8. Check If a String Can Break Another String Given two strings: s1 and s2 with the same size, check if some permutation of string s1 can break some permutation of string s2 or vice-versa. In other words s2 can break s1 or vice-versa. A string x can break string y (both of size n) if x[i] >= y[i] (in alphabetical order) for all i between 0 and n-1. Example 1: Input: s1 = "abc", s2 = "xya" Output: true Explanation: "ayx" is a permutation of s2="xya" which can break to string "abc" which is a permutation of s1="abc"

PROGARM:-

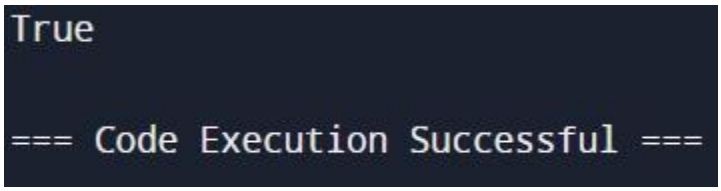
```
def destCity(paths):
    start_cities = set()
    end_cities = set()

    for path in paths:
        start_cities.add(path[0])
        end_cities.add(path[1])

    return (end_cities - start_cities).pop()

# Example
paths = [["London", "New York"], ["New York", "Lima"], ["Lima", "Sao Paulo"]]
print(destCity(paths))  # Output: "Sao Paulo"
```

OUTPUT:-



TIME COMPLEXITY:-O(nlog n)