

Programming Assignment #2
Airport simulation
From the text Chapter 7 #9 pg. 407

Description

Write a simulation for a small airport that has one runway. There will be a queue of planes waiting to land and an queue of planes waiting to takeoff. However, only one plane can use the runway at a time. So there can be only one takeoff or one landing in progress at any one time. Assume that all takeoffs take the same amount of time. Assume that all landings take the same amount of time, although this does not need to be the same as the takeoff time. Assume that planes arrive for landing at random times, but with a specified probability of a plane arriving during any given minute. Similarly, assume that the planes arrive at the takeoff queue at random times, but with a (possibly different) specified probability of a departure. (Despite the fact that takeoffs and landings are scheduled, delays make this a reasonable assumption.) Since it is more expensive, and more dangerous to keep a plane waiting to land than it is to keep a plane waiting to takeoff, landings will have a priority over takeoffs. Thus, as long as some plane is waiting to land, no plane can takeoff. Use a clock that is an unsigned integer variable that counts the number of minutes simulated. Use the random number generator to simulate arrival and departure times of airplanes.

The simulation can be used, among other things, for deciding when air traffic has become so heavy that a second runway must be built. Hence the simulation will simulate conditions would be a disaster in which planes crash because they ran out of fuel while waiting too long in the landing queue. By examining the simulated situation, the airport authority hopes to avoid the real tragedy. Assume all planes can remain in the queue the same amount of time before they run out of fuel. If a plane runs out of fuel, your simulation will not discover this until the simulated plane is removed from the queue; at that point, the fact that the plane crashed is recorded, that plane is discarded, and the next plane is processed. A crashed plane is not considered in the calculation of waiting time. At the end of the simulated time, the landing queue is examined to see whether any of the planes in the simulated queue have crashed. You can disregard the planes left in the queue at the end of the simulation, other than those that crashed for lack of sufficient fuel. Use the following input and output specification:

Input	Output
(1) The amount of time needed for one plane to land; (2) the amount of time needed for one plane to takeoff. (3) the average amount of time between arrival of planes to the landing queue. (4) the average amount of time between arrival of planes to the takeoff queue. (5) the maximum amount of time that a plane can stay in the landing queue without running out of fuel and crashing. (6) the total length of time to be simulated.	(1) The number of planes that took off in the simulated time. (2) the number of planes that landed in the simulated time. (3) the number of planes that crashed because they ran out of fuel before they could land. (4) the average time a plane spent in the takeoff queue. (5) the average time that a plane spent in the landing queue.

You can programmatically enter data or create a User Interface.

A data set you must run is as follows:

Time to land: 3 minutes
Time to takeoff: 4 minutes
Average time between landing planes: 2 minutes
Average time between taking off planes: 1 minute
Max time in queue: 3 minutes
Total time to simulate: 600 minutes

Grading

To receive full credit for this program the following must be submitted:

- | | |
|---|-------|
| 1) Pseudo code for the algorithms utilized. | (25%) |
| 2) Well commented Syntactically Correct "Java" code. | (60%) |
| 3) Runs with an assortment of data, including the data above. | (15%) |

Late programs will not be accepted without prior approval. Waiting until the last minute (to start this) is probably not a good idea. Assignment is due at the beginning of class on November 6th.