# Black Box Test Plan

In this section, you must provide your black-box test plan with at least 5 black-box test cases.

Make sure:

* You describe how to setup the system to begin black-box testing
* Test IDs are uniquely identified and descriptive
* Test descriptions are fully specified with complete inputs, specific values, and preconditions
  + Be sure to provide SPECIFIC INPUTs and VALUEs so that your test cases are repeatable
* Expected results are fully specified with specific output values
* The test process is fully provided and clear. No modifications are needed.
* All tests cover scenarios based on the problem statement
* All tests cover unique scenarios for the system
* All strategies for black-box testing are demonstrated in the tests (ECP, BVA, DT)

**Input File (Airports.txt):**

AIRPORT\_CODE,LATITUDE,LONGITUDE

DFW,32.89680099487305,-97.03800201416016

MIA,25.79319953918457,-80.29060363769531

ORH,42.26729965209961,-71.87570190429688

RDU,35.877601623535156,-78.7874984741211

SEA,47.44900131225586,-122.30899810791016

SFO,37.61899948120117,-122.375

|  |  |  |  |
| --- | --- | --- | --- |
| Test ID | Description | Expected Results | Actual Results |
| Test 1: Invalid Input File (DT) | **Preconditions:** “Invalid.txt” does not exist.  Run AirlineHubManager.  At the input file prompt, type: “Invalid.txt”. | The program starts and the user is prompted to specify an input file.  The program re-prompts the user to specify an input file. | Invalid filename, please try again.  Please specify a filename: |
| Test 2: Valid Input File (ECP) | **Preconditions:** “Airports.txt” exists.  Run AirlineHubManager.  At the input file prompt, type: “input/airports.txt”. | The program starts and the user is prompted to specify an input file.  The program prompts the user to choose an option from the operations menu. | Operations menu:  View (C)onnections  View (H)ubs  (Q)uit  Please choose one of the above options: |
| Test 3: Generate Flight Connections (ECP) | **Preconditions:** Test 2 has passed.  At the operations menu prompt, choose to generate a list of flight connections. | The program outputs:  FlightList[  Flight[airport1=ORH, airport2=RDU, distance=576.4],  Flight[airport1=SEA, airport2=SFO, distance=679.6],  Flight[airport1=MIA, airport2=RDU, distance=702.8],  Flight[airport1=DFW, airport2=RDU, distance=1059.7],  Flight[airport1=DFW, airport2=SFO, distance=1462.3]  ]  The program prompts the user to choose an option from the operations menu. | FlightList[  Flight[airport1=ORH, airport2=RDU, distance=576.4],  Flight[airport1=SEA, airport2=SFO, distance=679.6],  Flight[airport1=MIA, airport2=RDU, distance=702.8],  Flight[airport1=DFW, airport2=RDU, distance=1059.7],  Flight[airport1=DFW, airport2=SFO, distance=1462.3]  ]  Operations menu:  View (C)onnections  View (H)ubs  (Q)uit  Please choose one of the above options: |
| Test 4: Generate Airport Hub Report (ECP) | **Preconditions:** Test 2 has passed.  At the operations menu prompt, choose to generate an airport hub report. | The program outputs:  HubReport[  RDU has 3 connections.  ]  The program prompts the user to choose an option from the operations menu. | HubReport[  RDU has 3 connections.  ]  Operations menu:  View (C)onnections  View (H)ubs  (Q)uit  Please choose one of the above options: |
| Test 5: Quit Program (ECP) | **Preconditions:** Test 2 has passed.  At the operations menu prompt, choose to quit the program. | The program closes and the input file remains unedited. | Program terminates normally, airports.txt is unmodified. |