Module 5 Lab - Recursive Board Game

Model a circular board game consisting of numbered tiles. The numbers represent how many tiles you can move clockwise (CW) or counter-clockwise (CCW). It's okay to loop around - moves that go before the first tile or after the last are valid. The goal is to reach the final tile (the tile 1 counter-clockwise from the start).

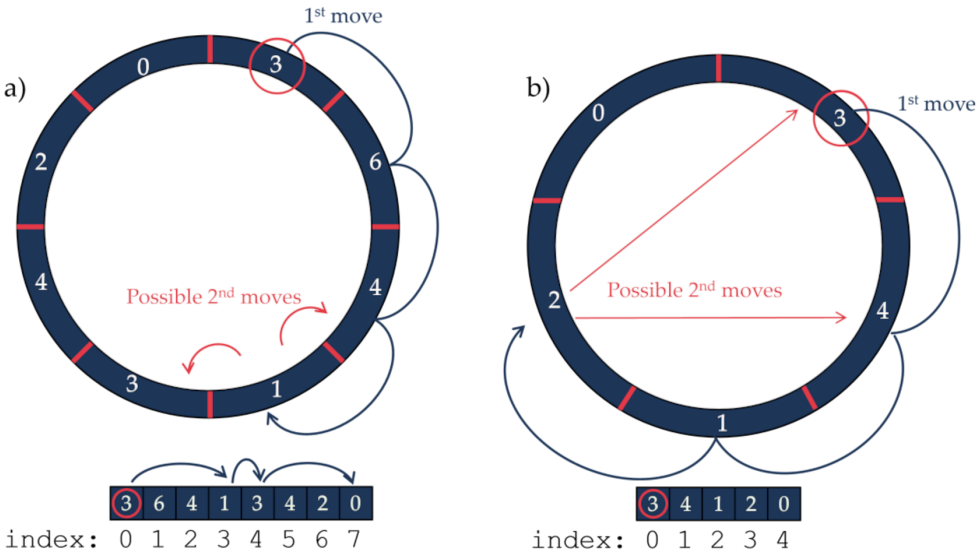

Figure 1: (a) `[3, 6, 4, 1, 3, 4, 2, 0]` is solveable in 3 moves. (b) `[3, 4, 1, 2, 0]` is unsolveable. While not shown, moving 3 spaces CCW at the start would also be a valid first move.

`SolvePuzzle.py`

- `solve_puzzle(board)` returns a boolean denoting if `board` is solveable.

```
>>> solve_puzzle([3, 6, 4, 1, 3, 4, 2, 0])
True
>>> solve_puzzle([3, 4, 1, 2, 0])
False
```

Tips

- Use memoization to to avoid infinite loops
- You can assume the numbers on tiles are non-negative integers (0 is valid, and may appear on any tile)
- The modulo operator `%` is helpful for finding indices when you loop around
- Add unittests to `TestSolvePuzzle.py` to help debug

Submission

At a minimum, submit the following files:

- `solve_puzzle.py`
- `test_solve_puzzle.py`

Students must submit **individually** by the due date (typically Sunday at 11:59 pm EST) to receive credit.