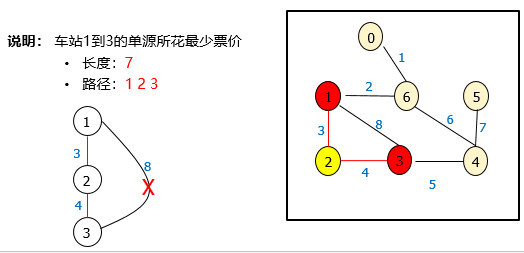
## **实验内容**

(1) 判断车站的路线图是否连通。从一个顶点开始遍历，如果能遍历到所有站点，那么图连通。

(2) 若图连通，使用Dijkstra算法计算从某个站点到另一站点的最小票价。如图所示，车站1到车站3的最小票价为7。



(3) 若图连通，计算图的直径（diameter），半径（radius）。定义如下：

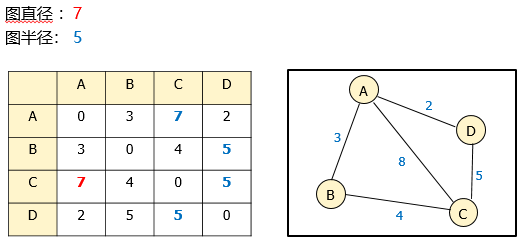
  节点距离：指的是两个节点间的最短路径的长度。

  Eccentricity：这个参数描述的是从任意一个节点，到达其他节点的最大距离

  Diameter：图中的最大的Eccentricity

  Radius：图中的最小的Eccentricity

  举例说明，如下图所示



  Eccentricity(A) = ABC = 7;

  Eccentricity(B) = BAD = 5;

  Eccentricity(C) = CBA = 7;

  Eccentricity(D) = DC = DAB = 5;

  所以半径是5，直径是7。

  思考题：Floyd算法如何输出路径？

## **输入输出格式**

  输入：

7 *// 节点数*8 *// 节点相邻边数*0 6 1 *////节点0到节点6有一条长度为1的边*1 6 21 2 32 3 43 4 54 6 64 5 71 3 8

  输出：

connected: 1

the minimum fare between 1 and 3: 7

Path: 1->2->3

diameter:16

radius:9

## **实验要求**

（1）请完成平台[http://10.249.176.82:9000/上的题，编号3002](http://10.249.176.82:9000/%E4%B8%8A%E7%9A%84%E9%A2%98%EF%BC%8C%E7%BC%96%E5%8F%B73002)

（2）请根据平台上的题目要求，提交本地测试通过的代码。

（3）本地提供的用例并未覆盖所有功能和边界条件，请务必自行设计用例全面测试代码后，再提交平台测试。）