Information Institute of Technology

Department of computer

(B.Eng.) in Software Engineering

Module: 5SENG001W Algorithms

Module Leader: Klaus Draeger

CourseWork 1

Date of submission: 31st of March, 2021

Student ID: 20191161

Student UOW: 17904023/1

Student First Name: Fathima Rinoza

Student Surname: Mohamed Jiffry

01) The task in this class is to find a flow network's MaxFlow. To determine that the FordFulkerson Algorithm is being used. To do so, we must first select the path from a given source node to a given destination. There are two path-finding algorithms to choose from.

a. BFS (Breadth First Search)
b. DFS (Depth First Search)

The BFS algorithms are used in this code to find the paths.

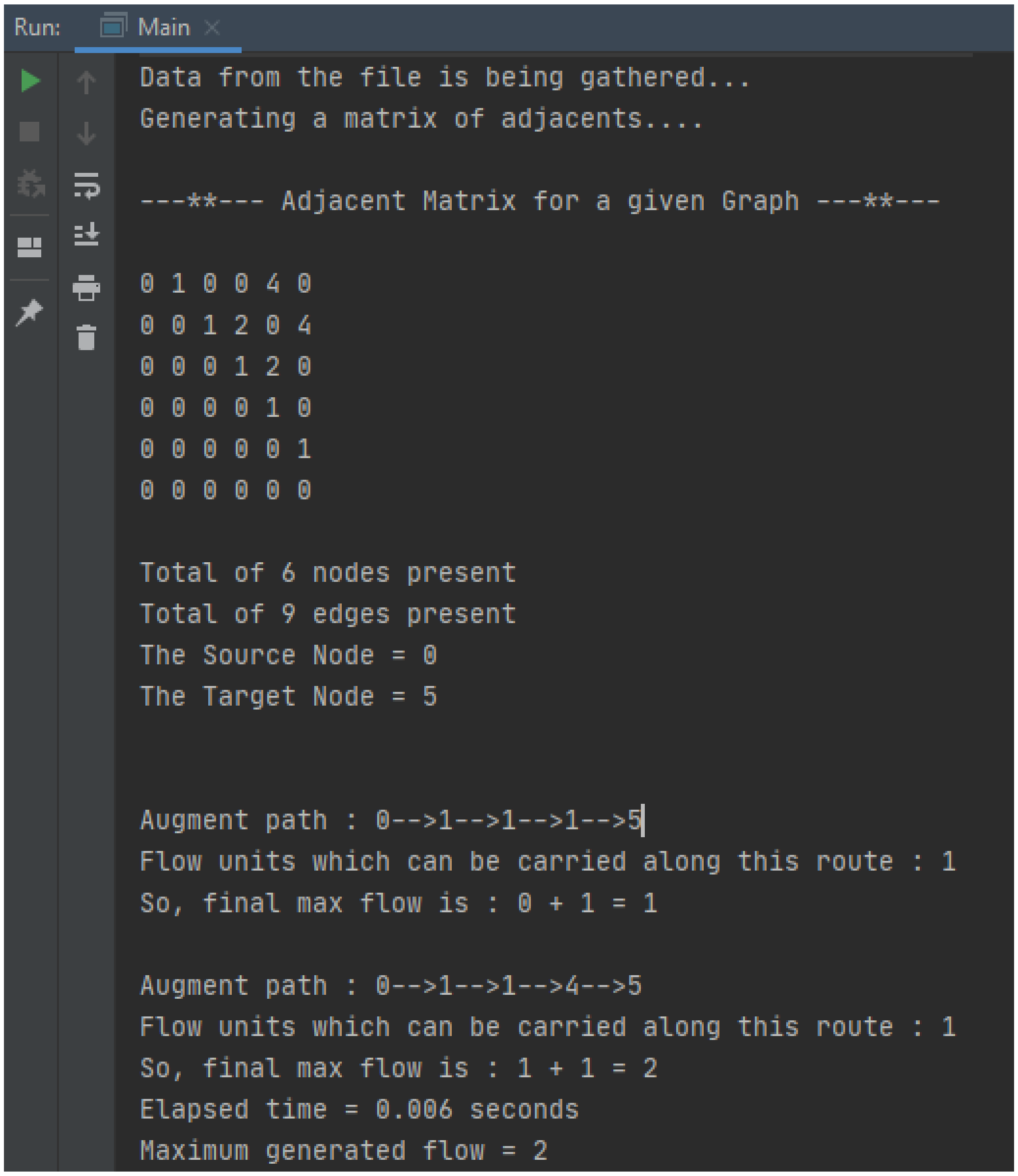
a) Data Structure

Two data structures were used in the BFS algorithm. Array and queue. Then it creates a queue, adds source vertex to it, and marks it as visited. Then it visits all of the unvisited vertex adjacent to it, marks it as visitesd, displays it, and adds it to the queue. It goes back and forth in the queue. Remove the first vertex from the queue if no adjacent vertex can be found. The process will be repeated until the queue is empty. The Ford Fulkerson Algorithm then uses a 2D array to represent the matrix. The nodes and edges of the residual graph are stored in a 2D matrix. In addition, an array has been created to store the matrix of visited paths. The vertices in the path are saved using the parent array.

