$					
f	$\infty$					
g	$\infty$					
S 终点集	{a,c}					

### 3. 算法设计题

(1) 分别以邻接矩阵和邻接表作为存储结构, 实现以下图的基本操作:

- ① 增加一个新顶点  $v$ ,  $\text{InsertVex}(G, v)$ ;
- ② 删除顶点  $v$  及其相关的边,  $\text{DeleteVex}(G, v)$ ;
- ③ 增加一条边  $\langle v, w \rangle$ ,  $\text{InsertArc}(G, v, w)$ ;
- ④ 删除一条边  $\langle v, w \rangle$ ,  $\text{DeleteArc}(G, v, w)$ 。

(2) 一个连通图采用邻接表作为存储结构, 设计一个算法, 实现从顶点  $v$  出发的深度优先遍历的非递归过程。

(3) 设计一个算法, 求图  $G$  中距离顶点  $v$  的最短路径长度最大的一个顶点, 设  $v$  可达其余各个顶点。

(4) 试基于图的深度优先搜索策略写一算法, 判别以邻接表方式存储的有向图中是否存在由顶点  $v_i$  到顶点  $v_j$  的路径 ( $i \neq j$ )。

(5) 采用邻接表存储结构, 编写一个算法, 判别无向图中任意给定的两个顶点之间是否存在一条长度为  $k$  的简单路径。