Black Box Test Plan

The black box tests will require the following files to be in the input folder:

airport.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

empty.txt (empty file)

none.txt (file which does not exist)

Test ID	Description	Expected Results	Actual Results
1. LoadValidFile Test (**ECP -** tests software starting up and loading valid file)	**Preconditions**: *airport.txt* must exist on the system as described above The user launches the program and gets prompted for file name. **Enter name of file:** User inputs “airport.txt”	The software asks the user to choose one of three options: **Menu:** **[1] Generate flight connections** **[2] Produce hub report** **[3] Quit** **Option ->**	The software asks the user to choose one of three options: **Menu:** **[1] Generate flight connections** **[2] Produce hub report** **[3] Quit** **Option ->**
2. FileDoesNot ExistTest (**DT -** tests invalid file being read in)	**Preconditions**: none The user launches the program and gets prompted for file name. **Enter name of file:** User inputs “none.txt”	The software re-prompts the user to specify a new file after displaying error message “File cannot be opened.”	The software re-prompts the user to specify a new file after displaying error message “File cannot be opened.”

3. GenerateFlightListTest **(ECP -** tests generating valid flight list and **BVA -** all data must be sifted through from first entry to last)	**Preconditions**: Test 1 passed The software asks the user to choose one of three options: **[1] Generate flight connections** **[2] Produce hub report** **[3] Quit** User chooses option to generate flight list for file by entering the number 1 **Option ->** 1	The software displays the following list of flight connections: **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 software displays the following list of flight connections: **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]** **]**
4. NoHubTest (**ECP -** tests valid file with no hub airports)	**Preconditions**: *empty.txt* must exist on the system as described above The user launches the program and gets prompted for a file name. **File Name:** User inputs “empty.txt” The software asks the user to choose one of three options: **[1] Generate flight connections** **[2] Produce hub report** **[3] Quit** **Option ->** User inputs “2”	The software outputs the following: **FlightList[** **No airports have at least 3 connecting flights.** **]**	The software outputs the following: **FlightList[** **No airports have at least 3 connecting flights.** **]**

5. QuitTest (**ECP -** tests software option to quit)	**Preconditions**: Test 1 passed User chooses option to quit software by entering the number 3 **Option ->** 3	The software closes, and the input files remain unedited.	The software closes, and the input files remain unedited.
6. InvalidOption Test (**DT -** testing invalid option for software)	**Preconditions**: Test 1 passed User chooses invalid option by entering the number xyz **Option ->** xyz	The software prints a message reading “Invalid option.” and reprompts the user to try picking an option again.	The software prints a message reading “Invalid option.” and reprompts the user to try picking an option again.