Task 1: Computing shortest path

Requirement: O(ElogV) time complexity, O(V + E) space complexity

V = vertices, E = edges

Step	**Time complexity**	**Space complexity**
Create an adjacency list from the input file	O(E)	O(E)
Implement Dijkstra's with min-heap, including vertices array	O(ElogV)	O(V) for vertices array

Overall meets requirements.

Task 2: The minimum detour path

Requirement: O(ElogV) time complexity, O(V + E) space complexity

Step	**Time complexity**	**Space complexity**
Create an adjacency list from the input file, with two sets of each vertex (indicated as discrete by a numerical offset) and connect the two vertices by a zero weight edge between the corresponding customer vertices	O(E)	O(E)
Implement Dijkstra's with min-heap, including vertices array, but add the offset so that the target vertex is on the second set of vertices in the graph	O(ElogV)	O(V) for vertices array