

Q.2: Symmetries in Tic-Tac-Toe

input = {'A2': 'x', 'B1': 'o', 'B3': 'o', 'C2': 'x'}

Output:

***** Original Table *****

```
[' ', 'x', ' ']  
['o', ' ', 'o']  
[' ', 'x', ' ']
```

***** Rotationally Equivalent Matrix: 1 *****

```
[' ', 'o', ' ']  
['x', ' ', 'x']  
[' ', 'o', ' ']
```

Returns: [{'A1': ' ', 'A2': 'o', 'A3': ' ', 'B1': 'x', 'B2': ' ', 'B3': 'x', 'C1': ' ', 'C2': 'o', 'C3': ' '}]

input = {'A2': 'x', 'B3': 'o', 'B2': 'x', 'A1': 'x', 'A3': 'o', 'B1': 'o'}

Output:

***** Original Table *****

```
['x', 'x', 'o']  
['o', 'x', 'o']  
[' ', ' ', ' ']
```

***** Rotationally Equivalent Matrix: 1 *****

```
['o', 'o', ' ']  
['x', 'x', ' ']  
['x', 'o', ' ']
```

***** Rotationally Equivalent Matrix: 2 *****

```
[' ', ' ', ' ']  
['o', 'x', 'o']  
['o', 'x', 'x']
```

***** Rotationally Equivalent Matrix: 3 *****

```
[' ', 'o', 'x']  
[' ', 'x', 'x']  
[' ', 'o', 'o']
```

Returns: [{ 'A1': 'o', 'A2': 'o', 'A3': ' ', 'B1': 'x', 'B2': 'x', 'B3': ' ', 'C1': 'x', 'C2': 'o', 'C3': ' ' }, { 'A1': ' ', 'A2': ' ', 'A3': ' ', 'B1': 'o', 'B2': 'x', 'B3': 'o', 'C1': 'o', 'C2': 'x', 'C3': 'x' }, { 'A1': ' ', 'A2': 'o', 'A3': 'x', 'B1': ' ', 'B2': 'x', 'B3': 'x', 'C1': ' ', 'C2': 'o', 'C3': 'o' }]

input = { 'A1': 'x', 'A2': 'o', 'B2': 'o' }

Output:

***** Original Table *****

```
[ 'x', 'o', ' ' ]
[ ' ', 'o', ' ' ]
[ ' ', ' ', ' ' ]
```

The given state is not a valid state of Tic-Tac-Toe table.

Returns: [{}]

input = { 'A1': 'o', 'A2': 'x', 'A3': 'o', 'C1': 'x', 'C2': 'o', 'C3': 'x' }

Output:

***** Original Table *****

```
[ 'o', 'x', 'o' ]
[ ' ', ' ', ' ' ]
[ 'x', 'o', 'x' ]
```

***** Rotationally Equivalent Matrix: 1 *****

```
[ 'o', ' ', 'x' ]
[ 'x', ' ', 'o' ]
[ 'o', ' ', 'x' ]
```

***** Rotationally Equivalent Matrix: 2 *****

```
[ 'x', 'o', 'x' ]
[ ' ', ' ', ' ' ]
[ 'o', 'x', 'o' ]
```

***** Rotationally Equivalent Matrix: 3 *****

```
[ 'x', ' ', 'o' ]
[ 'o', ' ', 'x' ]
[ 'x', ' ', 'o' ]
```

Returns: [{ 'A1': 'o', 'A2': ' ', 'A3': 'x', 'B1': 'x', 'B2': ' ', 'B3': 'o', 'C1': 'o', 'C2': ' ', 'C3': 'x' }, { 'A1': 'x', 'A2': 'o', 'A3': 'x', 'B1': ' ', 'B2': ' ', 'B3': ' ', 'C1': 'o', 'C2': 'x', 'C3': 'o' }, { 'A1': 'x', 'A2': ' ', 'A3': 'o', 'B1': 'o', 'B2': ' ', 'B3': 'x', 'C1': 'x', 'C2': ' ', 'C3': 'o' }]

input = { 'B2': 'x' }

Output:

***** Original Table *****

```
[ ' ', ' ', ' ' ]
[ ' ', 'x', ' ' ]
[ ' ', ' ', ' ' ]
```

Returns: []

```
input = {'A1': 'x', 'A2': 'o', 'A3': 'x', 'B1': 'o', 'B3': 'o', 'C1': 'x', 'C2':
'o', 'C3': 'x'}
```

Output:

***** Original Table *****

```
['x', 'o', 'x']
['o', ' ', 'o']
['x', 'o', 'x']
```

Returns: []

Q.3: Shortest Path Problem

```
Input = [[0, 2, 4, 2, math.inf, math.inf, math.inf, math.inf],
[2, 0, math.inf, math.inf, 7, 4, 6, math.inf],
[4, math.inf, 0, math.inf, 3, 2, 1, math.inf],
[2, math.inf, math.inf, 0, 4, 1, 5, math.inf],
[math.inf, 7, 3, 4, 0, math.inf, math.inf, 7],
[math.inf, 4, 2, 1, math.inf, 0, math.inf, 9],
[math.inf, 6, 1, 5, math.inf, math.inf, 0, 6],
[math.inf, math.inf, math.inf, math.inf, 7, 9, 6, 0]]
```

Output:

Iteration list at Start: [inf, inf, inf, inf, inf, inf, inf, 0]

Iteration_list 1 [inf, inf, inf, inf, 7, 9, 6, 0]

Iteration_list 2 [inf, 12, 7, 10, 7, 9, 6, 0]

Iteration_list 3 [11, 12, 7, 10, 7, 9, 6, 0]

Iteration_list 4 [11, 12, 7, 10, 7, 9, 6, 0]

***** Stop Iteration *****

***** Function Returns *****

Returns_1: Iteration list

{'S': 11, 'A': 12, 'B': 7, 'C': 10, 'D': 7, 'E': 9, 'F': 6, 'T': 0}

Returns_2: Policy_Table

{'S': 'B', 'A': 'F', 'B': 'F', 'C': 'E', 'D': 'T', 'E': 'T', 'F': 'T', 'T': 'T'}

```
input = [[0, 2, 3, math.inf],
         [2, 0, math.inf, 3],
         [3, math.inf, 0, 2],
         [math.inf, 3, 2, 0]]
```

Output:

Iteration list at Start: [inf, inf, inf, 0]

Iteration_list 1 [inf, 3, 2, 0]

Iteration_list 2 [5, 3, 2, 0]

Iteration_list 3 [5, 3, 2, 0]

***** Stop Iteration *****

***** Function Returns *****

Returns_1: Iteration list

{'S': 5, 'A': 3, 'B': 2, 'T': 0}

Returns_2: Policy_Table

{'S': ['A', 'B'], 'A': 'T', 'B': 'T', 'T': 'T'}