b) Algorithm Strategy

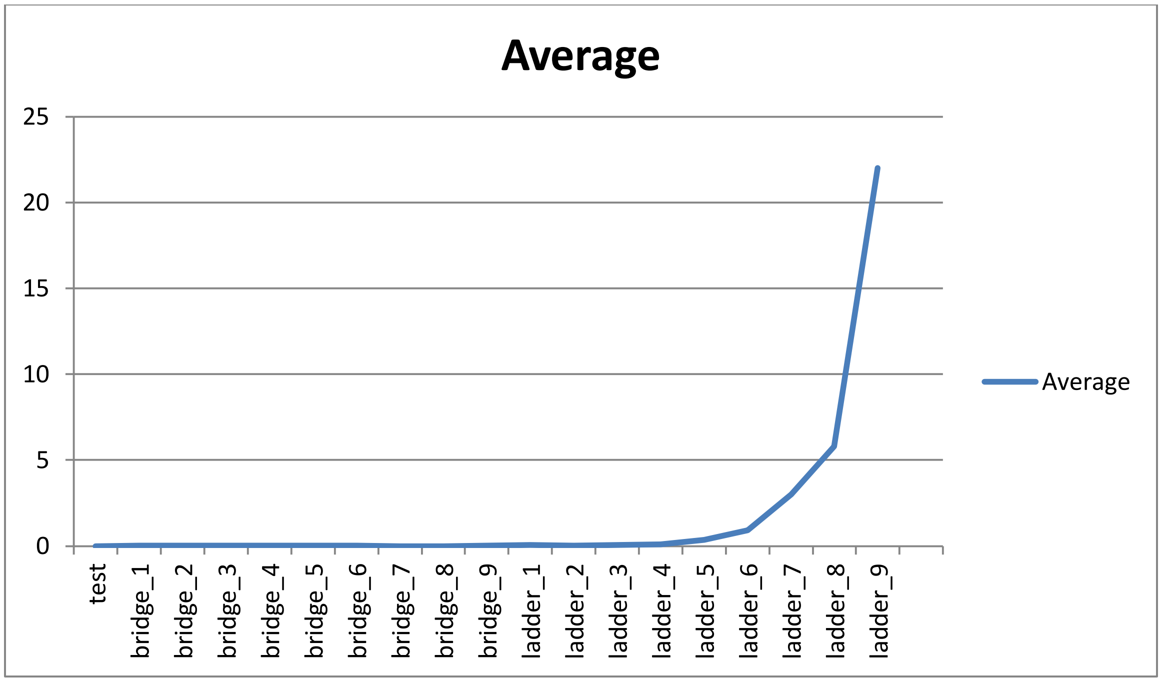
The BFS Algorithm is the best choice for finding the shortest path from a given node to a given destination. The final state will be reached in a small number of steps. This works on a level-by-level basis. The Queue Data Structure is used by BFS algorithms, while the Stack Data Structure is used by DFS algorithms. The BFS algorithm is a vertex-based algorithm. The size of the vertex has been changed in this code to produce four different data sets. As a result, the BFS is more appropriate than the DFS in this situation.

02)



