

## **Plane Algorithm Description**

### **Saved program plane\_seating.java**

The program will initially start with an empty airplane. The system will ask seating preferences (such as window, isle seat, section of the plane) as well as how many tickets.

The plane algorithm was simulated using an array of 18 rows and 6 columns to represent the passenger seating. Each passenger can choose a seat (1), have a preference (2) or have no preference (.3). After choosing the seating availability will be reduced by 1.

Based on availability and seating preferences, the program will choose the seats, display a diagram and the coordinates. The system should also allow for a customer to pick a specific seat. Passengers can choose the any row or section of the plane being window seat, aisle or middle. Assume that the middle 3 rows in section 2 are wing seats.

Potential passengers on the plane are able to choose their desired location based on being the first do so. If the seats have already been taken then the passengers would need to make another selection. The desired goal is to have the program function for family groupings being together which only works based on availability. The passengers would be able to control the proximity they will have to each other on the plane based on availability. This program has no bias in terms of seating. It is strictly based on availability.

The goal of the program is to group passengers together if they are travelling together, choose specific seats, then choose by preferences.

Also having a routine for placing passengers in the next available spot, if there are no preferences.

The way the information is displayed is based on seat availability which is (O for Empty, X for occupied).