Assignment 3 (mandatory)
Airport Queueing

Jacob Trier Frederiksen (instructor)
& Anders Kalhauge (AKA, assisting)

Algorithms & Data Structures, Spring 2019

In groups:

Implement a priotitized queueing system for an airport. You can use any priority queue algorithm, but you must be able to argue that the time complexity is no worse than $O(\log n)$ for enqueue and dequeue respectively.

You should implement the priority queue in a setup that simulates passengers arriving to an airport, and passengers passing security. So you must think about the frequency of passengers in an airport, and how many are rejected (or delayed) in security – and things like that, to simulate the queues populating. So; like; random, time-of-day, ...

Passenger priority is assumed to be derived from properties like passenger category and arrival time (you decide the priority...):

- Monkey
- Late to flight
- Disabled
- Business class
- Family

A template for such a setup can be found here:

Airport Queue Template

NB: You can use the template, but you are strongly encouraged to think it out for yourself first, then – perhaps – seek inspiration in the provided code template.

Create a priority queue instead of the `NotPrioritisingPassengerArrayQueue` used there. Experiment with other values for producer and consumer. Try to add more than one consumer.

NB: we will spread the topics of priority queues over two weeks (weeks 9 & 10). Consequently, you may want to implement only part of the code, this week, and pick up on the remainder next week.

The solution accompanied with a description in a text file should be uploaded to the Peergrade website, no later than Tuesday March 15th, 08:30. Please ask if you are in doubt about any of this.