## Task 1 (a):

First we create an empty redge list where we stoom stored row, col. and volve as a tuple from input tile and their creat create an empty matrix and for spacific from column we put the value on the natrix.

## Task 1 (1):

Same as previous, we create edge list and then ore create the dictionary considering key as number of vortices from o to N+1 and then are iterate the through the edge list and pit the column value at its rows keys as considering tuple which are livelies of the dictionary.

And rectain the adjacency Listing to the dictionary.

Firestly, we created a visited , list and initially make them all false as the keys are the vertices. From the input file we create a graph dictionary where keys are vertices and are lidirectional read. Now, We create a queae and from queue we pop and item and checked its children not visited one visited and make visited endrationaries value as it's vertices as true and append in queue and same paragain until the graph get to all withted and it will be level by levelii consories It motor but

For BES, I'll I'm an worked of state of For DFS, we do the some las previous but shore we will we stock and while g BFS we took a vertices ( and will / go until , there are ho commection and ilalli I and how I we bock track the wertices and elwill leheck of any unvirited vertices are left off, of the well do some Afsin some all hi has somes his The defference is of BFS and DFS is, in dfs we go into the last vertices and then lacktrack where in BFS we sheek ansis ell line ou similared la color children and from children to it grand reserve how stock out miles talso. children.

Fore eyele ditection, me will use IFS and as if goes into the entast of voitices. 19 As the 78 previous PFs dode, how muil modity was he create Ha parent stor dictamony where key are vertices and values broth initially More writing, some previous là me ellet mode fret visited ment winter it and call offs the will the will theck it adjecence and if the source node is not the parent's Padj and its already in Stock that means here we found the " cycle. Otherwise we visit the source and Land to the southings must be nouthings remove from Hock and return False. mildren.

We person as shows with the For shorted path we consider DFs as a graph traversal. From BFS code here we created a groene and put the stort and end volue as pain and create a set of visited value. Wer traversing, it counts name of diamonds. the gruene and and to Now, the eddenice from the queue and new ton the shristed meditt. Now your itereste to through the neighbor of every node of the graph, if not visited yet me will append in queue as First value is the neighbor and a light of nodes Start and end that volues and added to the Vilited set. If we bund node is some as the end which is provided then will redun the path value, and the time will be the ben length of the list - 1.

Firstly, create a worray where each rows stone ... Now, we use off to explore the and starting trom each empty cell (.) and recursively treaversing its neighbore cells ists up, down, left, reight wise. When treavering, it counts num. of diamonds while not visited un pass through obstacles (#). maxof loss han every odd diamond count the all acornors possible standing position and I country it was to modepear set not visited yet me will append in quene as First value is the neighbor and a libbol moder sold and and there value and added to his visited let. It we bond made Is some is the end which is provided the will 1000-- the perfect value. and the 1100 100 him length of the 11st - 1.