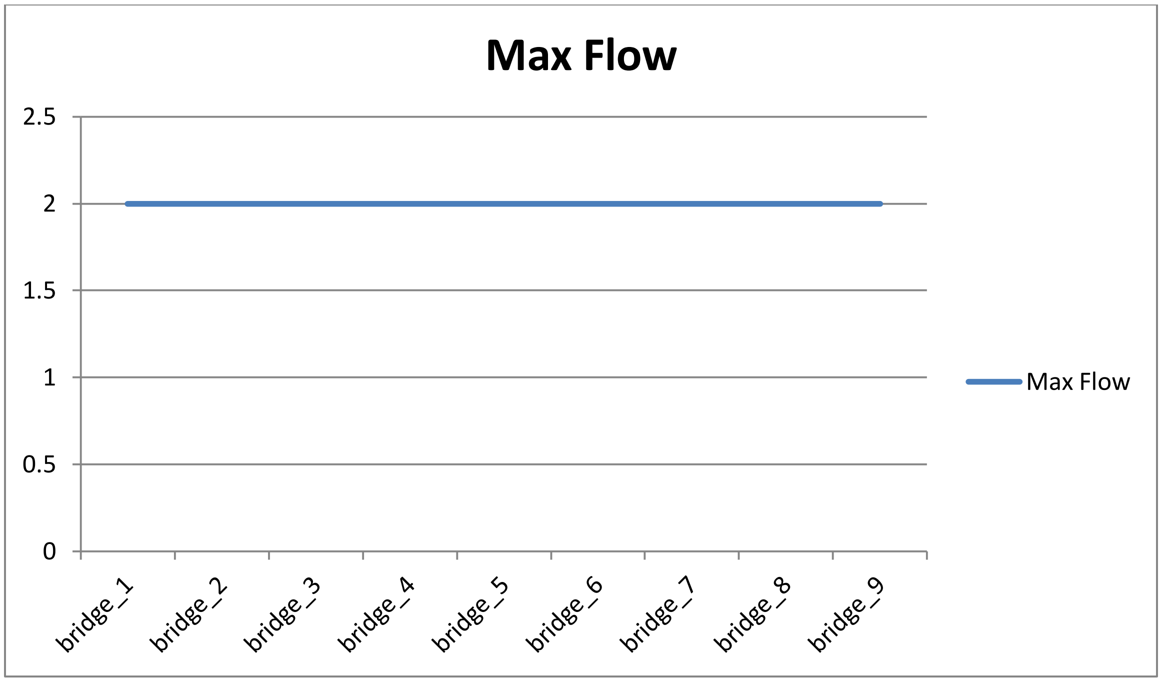
```
Data from the file is being gathered...
Generating a matrix of adjacents....

---**--- Adjacent Matrix for a given Graph ---**---

0 1 0 0 4 0
0 0 1 2 0 4
0 0 0 1 2 0
0 0 0 0 1 0
0 0 0 0 0 1
0 0 0 0 0 0

Total of 6 nodes present
Total of 9 edges present
The Source Node = 0
The Target Node = 5


Augment path : 0-->1-->1-->1-->5
Flow units which can be carried along this route : 1
So, final max flow is : 0 + 1 = 1

Augment path : 0-->1-->1-->4-->5
Flow units which can be carried along this route : 1
So, final max flow is : 1 + 1 = 2
Elapsed time = 0.006 seconds
Maximum generated flow = 2
```

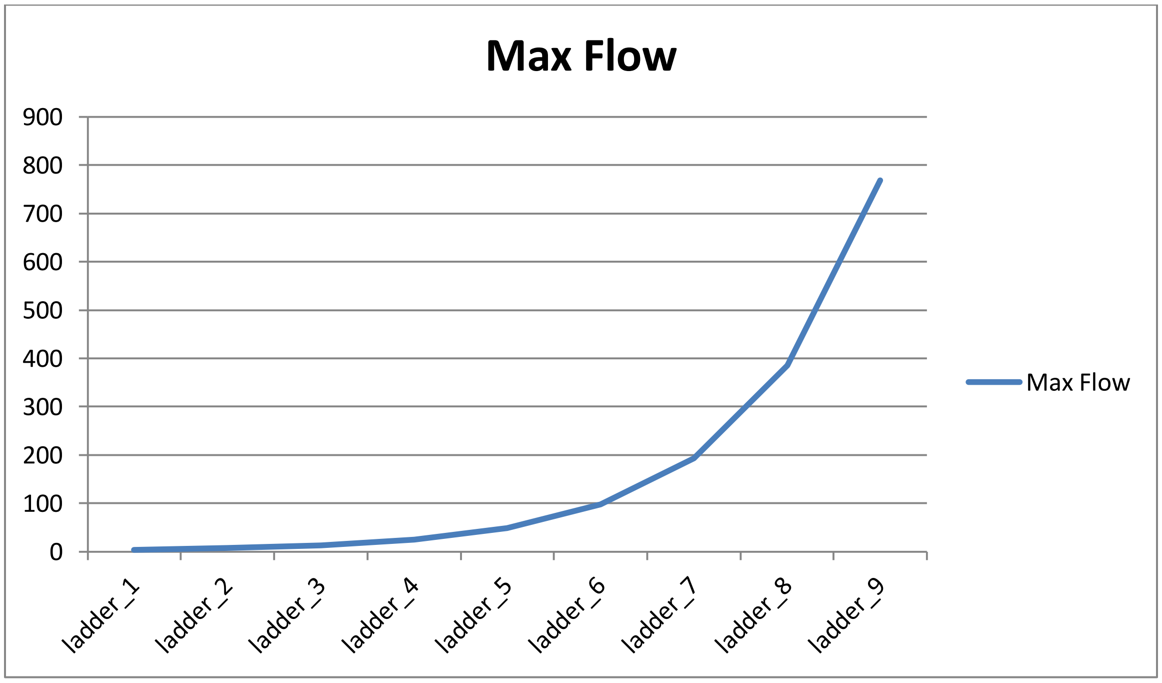
03) When we change the input files, the execution time changes, a shown in the outputs. From the smallest to the largest file, the average time changes accordingly.



