Algorithm Coursework 30-03-2020

Mohamed Farook Mohamed Sajidh

Westminster ID: W17154204

IIT ID: 2018851

Contents

[Algorithmic Strategy 3](#_Toc36447302)

[Data Structure & Traversal 3](#_Toc36447303)

[Analysing the Algorithm 3](#_Toc36447304)

[Conclusion 3](#_Toc36447305)

# Algorithmic Strategy, Data Structure & Traversal

Ford-Fulker

# Pseudocode: Ford-Fulkerson Algorithm

1. Start with initial flow as 0
2. While there is an augmenting path from source to sink
   1. Add this path-flow to flow
3. Return flow

# Analysing the Algorithm

## Time Complexity

The time complexity of the above algorithm is O(max\_flow\*E). We run a loop while there is an augmenting path. In worst case, we may add 1-unit flow in every iteration. Therefore, the time complexity becomes O(max\_flow\*E)

# Conclusion

# References

* <https://www.baeldung.com/java-graphs>
* <https://www.geeksforgeeks.org/multidimensional-arrays-in-java/>
* <https://www.geeksforgeeks.org/ford-fulkerson-algorithm-for-maximum-flow-problem/>
* <https://www.geeksforgeeks.org/max-flow-problem-introduction/>
* <https://www.youtube.com/watch?v=Xu8jjJnwvxE>
* <https://www.geeksforgeeks.org/java-util-linkedlist-poll-pollfirst-polllast-examples-java/>
* <https://stackoverflow.com/questions/24108417/simple-way-of-creating-a-2d-array-with-random-numbers-python>