Evidence of Testing and Function brief

Qualitative Description of the Principal Functions used in the

Program:

goalstate():

* Returns True if the received state is the goal state.
* Returns False if the received state is not the goal state.

generatestates():

* Generates the successive nodes for the inputted string.
* Returns the list of successive nodes to caller.

create\_adj\_list():

* Creates the complete tree from the parent dictionary and returns back the resultant dictionary.

bfs():

* performs breadth first search by using the levels of the tree that is already generated
* It checks for every node that if goal state is found,then the algorithm stops and print the goal state.

dfs():

* performs depth first search by taking given state as input and checks whether the state is goal state.
* If the state is goal state,then it stops.

movegen() :

* function used by heuristic search methods for taking up the successors of the given node.
* Returns the list of successor nodes.

Append():

* Takes two lists as input and merge them into one list and return the new list.

bestfs():

* Takes the string as input and compares the heuristic values and defines the path to achieve the goal state.
* If goal state is achieved it stops.
* It returns back the list of the paths it traversed.

generatestates\_heuristic():

* Generate successive nodes for the heuristic searches especially for best first search.

hill\_climbing():

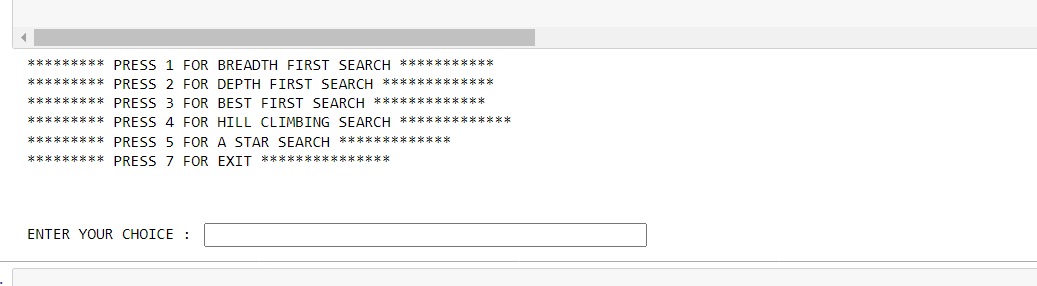
* Performs hill climbing based on heuristic values.if goal state is reached,it stops.
* It returns back the path it traversed.

astar():

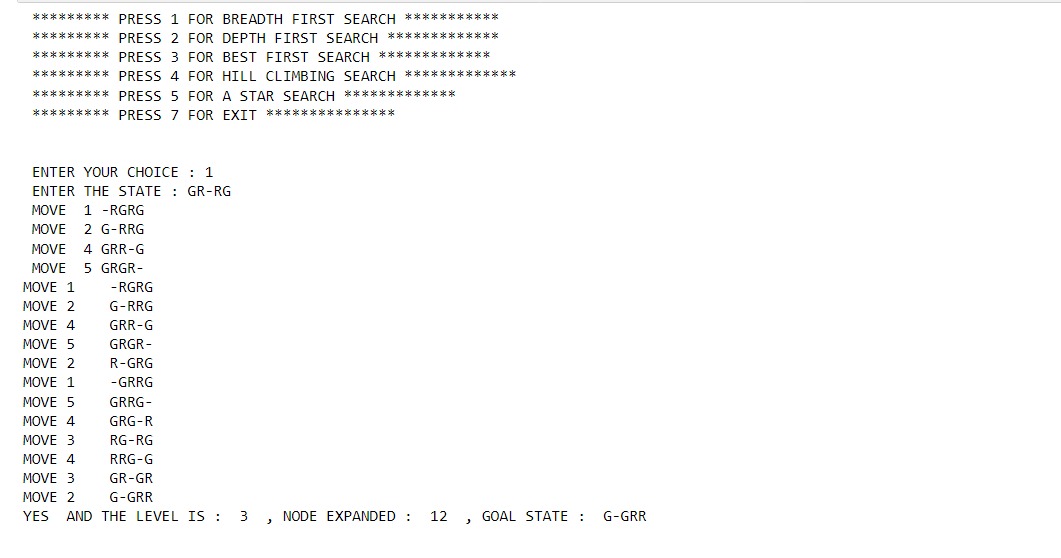
* Takes the state as input argument and performs the move operation based on pathcost and heuristic values.
* Returns the best path,it traversed.

