

```

def bfs(src, target):
    queue = []
    queue.append(src)

    exp = []

    while len(queue) > 0:
        source = queue.pop(0)
        exp.append(source)

        print("Current State:")
        print_matrix(source)

        if source == target:
            print("Success!")
            return

        poss_moves_to_do = possible_moves(source, exp)

        for move in poss_moves_to_do:
            if move not in exp and move not in queue:
                queue.append(move)

def possible_moves(state, visited_states):
    # Index of empty spot
    b = state.index(0)

    # Directions array
    d = []

    # Add all possible directions
    if b not in [0, 1, 2]:
        d.append('u')
    if b not in [6, 7, 8]:
        d.append('d')
    if b not in [0, 3, 6]:
        d.append('l')
    if b not in [2, 5, 8]:
        d.append('r')

    # Generate possible moves
    pos_moves_it_can = []
    for i in d:
        pos_moves_it_can.append(gen(state, i, b))

    return [move_it_can for move_it_can in pos_moves_it_can if move_it_can not in visited_states]

def gen(state, m, b):
    temp = state.copy()

    if m == 'd':
        temp[b + 3], temp[b] = temp[b], temp[b + 3]
    elif m == 'u':
        temp[b - 3], temp[b] = temp[b], temp[b - 3]
    elif m == 'l':
        temp[b - 1], temp[b] = temp[b], temp[b - 1]
    elif m == 'r':
        temp[b + 1], temp[b] = temp[b], temp[b + 1]

    return temp

def convert_to_matrix(state):
    return [state[i:i + 3] for i in range(0, 9, 3)]

def print_matrix(state):
    matrix = convert_to_matrix(state)
    for row in matrix:
        print(row)
    print()

# Example usage
src = [1, 0, 3, 4, 2, 6, 7, 5, 8]
target = [1, 2, 3, 4, 5, 6, 7, 8, 0]
bfs(src, target)

```



Current State:

[1, 0, 3]
[4, 2, 6]
[7, 5, 8]

Current State:

[1, 2, 3]
[4, 0, 6]
[7, 5, 8]

Current State:

[0, 1, 3]
[4, 2, 6]
[7, 5, 8]

Current State:

[1, 3, 0]
[4, 2, 6]
[7, 5, 8]

Current State:

[1, 2, 3]
[4, 5, 6]
[7, 0, 8]

Current State:

[1, 2, 3]
[0, 4, 6]
[7, 5, 8]

Current State:

[1, 2, 3]
[4, 6, 0]
[7, 5, 8]

Current State:

[4, 1, 3]
[0, 2, 6]
[7, 5, 8]

Current State:

[1, 3, 6]
[4, 2, 0]
[7, 5, 8]

Current State:

[1, 2, 3]
[4, 5, 6]
[0, 7, 8]

Current State:

[1, 2, 3]
[4, 5, 6]
[7, 8, 0]

Success!