

DS Homework 11

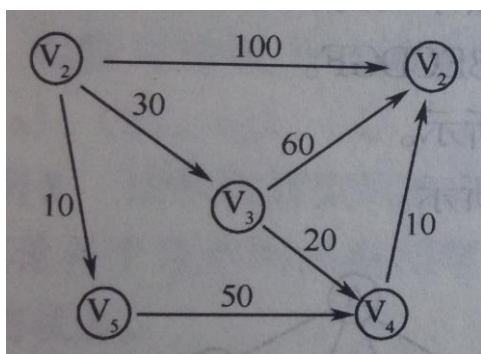
注：请使用 A4 纸作答，写上姓名学号，并于下一次上课时提交。

1、对有五个顶点{V1, V2, V3, V4, V5}的图的邻接矩阵如图所示：

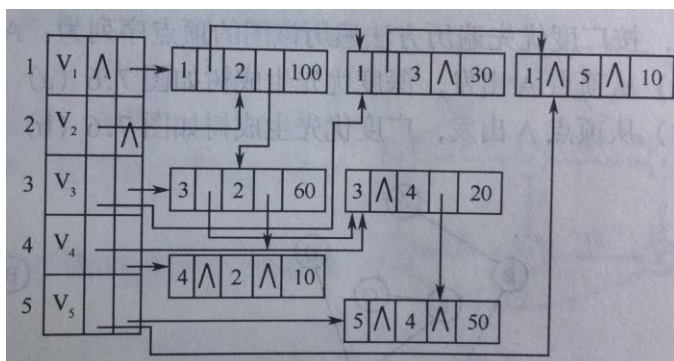
$$\begin{bmatrix} 0 & 100 & 30 & \infty & 10 \\ \infty & 0 & \infty & \infty & \infty \\ \infty & 60 & 0 & 20 & \infty \\ \infty & 10 & \infty & 0 & \infty \\ \infty & \infty & \infty & 50 & 0 \end{bmatrix}$$

解答下列问题：

(1) 画出逻辑结构图。



(2) 绘制出十字链表。



(3) 基于邻接矩阵写出图的深度和广度优先遍历序列。

以 V1 为源点，深度优先遍历为：V1, V2, V3, V4, V5；广度优先遍历为：

V1, V2, V3, V5, V4

2、已知无向图 G ，写一个算法判别 v_0 和 v_1 之间是否存在一条长度为 k 的路径。（提示：可利用深度优先遍历算法）

解：本题求解可利用深度优先遍历算法。设栈 S 中保存遍历路径上的顶点，并以 d 记下当前路径长度。从 $v = v_0$ 出发，找 v 的邻接点 w ，若 w 已访问过，则找下一个邻接顶点，若 $w = v_1$ 且 $d = k - 1$ ，则找到满足要求的路径；若 $w \neq v_1$ 且 $d < k - 1$ ，则从 w 出发继续遍历，否则退栈，找前一顶点的下一邻接点继续进行深度优先遍历。算法描述如下：

A1. [初始化栈 S 为空； $v \leftarrow v_0$ ； v 入栈； $d \leftarrow 0$ ； $found \leftarrow 0$ ；]

A2. 循环：当栈 S 不为空且 $found = 0$ 时，重复执行

[A2.1. 如果 v 不存在未被访问的邻接点，则 转(A2.6)；

A2.2. [$v \leftarrow v$ 的未被访问的邻接点； $d \leftarrow d + 1$ ；]

A2.3. 如果 $v = v_1$ 且 $d = k - 1$ ，则

[输出栈所示之路径； $found \leftarrow 1$ ；]

A2.4. 如果 $v = v_1$ 且 $d \neq k - 1$ ，则 转(A2.6)；

A2.5. [$push(S, v)$ ； 转(A2.8)；]

A2.6. $pop(S)$ ；

A2.7. $v \leftarrow$ 栈顶结点；

A2.8. 空操作；

]

A3. 如果 $found = 0$ ，则输出“顶点 v_1 到 v_2 之间不存在长度为 k 的路径”；

A4. 算法结束。