G Yeung, AR Sevgili, RG Xu, SC Lin, SH Dabbaghian, HK Mok

[Email address]

Delivery program

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

Intergration testing

SFT244 Group5

# Test Description

**Test Name or ID**: Integration T01-04

**Test Type**: Integration Test

**Description**: This set of test exam if the functions can print the base map and the three truck routes as base map for the following functions to select the shortest path.

**Setup:** Populate a map and add three routes to it.

**Test Function**: populateMap(), getBlueRoute(), getYellowRoute(), addRoute(), printMap()

**Test Scenarios:**

|  |  |  |
| --- | --- | --- |
| Description | Test Data | Expected Result |
| Add blue truck route to base map | struct Map, blueRoute | Display base map and the blue truck route |
| Add green truck route to base map | struct Map, greenRoute | Display base map and the green truck route |
| Add yellow truck route to base map | struct Map, yellowRoute | Display base map and the yellow truck route |
| Add all the three truck routes to base map | struct Map, blueRoute, greenRoute, yellowRoute | Display base map and three trucks route |

**Test Name or ID**: Integration T05-08

**Test Type**: Integration test

**Description**: Testing the integration between the “shortestPath” and “distance” functions to ensure the correct calculation of the distance for the shortest path.

**Setup:** Set up a map and points.

**Test Function**: “shortestPath” and “distance”

**Test Scenarios:**

|  |  |  |
| --- | --- | --- |
| Description | Test Data | Expected Result |
| Calculate distance for the shortest path | Start Point: (0,0)  Destination Point: (9,24) | Calculate the distance for the shortest path between points (0,0) and (9,24). |
| Calculate distance for an invalid path | Start Point: (0,0)  Destination Point: Invalid Path (e.g., points with obstacles) | Error handling for an invalid path, distance calculation fails. |
| Calculate distance for an empty path | Start Point: (0,0)  Destination Point: (0,0) | Distance for an empty path (same start and destination) is 0. |
| Calculate distance with invalid parameters | Invalid Start Point (e.g. -1, -1)  Destination Point: (9,24) | Error handling for invalid start point, distance calculation fails. |

**Test Name or ID**: Integration T09-T12

**Test Type**: Integration test

**Description**: Testing the integration between the “shortestPath” and “distance” functions to ensure the correct calculation of the distance for the closest route with some extra midway point.

**Setup:** Set up a map, points, and routes.

**Test Function**: “shortestPath” and “distance”

**Test Scenarios:**

|  |  |  |
| --- | --- | --- |
| Description | Test Data | Expected Result |
| Calculate distance for the closest route | Start Point: (0,0)  Routes:  Route 1: (0,0) -> (4,0) -> (9,24)  Route 2: (2,2) -> (8,8) -> (17,17) -> (9,24) | Calculate the distance for the closest route between points (0,0) and (9,24). |
| Calculate distance for an invalid route | Start Point: (0,0)  Invalid Route (e.g., points with obstacles) | Error handling for an invalid route, distance calculation fails. |
| Calculate distance for an empty route | Start Point: (0,0)  Empty Route (0,0) | Distance for an empty route is 0. |
| Calculate distance with invalid parameters | Start Point: (0,0)  Invalid Route (e.g., points outside map boundaries) | Error handling for invalid route, distance calculation fails. |

**Test Name or ID**: Integration T13-T16

**Test Type**: Integration test

**Description**: Testing the integration of functions related to route management involving 2 functions.

**Setup:** No specific setup required.

**Test Function**: “addPointToRouteIfNot”, “ptExistInRt”

**Test Scenarios:**

|  |  |  |
| --- | --- | --- |
| Description | Test Data | Expected Result |
| Identify possible moves and add points to the route | Current Point: (2, 2)  Existing Route: Empty | Return 0 from the function “ptExistInRt”. No new point added. |
| Add points to an existing route | Existing Route: [(2, 2), (2, 3), (2, 4)]  New Point: (2, 2) | Return 1 from the function “ptExistInRt”. New point added. |
| Check if a point exists in the route | Existing Route: [(2, 2), (2, 3), (2, 4)]  Point to Check: (3, 4) | Return 0 from the function “ptExistInRt”,  New point added. |
| Add a point to the route only if it doesn't exist | Existing Route: [(2, 2), (2, 3), (2, 4)]  New Point: (2, 3) (already exists) | Return 1 from the function “ptExistInRt”. No new point added. |