

Queue, Efficiency - Airport Simulation

Delaware Technical Community College

Program Specifications:

A queue is another ADT we see everywhere in the real world. Just like the stack, it is a fundamental data structure for computers as well - as processes and threads are often organized by queues so that the CPU organizes what instructions to execute.

Just as a stack is an intuitive way to organize card games, a queue could be an intuitive way to simulate anything where a line forms and there is a first in, first out order of operations.

Implement a queue using a circular array or doubly linked nodes to simulate a simple airport with the following rules:

- There are 2 runways
- Only one airplane can take off at any given turn
- The airplanes alternate between runways - unless one runway is empty.

Secondly, write a brief justification (minimum 250 words) why you chose to use a circular array or doubly linked node implementation. Your justification should clearly explain your choice in terms of big O notation for the efficiency of storage, runtime, and/or major operations that occur in this program. Include references to specific line numbers in your code where appropriate.

Notes:

- The driver class should include two queues - one for each runway
- The program should be robust and correctly deal with an empty queue; in other words, you should not simply assume that there are equal number of airplanes in each queue

To Do:

- Design and implement an Airplane class
- Design and implement a driver class that simulates this airport
- Write a justification of why you chose a circular array or doubly linked node implementation for the queue ADT
 - Please include the justification as a comment on the top of your driver class