

---

注意：

1. 请独立完成，杜绝抄袭；
2. 提交时间： 11-29。

## 第 6 章 图

### 1. 选择题

- (1) 在一个图中，所有顶点的度数之和等于图的边数的（ ）倍。  
A.  $1/2$                       B. 1                      C. 2                      D. 4
- (2) 在一个有向图中，所有顶点的入度之和等于所有顶点的出度之和的（ ）倍。  
A.  $1/2$                       B. 1                      C. 2                      D. 4
- (3) 具有  $n$  个顶点的有向图最多有（ ）条边。  
A.  $n$                       B.  $n(n-1)$                       C.  $n(n+1)$                       D.  $n^2$
- (4)  $n$  个顶点的连通图用邻接矩阵表示时，该矩阵至少有（ ）个非零元素。  
A.  $n$                       B.  $2(n-1)$                       C.  $n/2$                       D.  $n^2$
- (5)  $G$  是一个非连通无向图，共有 28 条边，则该图至少有（ ）个顶点。  
A. 7                      B. 8                      C. 9                      D. 10
- (6) 若从无向图的任意一个顶点出发进行一次深度优先搜索可以访问图中所有的顶点，则该图一定是（ ）图。  
A. 非连通                      B. 连通                      C. 强连通                      D. 有向
- (7) 下面（ ）算法适合构造一个稠密图  $G$  的最小生成树。  
A. Prim 算法                      B. Kruskal 算法                      C. Floyd 算法                      D. Dijkstra 算法
- (8) 用邻接表表示图进行广度优先遍历时，通常借助（ ）来实现算法。  
A. 栈                      B. 队列                      C. 树                      D. 图
- (9) 用邻接表表示图进行深度优先遍历时，通常借助（ ）来实现算法。  
A. 栈                      B. 队列                      C. 树                      D. 图
- (10) 深度优先遍历类似于二叉树的（ ）。  
A. 先序遍历                      B. 中序遍历                      C. 后序遍历                      D. 层次遍历

(11) 广度优先遍历类似于二叉树的 ( )。

- A. 先序遍历      B. 中序遍历      C. 后序遍历      D. 层次遍历

(12) 图的 BFS 生成树的树高比 DFS 生成树的树高 ( )。

- A. 小      B. 相等      C. 小或相等      D. 大或相等

(13) 已知图的邻接矩阵如图 6.1 所示, 则从顶点  $v_0$  出发按深度优先遍历的结果是 ( )。

$v_0$	0	1	1	1	1	0	1
$v_1$	1	0	0	1	0	0	1
$v_2$	1	0	0	0	1	0	0
$v_3$	1	1	0	0	1	1	0
$v_4$	1	0	1	1	0	1	0
$v_5$	0	0	0	1	1	0	1
$v_6$	1	1	0	0	0	1	0

- A. 0 2 4 3 1 5 6  
B. 0 1 3 6 5 4 2  
C. 0 1 3 4 2 5 6  
D. 0 3 6 1 5 4 2

图 6.1 邻接矩阵

(14) 已知图的邻接表如图 6.2 所示, 则从顶点  $v_0$  出发按广度优先遍历的结果是 ( ), 按深度优先遍历的结果是 ( )。

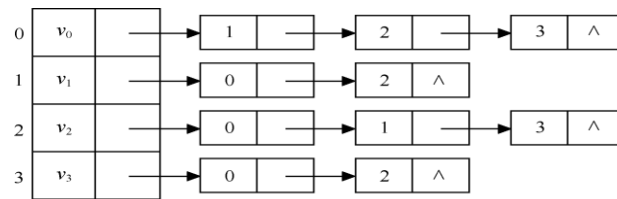


图 6.2 邻接表

- A. 0 1 3 2      B. 0 2 3 1      C. 0 3 2 1      D. 0 1 2 3

(15) 下面 ( ) 方法可以判断出一个有向图是否有环。

- A. 深度优先遍历      B. 拓扑排序      C. 求最短路径      D. 求关键路径

## 2. 应用题

(1) 已知图 6.3 所示的有向图, 请给出:

- ① 每个顶点的入度和出度;
- ② 邻接矩阵;
- ③ 邻接表;
- ④ 逆邻接表。

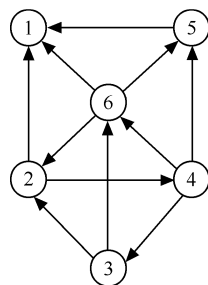


图 6.3 有向图

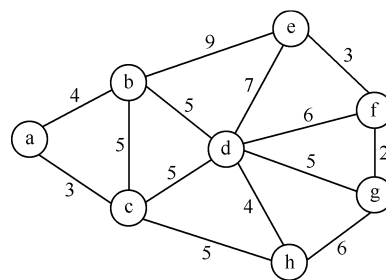


图 6.4 无向网

(2) 已知如图 6.4 所示的无向网，请给出：

- ① 邻接矩阵；
- ② 邻接表；
- ③ 最小生成树

(3) 已知图的邻接矩阵如图 6.5 所示。试分别画出自顶点 1 出发进行遍历所得的深度优先生成树和广度优先生成树。

(4) 有向网如图 6.6 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 6.1。

	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 6.5 邻接矩阵

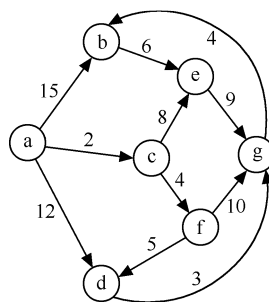


图 6.6 有向网

表 6.1

D 终点	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}					

- (5) 试对图 6.7 所示的 AOE-网：
- ① 求这个工程最早可能在什么时间结束；
  - ② 求每个活动的最早开始时间和最迟开始时间；
  - ③ 确定哪些活动是关键活动

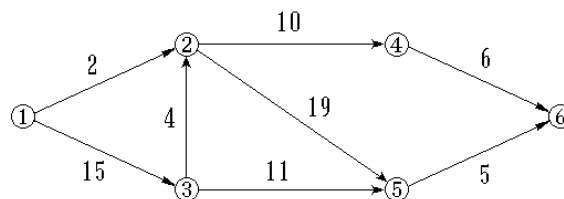


图 6.7 AOE-网

**3. 算法设计题**（请进行算法分析, 并写出相应的函数代码）

- (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) 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。