```
Total time: 10.499s
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    232876      168.4ms            .      1.6%        for i in range(len(board.data)):                            # to find the loca
    27    582167      208.5ms            .      2.0%            for j in range(len(board.data[i])):
    28    465729      176.8ms            .      1.7%                if board.data[i][j] == '*':
    29     58219       23.8ms            .      0.2%                    location = [i,j];
    30     58219       15.4ms            .      0.1%                    break
    31
    32     58219       15.6ms            .      0.1%        actions = []
    33    214012        1.20s            .     11.4%        for move in possible_actions(constants.board, location):        # to find all poss
    34    155793        8.68s        0.1ms     82.6%            actions.append([result(location, move, board.data) , move]) # prepare all poss
    35
    36     58219       16.3ms            .      0.2%        return actions                                              # After expanding


Total time: 0.540s
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     58219       36.6ms            .      6.8%        actions = ["RIGHT","LEFT","UP","DOWN"]
    41     58219       18.7ms            .      3.5%        actionstopeform = []
    42
    43    291095       92.4ms            .     17.1%        for x in actions:
    44                                                          # for moving right
    45    232876       71.4ms            .     13.2%            if x == "RIGHT":
    46     58219       29.4ms            .      5.4%                if location[1]+1 < len(board):
    47     39012       26.7ms            .      5.0%                    actionstopeform.append([x,location[0],location[1]+1])
    48                                                          # for moving left
    49    174657       59.6ms            .     11.0%            elif x == "LEFT":
    50     58219       24.4ms            .      4.5%                if location[1]-1 >= 0:
    51     38989       23.9ms            .      4.4%                    actionstopeform.append([x,location[0],location[1]-1])
    52                                                          # for moving up
    53    116438       34.4ms            .      6.4%            elif x == "UP":
    54     58219       22.7ms            .      4.2%                if location[0]-1 >= 0:
    55     38894       19.7ms            .      3.6%                    actionstopeform.append([x,location[0]-1,location[1]])
    56                                                          # for moving down
    57     58219       17.3ms            .      3.2%            elif x == "DOWN":
    58     58219       26.3ms            .      4.9%                if location[0]+1 < len(board):
    59     38898       19.7ms            .      3.6%                    actionstopeform.append([x,location[0]+1,location[1]])
    60
    61     58219       16.2ms            .      3.0%        return actionstopeform


Total time: 8.192s
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    155793        6.81s            .     83.1%        newBoard = copy.deepcopy(board)                         # copy of a board so t
    66    155793      461.8ms            .      5.6%        temp = copy.deepcopy(newBoard[action[1]][action[2]])
    67    155793      445.2ms            .      5.4%        newBoard[action[1]][action[2]] = copy.deepcopy('*')
    68    155793      437.4ms            .      5.3%        newBoard[location[0]][location[1]]  = copy.deepcopy(temp)
    69    155793       40.3ms            .      0.5%        return newBoard                                         # return new board aft


Total time: 314.798s
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-FIR
    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     58219      319.4ms            .      0.1%        while not frontier.empty():                         # while not IS-EMPTY(f
    83     58219      574.2ms            .      0.2%            val = frontier.get()                            # node ← POP(frontier
    84    214010       11.42s        0.1ms      3.6%            for child in expand(val):                       # for each child in EX
    85    155792      694.3ms            .      0.2%                s =  Node(data=child[0], depth = val.depth + 1, move= child[1] , prev=val)
    86    155792      163.4ms            .      0.1%                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    155791      300.05s        1.9ms     95.3%                if s.data not in reached:                   # if s is not in reach
    90     76968      210.0ms            .      0.1%                    reached.append(s.data)                  # add s to reached
    91     76968        1.37s            .      0.4%                    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            .            .      0.8%        pathCost = 0
    99         1            .            .      0.4%        stateSequence = []
   100         1            .            .      0.4%        actionSequence = []
   101
   102        23            .            .      5.2%        while solution.prev != None:
   103        22            .            .      3.6%            stateSequence.insert(0, solution.data)
   104        22            .            .      4.4%            actionSequence.insert(0, solution.move)
   105        22            .            .      3.2%            solution = solution.prev
   106        22            .            .      2.8%            pathCost += 1
   107
   108         1            .            .     13.5%        print('Action sequence:')
   109         1        0.1ms        0.1ms     33.7%        print(*actionSequence, sep='\n')
   110
   111         1            .            .      3.6%        print('\nState sequence:')
   112         1        0.1ms        0.1ms     24.2%        print(*stateSequence, sep='\n')
   113
   114         1            .            .      4.4%        print('\nPath cost:', pathCost)
```