<Introduction Paragraph>

Class and Hierarchy Details

Input Format

The first line of the input will consist of the max

Output Format

Some Remarks

Your task is to load, transfer and disembark passengers. Each passenger will have a list of destinations which is a list of airports and their starting airport. You will load a passenger to an airplane from its initial airport. From there the plane can fly to any airport it wants to, however, the passenger cannot disembark from the plane if the plane lands in an airport which is not a future destination of the passenger. If you don’t want to take the passenger to one of its future destinations by the aircraft it’s in, you can transfer the passenger to another airplane in the current airport.

If a passenger lands in an airport which has a higher index in the destinations airport list, then the currentAirport integer will be set to the index of that airport. Passengers cannot disembark in airports that have a smaller index than the currentAirport index.

In the input part, destinations list of airports will be given as a list of integers. These integers are airport ID’s. The airports which the passenger wants to visit will be in sorted order of airport types. If the index i is a hub airport, index i + 1 can be all three types of airports or the list could end. If the index i is a major airport, index i + 1 can be a major airport or a regional airport or the list could end. If the index i is a regional airport, the list must end.

Revenue (ticket price) will be collected only when the passenger disembarks. This revenue will be calculated from the current budget, the position of disembarkation, baggage count and from the seat type. The revenue will be subtracted from the budget of the passenger and added to the total revenue of the airline.

Every operation with an airplane must check if the airplane is in the

Funtionalities:

boolean addFuel(double fuel);

Adds the fuel amount to the current fuel of the plane. Must check the validity of the action. If it’s not valid, it should return false, if the operation is valid, it must take the necessary actions and then return true.

Fuel has some weight; plane has maximum fuel capacity and maximum weight limit.

Airport.java

Airport.java be an abstract class that will be the parent of the concrete Airport classes. These concrete classes are HubAirport, MajorAirport and RegionalAirport. All will extend the Airport class.

Aircrafts are refueled at the airport. The price of fuel will be determined by the airport the aircraft is currently in.

The airport has an operationCost. This is the fee paid of every operation done in the airport.