**Lab 4 Instructions**

You will be solving the same problem as that of the last week. You only need to add a suitable heuristic function. So, each time you generate a child node, you will have to calculate the heuristic value of that node. You will add that node in the list of nodes to be expanded (i.e. frontier / fringe) in sorted order. Design your own heuristic function. On output, note the total number of nodes visited.

Consider a variant of the 8 – puzzle. There are 8 tiles in a 3x3 grid. Out of 8 tiles, 4 tiles have ‘1’ written on it and the remaining 4 have ‘2’ written on it. There is a blank. Initially, the 1’s and 2’s are in arbitrary positions. The goal state is that starting from upper left corner, all the 1’s should come first, then all the 2’s and the blank should come at the end. An example is given below:

Initial State

Goal State

The rules for movement is the same as in the 8 – puzzle i.e. the blank (‘b’) can be swapped with any of the neighboring tiles in the NEWS directions.

Write a program that solves the above using BFS and DFS. Your program should take an arbitrary initial state. On termination it should print the complete path from the initial to goal state (i.e. all the sequence of states) as well as the total number of states visited before reaching the goal state.

Upload your program and a sample output on Moodle. Also, show your work, even if it is partial, during the lab.