**实验三、Pagerank算法和六度空间**

输入：有向图，n\*n的邻接矩阵

输出：

1. 实现pagerank算法，算出每个节点的pagerank值
2. 计算符合“六度空间”理论的结点占结点总数的百分比。（可以将图改为无向图）

相关概念：

实验可以考虑随机生成的图以及符合互联网网页链接特征的图（自行设计算法生成）

算法分析:

对PageRank做k次更新操作，每次使用如下规则：

1. 每个节点将自己当前的PageRank值通过出向链接均分传递给所指向的节点。（若没有出向链接，则认为传递给自己，或者说，保留）
2. 每个节点以从入向链接获得的（包括可能自传的）所有值之和更新它的PageRank。

代码实现(java):

import java.util.Random;  
public class Experiment3\_page\_rank {  
 int linjiejuzhen[][];//二维数组存放邻接矩阵  
 int chubian[];//记录每个点的出边数目  
 double rank[];//记录每个点的rank值  
 int size;//矩阵的边长 //打印有向邻接矩阵  
 public Experiment3\_page\_rank(int n)  
 {  
 init(n);  
 print();  
 countchubian(); }  
  
 public void print()  
 {  
 for(int i=0;i<size;i++)  
 {  
 for(int j=0;j<size;j++)  
 {  
 System.*out*.print(linjiejuzhen[i][j]+" "); }  
 System.*out*.println(); } }  
  
 public void init(int n)//【i】【j】为0表示无边，为1表示i指向j  
 { //随机生成有向图  
 size=n;  
 this.linjiejuzhen=new int[size][size];  
 Random rand = new Random();  
 for(int i=0;i<size;i++)  
 for(int j=0;j<size;j++)  
 {  
 if(i==j)  
 linjiejuzhen[i][j]=0;  
 else  
 linjiejuzhen[i][j]=rand.nextInt(2);  
 }  
 //初始化每个节点的rank值  
 rank=new double[size];  
 for(int i=0;i<size;i++)  
 rank[i]=1.0/size;  
 }  
  
 public void countchubian()//统计每个节点有几个出边  
 {  
 chubian=new int[size];  
 for(int i=0;i<size;i++)  
 {  
 chubian[i]=0;  
 }  
  
 for(int i=0;i<size;i++)  
 for(int j=0;j<size;j++)  
 {  
 if(i!=j&&linjiejuzhen[i][j]==1)  
 chubian[i]++;  
 }  
 }  
  
 public void rank()  
 {  
 for(int i=0;i<size;i++)//遍历所有节点  
 {  
 for(int j=0;j<size;j++)//找出指向当前节点的所有节点，加上rank值  
 {  
 if(i!=j&&linjiejuzhen[j][i]==1)  
 {  
 rank[i]+=rank[j]/chubian[j];  
 }  
 }  
 }  
 }  
  
 public void printChubian()  
 {  
 String r="";  
 for(int i=1;i<size+1;i++)  
 r+=i+":"+chubian[i-1]+" ";  
 System.*out*.println("节点出边"+"\n"+r);  
 }  
  
 public void printRank()  
 {  
 String r="";  
 for(int i=1;i<size+1;i++)  
 r+=i+":"+rank[i-1]+" ";  
 System.*out*.println("节点rank值"+"\n"+r);  
 }  
  
  
 public static void main(String[] args) { // *TODO Auto-generated method stub* Experiment3\_page\_rank temp=new Experiment3\_page\_rank(4);  
 temp.printRank();  
 temp.printChubian();  
 for(int i=0;i<5;i++)  
 {  
 System.*out*.println("第"+(i+1)+"次rank");  
 temp.rank();  
 temp.printRank();  
 }  
 }  
}

运行结果：

