CS341

2. The graph can be modeled as such:
 - the vertices represent all of the possible permutations of size n.
 - the edges represent the required reversals to reach a permutation from another node.

Since there are $n!$ permutations of a string sized n, there are $n!$ nodes.
Since there are n^2 possible intervals of a string, there are n^2 edges for each node.
Therefore, performing a Breadth First Search would yield a worst case of $O(\min(n!n^2, n^{2k}))$ where k is the least number of reversals required.

The $n!n^2$ part is from the worst case where BFS goes through each of the n^2 edges through each of the $n!$ nodes.
The n^{2k} is the least number of edges required to find the sorted permutation. BFS goes through n^2 edges for each node in a layer, and there are at least k layers because k is the # of reversals. Hence n^{2k}.

```
queue.enqueue([v]);
while (!queue.empty()) {
    path = queue.dequeue();
    visited.push(path[path.size()]);
    if (path[path.size()] == sorted)  return path;
    for (w in path[path.size].neighbours) {
        if(!visited.contains(w)) queue.enqueue([path, w]);
    }
}
```

Correctness:

Since BFS goes through all of a vertex's neighbours before proceeding to another vertex, it is guaranteed to reach the target node in the least number of edges.

---

3. numShortestPaths

3. BFS(node s, node t)