Daily Coding Problem #108

Problem

This problem was asked by Google.

Given two strings A and B, return whether or not A can be shifted some number of times to get B.

For example, if A is `abcde` and B is `cdeab`, return true. If A is `abc` and B is `acb`, return false.

Solution

If the strings are not the same length, then we can immediately return false.

One solution might be to use a doubly nested for loop, and compare each character starting at different offsets and verifying that they all match up:

```
def is_shifted(a, b):
```

```
    if len(a) != len(b):
        return False

    for i in range(len(a)):
        if all(a[(i + j) % len(a)] == b[j] for j in range(len(a))):
            return True

    return False
```

Another cleaner way to solve this might be to concatenate one of the strings to itself (like a + a), and try looking for the other string in this concatenated string. If the string is shifted, we should find it in the concatenated string.

```
def is_shifted(a, b):
    if len(a) != len(b):
        return False

    return b in a + a
```

These solutions both take O(N^2) time.
