Practical No 2

**Name :** Rahul Sampatrao Patil

**Roll No :** 4176

# Code ( java file ):-

import java.util.\*; import java.io.\*;

public class two\_PageRank {

public int path[][] = new int[10][10];

public double pagerank[] = new double[10]; public void calc(double totalNodes){

double InitialPageRank; double OutgoingLinks=0; double DampingFactor = 0.85;

double TempPageRank[] = new double[10]; int ExternalNodeNumber;

int InternalNodeNumber; int k=1; // For Traversing int ITERATION\_STEP=1;

InitialPageRank = 1/totalNodes;

System.out.printf(" Total Number of Nodes :"+totalNodes+"\t Initial PageRank of All Nodes

:"+InitialPageRank+"\n"); for(k=1;k<=totalNodes;k++)

{

this.pagerank[k]=InitialPageRank;

}

System.out.printf("\n Initial PageRank Values , 0th Step \n"); for(k=1;k<=totalNodes;k++)

{

System.out.printf(" Page Rank of "+k+" is :\t"+this.pagerank[k]+"\n");

}

while(ITERATION\_STEP<=2) // Iterations

{

for(k=1;k<=totalNodes;k++)

{

TempPageRank[k]=this.pagerank[k]; this.pagerank[k]=0;

}

for(InternalNodeNumber=1;InternalNodeNumber<=totalNodes;InternalNodeNumber++)

{

for(ExternalNodeNumber=1;ExternalNodeNumber<=totalNodes;ExternalNodeNumber++)

{

if(this.path[ExternalNodeNumber][InternalNodeNumber] == 1)

{ k=1;

OutgoingLinks=0; // Count the Number of Outgoing Links for each ExternalNodeNumber while(k<=totalNodes)

{

if(this.path[ExternalNodeNumber][k] == 1 )

{

OutgoingLinks=OutgoingLinks+1; // Counter for Outgoing Links

}

k=k+1;

}

this.pagerank[InternalNodeNumber]+=TempPageRank[ExternalNodeNumber]\*(1/OutgoingL inks);

}

}

}

System.out.printf("\n After "+ITERATION\_STEP+"th Step \n"); for(k=1;k<=totalNodes;k++)

System.out.printf(" Page Rank of "+k+" is :\t"+this.pagerank[k]+"\n"); ITERATION\_STEP = ITERATION\_STEP+1;

}

for(k=1;k<=totalNodes;k++)

{

this.pagerank[k]=(1-DampingFactor)+ DampingFactor\*this.pagerank[k];

}

System.out.printf("\n Final Page Rank : \n"); for(k=1;k<=totalNodes;k++)

{

System.out.printf(" Page Rank of "+k+" is :\t"+this.pagerank[k]+"\n");

}

}

public static void main(String args[])

{

int nodes,i,j,cost;

Scanner in = new Scanner(System.in); System.out.println("Enter the Number of WebPages \n"); nodes = in.nextInt();

two\_PageRank p = new two\_PageRank();

System.out.println("Enter the Adjacency Matrix with 1->PATH & 0->NO PATH Between two WebPages: \n");

for(i=1;i<=nodes;i++) for(j=1;j<=nodes;j++)

{

p.path[i][j]=in.nextInt(); if(j==i)

p.path[i][j]=0;

}

p.calc(nodes);

}

}

# Output :-



