Our Solution(s)

Run Code

Solution 1

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(n^2 + m) time | O(n + m) space
def patternMatcher(pattern, string):
    if len(pattern) > len(string):
        return []
    newPattern = getNewPattern(pattern)
    didSwitch = newPattern[0] != pattern[0]
    counts = {"x": 0, "y": 0}
    firstYPos = getCountsAndFirstYPos(newPattern, counts)
    if counts["y"] != 0:
        for lenOfX in range(1, len(string)):
            lenOfY = (len(string) - lenOfX * counts["x"]) / counts["y"]
            if lenOfY <= 0 or lenOfY % 1 != 0:
                continue
            lenOfY = int(lenOfY)
            yIdx = firstYPos * lenOfX
            x = string[:lenOfX]
            y = string[yIdx : yIdx + lenOfY]
            potentialMatch = map(lambda char: x if char == "x" else y,
            if string == "".join(potentialMatch):
                return [x, y] if not didSwitch else [y, x]
    else:
        lenOfX = len(string) / counts["x"]
        if lenOfX % 1 == 0:
            lenOfX = int(lenOfX)
            x = string[:lenOfX]
            potentialMatch = map(lambda char: x, newPattern)
            if string == "".join(potentialMatch):
                return [x, ""] if not didSwitch else ["", x]
    return []
```

Your Solutions

Run Code

Solution 1   Solution 2   Solution 3

```python
def patternMatcher(pattern, string):
    # Write your code here.
    pass
```

Our Tests

Custom Output

Submit Code

Run or submit code when you're ready.