Brute Force Analysis

combinationHelper():

- n is number of vertices
- Recursively generates all possible paths using permutations of the given vertices.
- combinationHelper is called n! times due to checking each permutation
- Each call to combinationHelper() calls 1. test(): calls checkCycle() on updated path
- checkCycle() : runs m - 1 times , m being the length of the path, calls searchLinkedList inside a loop so its time complexity is θ(n x m)
- searchLinkedList() gets called inside checkCycle and has a time complexity of θ(n) since it checks every node in the linked list

    2. Updates current path - θ(n)
    Updates remaining vertices - θ(n)

Time Complexity = θ(n! x (m x n))
But in the worst case m can = n so
Time complexity = θ(n! x n^2)