Questions & Answers

Please detail any assumptions you have made

Outlined assumptions:

- Each co-ordinate can hold a maximum of one event.
- Each event has zero or more tickets.
- Each ticket has a non-zero price,

Other assumptions:

- Tickets for the same event can have numerous different prices
- No priority is made between events with the same distance

How might you change your program if you needed to support multiple events at the same location?

I would change the Event attribute in MapCoordinates to be a Set of events to prevent accidental duplication, and change the MapCoordinates getters and setters for this accordingly. Then in findEvents(), in the EventFinder class, I would add an iterator so go through the Events for each coordinate being checked.

How would you change your program if you were working with a much larger world size?

Rather than looping through the known events, I would implement Dijkstra's algorithm to find the events closest to the input coordinates. I would also move the coordinates into a different data structure that would be faster to traverse, such as a tree.

Adding concurrency would also help improve the search time. Rather than checking the closest events one at a time, I would run multiple threads to find the five closest concurrently.