Kia Babashahi Ashtiani 40043898

Important information on the code and the analysis:

In order to read the input, you need to write the directory of your code in line 9 in the Main class. You can also set the scaling factor, the count (the maximum number of repeat) and the epsilon (The tolerance value) in lines 10-12. The program creates a file called output.txt which has the page ranks in the format you desired.

Note 1: The tables in this analysis consist of five parts: the first part is the time it takes to create the network which is presented in Nano-seconds. The second part is the convergence time which is calculated in milli seconds and since the number was small I repeated the experiment for 100,000 times for all cases and wrote down the results. The third part is the scaling factor the fourth part is the number of vertices and the last part is the number of edges.

Note 2 : For the test cases I used random graphs and the generation of test cases 6,7,8 9 was as following: after generating the tree I selected a random node and linked it to all other nodes and repeated this process in order to get a denser graph. I used the edges of one graph in the test case before of it and added more links to it in order to be able to make a better analysis. Since I wanted to add edges to the same graph to check the effect of the type of the graph which started with a sparse tree and reached a dense graph on a graph with the same amount of Nodes.

Note 3:The times presented in this analysis are based on the Machine I am working on so based on the machine the number might change but the ratios stay the same.

1-The effect of the Network size (the Number of Nodes):

In order to do this analysis, we need to fix the damping factor and the number of edges and also remove the maximum number of iterations in the code (you can also set it to infinity). I sat the damping factor to 0.85 However, since the network is connected, as the number of vertices grows so does the number of edges. Therefore I considered graphs with a certain structure such that the number of edges does not play an important role. For this means I created a tree with n vertices and n-1 edges to analyze the effect of the number of Nodes or the network size on the running time. As you can see in the table below the running time of page rank calculation is directly affected by the network size, meaning that as the network size grows the page rank increases. The change is significant in test case five in which the number of Nodes is doubled.

	Creating the network	Convergence time	Scaling factor (damping)	#vertices	#Edges
Test case 3	466419	31	0.85	6	5
Test case 4	452451	36	0.85	10	9
test case 5	517738	150	0.85	21	20

2-The effect of the Scaling Factor (Damping Factor):

For the sake of this analysis, I removed the Maximum Number of repeats condition and I also did the analysis on test case 8, a graph with 10 vertices and 40 edges (a Graph which is neither sparse nor has a lot of loops or is too dense.)

As you can see as the scaling factor is decreased the Convergence time is also decreased.

	Creating the network	Convergence time	Scaling factor (damping)	#vertices	#Edges
Test case data 8	573617	130	0.90	10	40
Test case data 8	573617	120	0.50	10	40
Test case data 8	573617	100	0.10	10	40

3- The effect of the type of the input graph on

1- The **distribution** of the page ranks
2- The convergence **time**

For this experiment, I used test cases 6, 7, 8 and 9. The number of Vertices and the scaling factor are fixed and the Maximum number of repeats condition is also removed. I am analyzing four graphs , one which is a tree(6) and is sparse another one which is still a sparse graph but has more number of edges(7) and a more dense graph with 40 edges and finally a graph with 66 edges which is much more dense and has a lot of loops. The graph are generated randomly with the method described in the first part of the analysis. What I observed was the following:

1- Analysis of the time: As the number of Edges increased so did the convergence time of the algorithm and the denser our graph became the more time it takes the algorithm to converge.
2- Analysis of the **distribution** part A: As the number of edges is increased specially with the increase in the number of loops and loops on important nodes who have a link to other important nodes(like node e in test case 8 which has a Page rank of 1.03) we are seeing that some of the pages might have higher values than the sparse cases(in other words we see a lot of edges with page rank 0.9 or even more than 1 in the dense graphs (depending on what we set our scaling factor to be) which is not happening in the sparse case. In the sparse cases we are usually dealing with small numbers mostly less than 0.6.
3- Analysis of the distribution part B: Another significant effect is the change in the gap between the page ranks. For the sparse cases it happened from time to time that a page let's say page 'a' in test case 6 had rank 0.15 and page j had rank of 0.80 but as the number of edges is increased the gap is removed and the numbers become closer to each other so the distribution is becoming more and more uniform. As you can see in the list below which belongs to test case 9 :

Page Ranks of the pages of test case 10 with a damping factor of 0.85
www.a.com 0.5519775060797152
www.b.com 0.6041087149872439
www.c.com 1.117601122726401
www.d.com 0.5048701718357744
www.e.com 0.9340098178961825
www.f.com 0.5519775060797152
www.g.com 0.6041087149872439
www.h.com 0.5519775060797152
www.i.com 0.5519775060797152
www.j.com 0.6041087149872439

As the number of edges is increased the numbers are becoming in the same range.

	Creating the network	Convergence time	Scaling factor (damping)	#vertices	#Edges
Test case data 6	468131	39	0.85	10	9
Test case data 7	434489	110	0.85	10	24
Test case data 8	573617	118	0.85	10	40
Test case data 9	500632	140	0.85	10	66

Test cases 1, 2 provided by you:

Below you will find the data associated with test cases one and two.

	Creating the network	Convergence time	Scaling factor (damping)	#vertices	#Edges
Test case 1	446178	39	0.85	5	7
Test case 2	573617	36	0.65	5	7

Conclusion:

The convergence time of the algorithm is correlated to the number of edges and the nodes in the graph so as the size of the network increases the so does the convergence time. The damping factor plays also an important role both in the value of the page ranks also in the convergence time so, as the damping factor increases so does the convergence time. And as we saw the denser the graph becomes the more uniform the distribution of the page ranks becomes.