EVIDENCE OF TESTING :

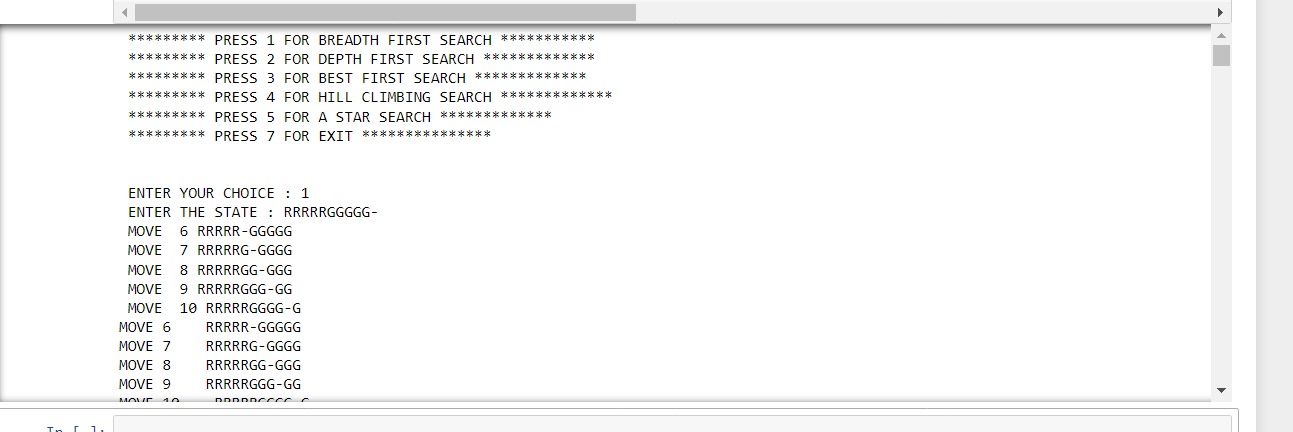
Initial Display for selecting the algorithm:

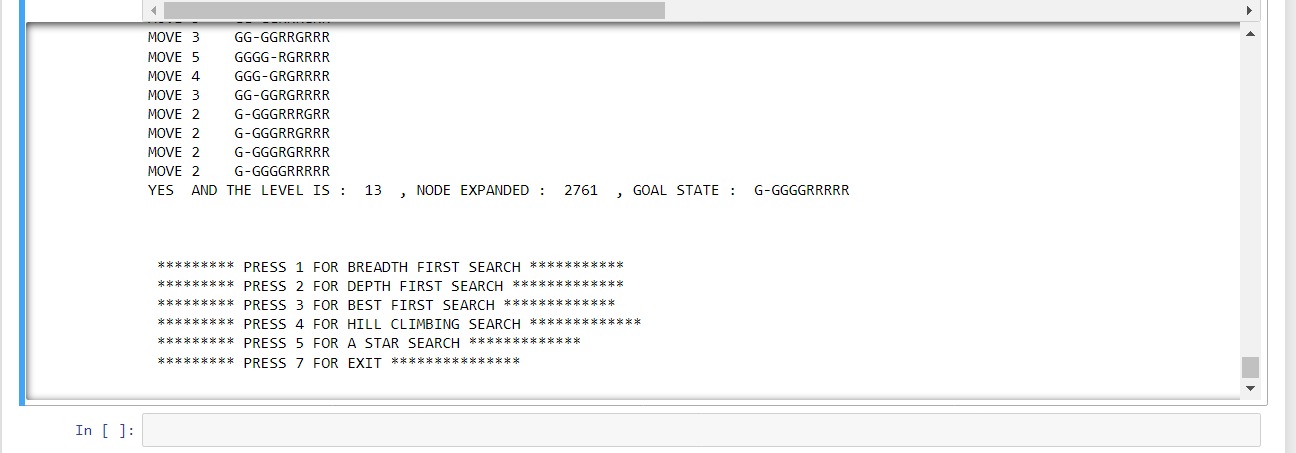


Selecting BFS and entering the state:

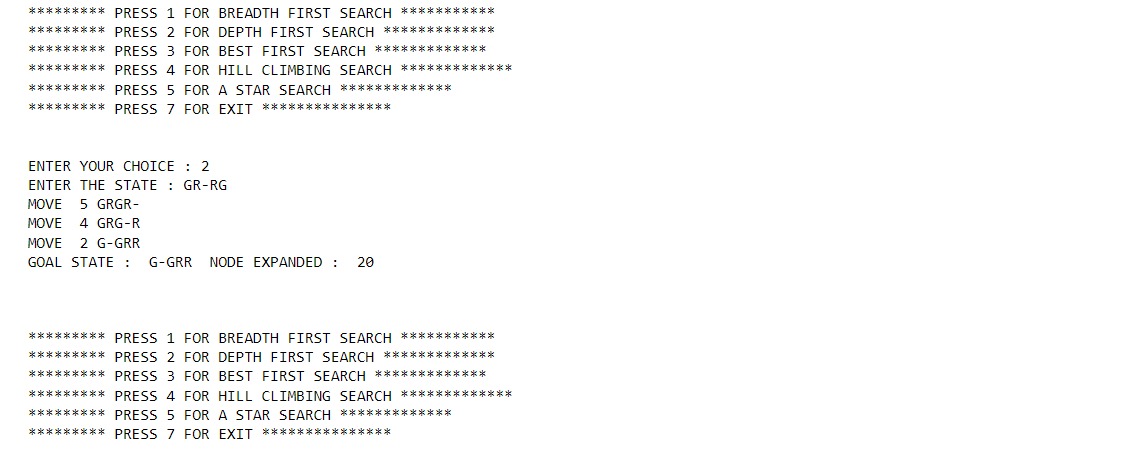


Selecting BFS and choosing 10 Tiles as input:

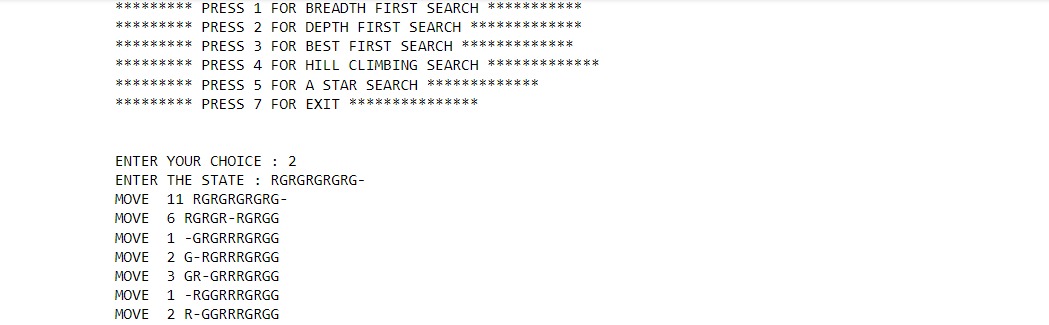


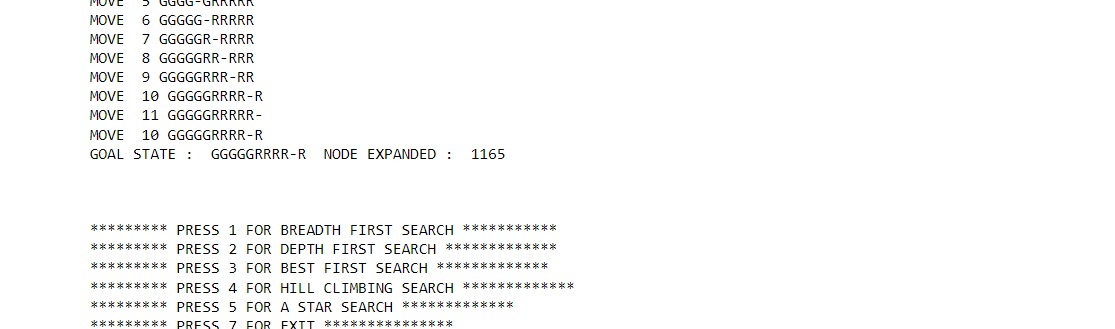


Selecting DFS and input 4 Tiles as input:

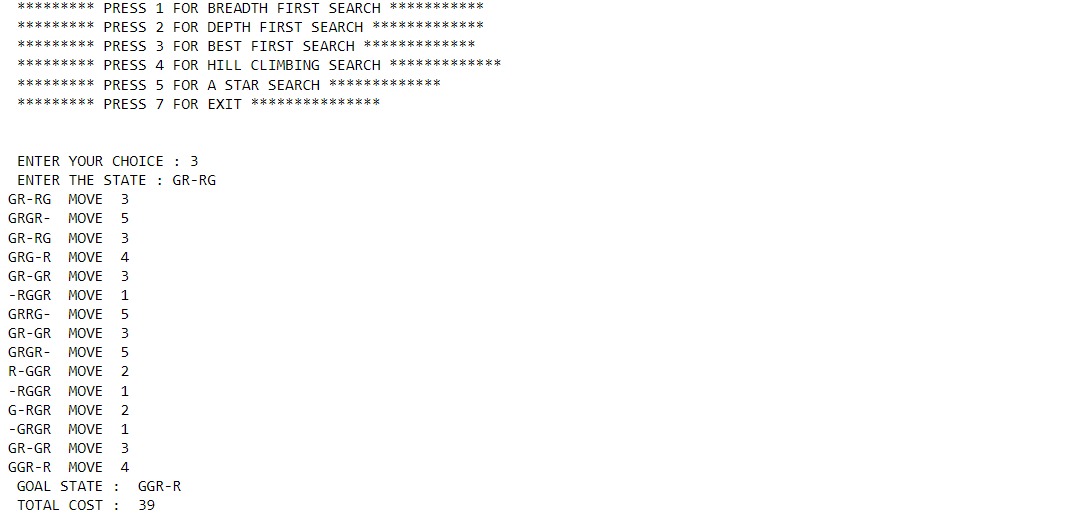


Selecting DFS and input 10 Tiles as Input:

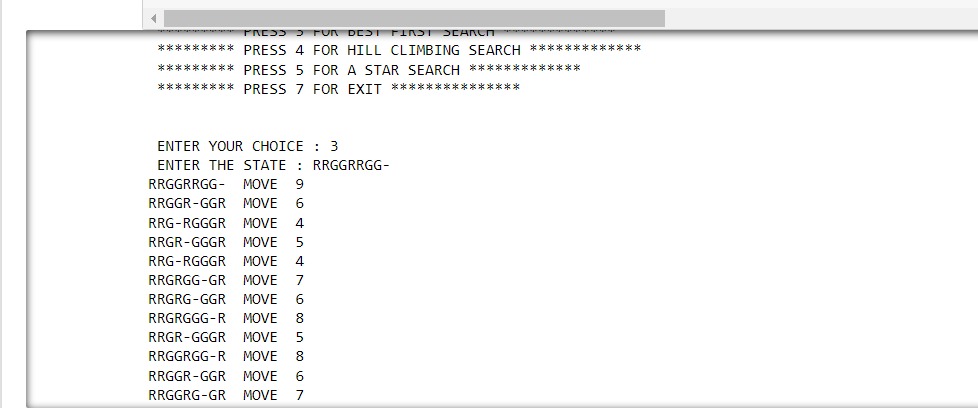


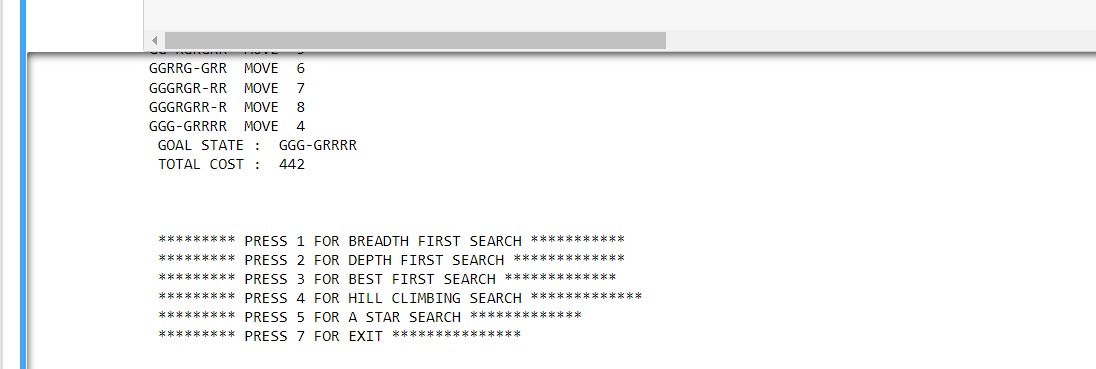


Selecting Best First search and input 4 Tiles:

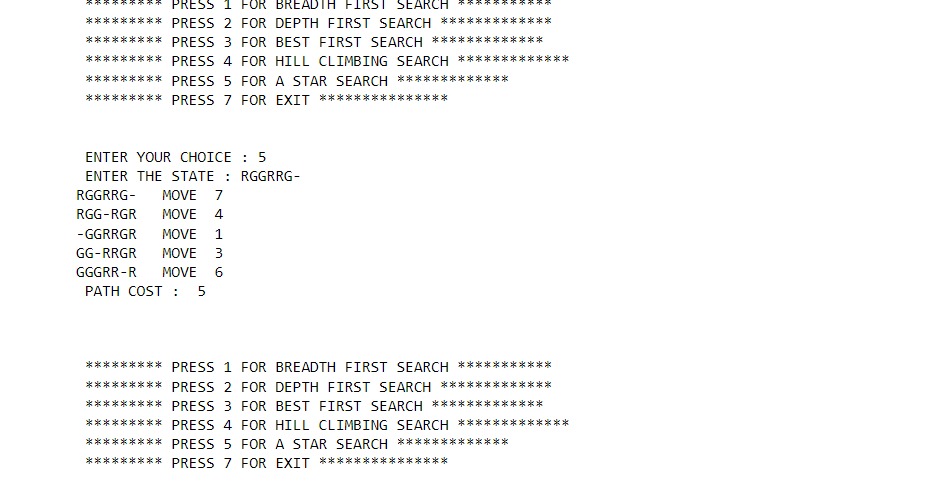


Selecting Best First Search and input 8 Tiles:

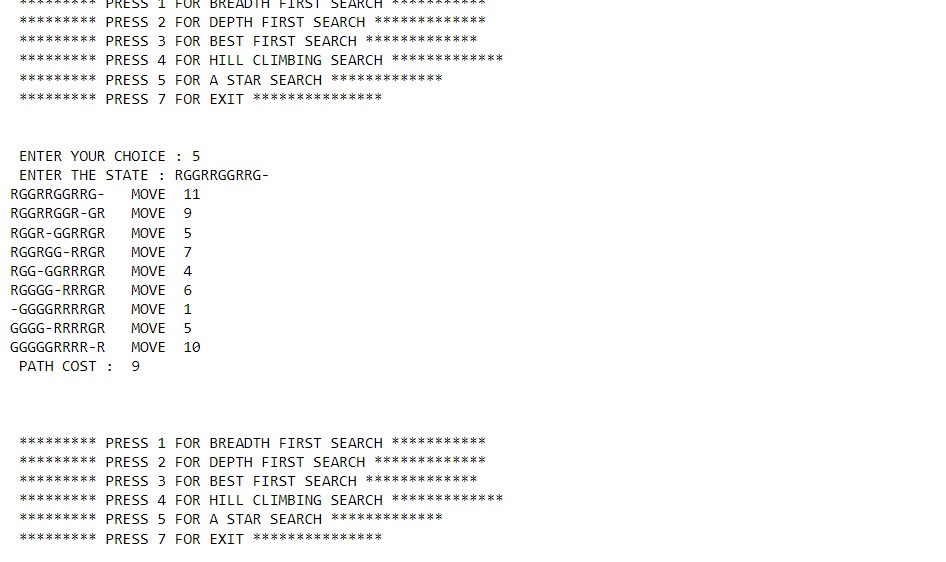




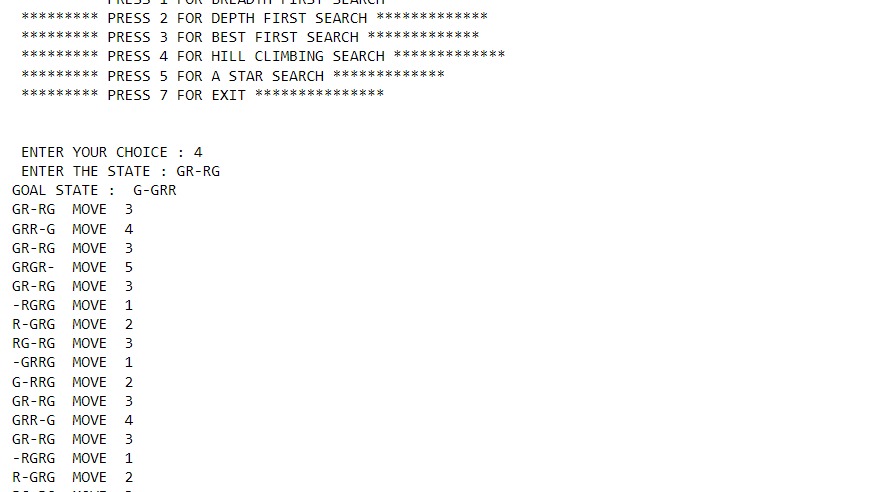
Selecting A Star Search and input 6 Tiles:

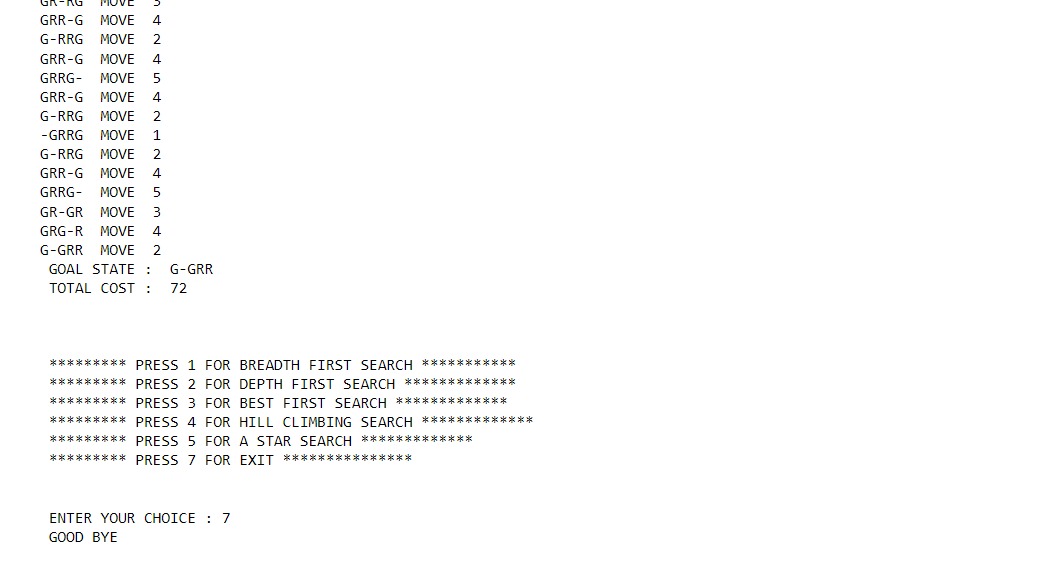


Selecting A star and input 10 Tiles:



Selecting Hill climbing and input 4 Tiles:





Giving Goal State as Input:

