```
Total time: 19.367s
File: /Users/rishabhjain/Documents/Masters/SEM 2/Aritificial Intelligence/Program/assignment_1/breathFirstSearch.py
Function: expand at line 24

Line #      Hits         Time      Per Hit    % Time    Line Contents
====================================================================
    24                                                  @cpu
    25                                                  def expand(board):
    26    449564      278.8ms            .      1.4%        for i in range(len(board.data)):                               # to find the loca
    27   1123859      380.8ms            .      2.0%            for j in range(len(board.data[i])):
    28    899077      324.6ms            .      1.7%                if board.data[i][j] == '*':
    29    112391       46.9ms            .      0.2%                    location = [i,j];
    30    112391       28.1ms            .      0.1%                    break
    31
    32    112391       28.0ms            .      0.1%        actions = []
    33    416708        2.38s            .     12.3%        for move in possible_actions(constants.board, location):       # to find all poss
    34    304317       15.87s        0.1ms     82.0%            actions.append([result(location, move, board.data) , move]) # prepare all poss
    35
    36    112391       30.0ms            .      0.2%        return actions                                                 # After expanding

Total time: 1.074s
File: /Users/rishabhjain/Documents/Masters/SEM 2/Aritificial Intelligence/Program/assignment_1/breathFirstSearch.py
Function: possible_actions at line 38

Line #      Hits         Time      Per Hit    % Time    Line Contents
====================================================================
    38                                                  @cpu
    39                                                  def possible_actions(board, location):                             # to find all poss
    40    112391       64.8ms            .      6.0%        actions = ["RIGHT","LEFT","UP","DOWN"]
    41    112391       36.0ms            .      3.4%        actionstopeform = []
    42
    43    561955      202.3ms            .     18.8%        for x in actions:
    44                                                          # for moving right
    45    449564      142.2ms            .     13.2%            if x == "RIGHT":
    46    112391       55.7ms            .      5.2%                if location[1]+1 < len(board):
    47     76061       52.1ms            .      4.8%                    actionstopeform.append([x,location[0],location[1]+1])
    48                                                          # for moving left
    49    337173      119.2ms            .     11.1%            elif x == "LEFT":
    50    112391       47.9ms            .      4.5%                if location[1]-1 >= 0:
    51     76010       46.2ms            .      4.3%                    actionstopeform.append([x,location[0],location[1]-1])
    52                                                          # for moving up
    53    224782       69.4ms            .      6.5%            elif x == "UP":
    54    112391       44.1ms            .      4.1%                if location[0]-1 >= 0:
    55     76063       37.6ms            .      3.5%                    actionstopeform.append([x,location[0]-1,location[1]])
    56                                                          # for moving down
    57    112391       34.6ms            .      3.2%            elif x == "DOWN":
    58    112391       49.9ms            .      4.7%                if location[0]+1 < len(board):
    59     76183       39.2ms            .      3.7%                    actionstopeform.append([x,location[0]+1,location[1]])
    60
    61    112391       32.4ms            .      3.0%        return actionstopeform

Total time: 14.991s
File: /Users/rishabhjain/Documents/Masters/SEM 2/Aritificial Intelligence/Program/assignment_1/breathFirstSearch.py
Function: result at line 63

Line #      Hits         Time      Per Hit    % Time    Line Contents
====================================================================
    63                                                  @cpu
    64                                                  def result(location,action,board):
    65    304317       12.43s            .     82.9%        newBoard = copy.deepcopy(board)                            # copy of a board so t
    66    304317      858.5ms            .      5.7%        temp = copy.deepcopy(newBoard[action[1]][action[2]])
    67    304317      825.1ms            .      5.5%        newBoard[action[1]][action[2]] = copy.deepcopy('*')
    68    304317      808.3ms            .      5.4%        newBoard[location[0]][location[1]]  = copy.deepcopy(temp)
    69    304317       73.9ms            .      0.5%        return newBoard                                            # return new board aft

Total time: 809.798s
File: /Users/rishabhjain/Documents/Masters/SEM 2/Aritificial Intelligence/Program/assignment_1/breathFirstSearch.py
Function: bfs at line 71

Line #      Hits         Time      Per Hit    % Time    Line Contents
====================================================================
    71                                                  @memory_profiler.profile
    72                                                  @cpu
    73                                                  def bfs(board):                                                # function BREADTH-FIF
    74         1            .            .         .        frontier = queue.Queue()
    75         1            .            .         .        node = Node(data = board)                                  # node ← NODE(problem.
    76         1            .            .         .        frontier.put(node)                                         # frontier ← a FIFO qu
    77                                                      # maxQueueSize = 1                                         # only for debug
    78         1            .            .         .        if constants.goalBoard == node.data:                       # if problem.IS-GOAL(r
    79                                                          return node                                            # then return node
    80         1            .            .         .        reached = []
    81         1            .            .         .        reached.append(board)                                      # reached ← {problem.I
    82    112391      600.6ms            .      0.1%        while not frontier.empty():                                # while not IS-EMPTY(f
    83    112391        1.02s            .      0.1%            val = frontier.get()                                   # node ← POP(frontier
    84    416706       21.03s        0.1ms      2.6%            for child in expand(val):                              # for each child in EX
    85    304316        1.19s            .      0.1%                s =  Node(data=child[0], depth = val.depth + 1, move= child[1] , prev=val)
    86    304316      260.4ms            .         .                if goalBoard == s.data:                            # if problem.IS-GOAL(s
    87                                                                  #print('Max queue size:', maxQueueSize)        # only for debug
    88         1            .            .         .                    return s                                       # then return child
    89    304315      783.15s        2.6ms     96.7%                if s.data not in reached:                          # if s is not in reach
    90    136150      347.0ms            .         .                    reached.append(s.data)                         # add s to reached
    91    136150        2.20s            .      0.3%                    frontier.put(s)                                # add child to frontie
    92                                                                  # maxQueueSize+=1                              # only for debug
    93                                                          #print('Max queue size:', maxQueueSize)                # only for debug
    94                                                 return failure                                           # return failure


Total time: 0.000s
File: /Users/rishabhjain/Documents/Masters/SEM 2/Aritificial Intelligence/Program/assignment_1/breathFirstSearch.py
Function: printStatistics at line 96

Line #      Hits         Time      Per Hit    % Time    Line Contents
=====================================================================
    96                                                  @cpu
    97                                                  def printStatistics(solution):
    98         1            .            .      0.3%        pathCost = 0
    99         1            .            .      0.3%        stateSequence = []
   100         1            .            .         .        actionSequence = []
   101
   102        25            .            .      2.1%        while solution.prev != None:
   103        24            .            .      2.4%            stateSequence.insert(0, solution.data)
   104        24            .            .      3.2%            actionSequence.insert(0, solution.move)
   105        24            .            .      2.1%            solution = solution.prev
   106        24            .            .      1.5%            pathCost += 1
   107
   108         1            .            .      7.4%        print('Action sequence:')
   109         1        0.2ms        0.2ms     59.9%        print(*actionSequence, sep='\n')
   110
   111         1            .            .      3.8%        print('\nState sequence:')
   112         1        0.1ms        0.1ms     14.7%        print(*stateSequence, sep='\n')
   113
   114         1            .            .      2.4%        print('\nPath cost:', pathCost)
```