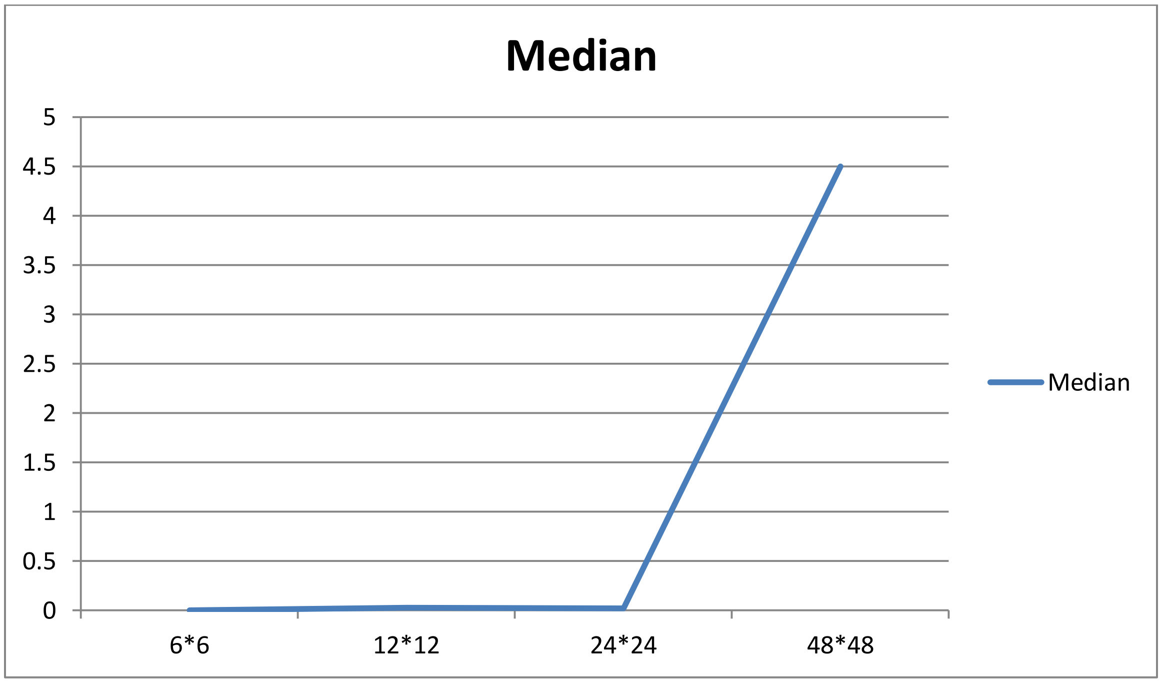
Since the graph is linear from bridge 1 to bridge 9, its Big-O notation is O. (n). And since the graph is quadratic from ladder 1 to ladder 9, its Big-O notation is O. (n^2).

File Name	Max Flow	Average Time	Input Files	Ratio
test	8	0.00067	4	-
Bridge_1	2	0.0016	6	2.3880
Bridge_2	2	0.006	10	3.75
Bridge_3	2	0.003	18	0.5
Bridge_4	2	0.0026	34	0.867
Bridge_5	2	0.029	66	11.154
Bridge_6	2	0.0096	130	0.3310
Bridge_7	2	0.00067	258	0.0698
Bridge_8	2	0.00067	514	1
Bridge_9	2	0.027	1026	40.299
Ladder_1	4	0.073	6	2.7037
Ladder_2	7	0.01	12	0.137
Ladder_3	13	0.066	24	6.6
Ladder_4	25	0.097	48	1.4697
Ladder_5	49	0.343	96	3.5360
Ladder_6	97	0.928	192	2.7055
Ladder_7	193	3.0073	384	3.2406
Ladder_8	385	5.806	768	1.9306
Ladder_9	769	21.995	1536	3.7883


Max Flow
2.5
2
1.5
1
0.5
0
bridge_1
bridge_2
bridge_3
bridge_4
bridge_5
bridge_6
bridge_7
bridge_8
bridge_9
Max Flow


Max Flow
900
800
700
600
500
400
300
200
100
0
ladder_1
ladder_2
ladder_3
ladder_4
ladder_5
ladder_6
ladder_7
ladder_8
ladder_9
Max Flow

Data Set	Time	Median	Ratio
6*6	(0.003+0.004+0.005+0.003+0.005)/5	0.004	
12*12	(0.048+0.011+0.018+0.031+0.04)/5	0.0296	7.4
24*24	(0.02+0.015+0.03+0.022+0.019)/5	0.0212	0.71
48*48	(0.084+0.255+0.036+0.076+0.26)/5	0.1422	6.70


Median
6*6
12*12
24*24
48*48
Median