description:

1.class: airplane

assume: airplane only shuffles between two airport. attributes like speed, number of passengers at set as default value two airport references are save inside as destination and origin. and swap according to "departure" event

2.class:airport

airport saves latitude and longitude information, the distance between two airport is calculated according to function `double GetDistance(Airport origin, Airport dest)` and distance to other airports are saved in each airport as a hashtable field after being computed for the first time.

```
public double distanceTo(Airport dest)
{
    double dist;
    if(distanceToOtherAirport.containsKey(dest.getName()))
    dist = distanceToOtherAirport.get(dest.getName());
    else {
    dist = GetDistance(this,dest);
    distanceToOtherAirport.put(dest.getName(), dist);
        }
return dist;
    }
```

this distance information is used together with airplane speed to determine to schedule 'arrival event'.

four type of events: 1. **plane_depart_ready**: set passengers number to airplane check if runway is clear, if so, schedule **plane_depart_ready** event, else add to waiting list. 2. **plane_departs_finished**: schedule **plane_arrives** event at another airport, set runway to free or hand runway to next waiting airplane in list. 3. **plane_arrives**: schedule **plane_landed** event or add airplane to list 4. **plane_landed**: swap the dest and origin count passenger number in the arrival airplane schedul **plane_depart_ready** event