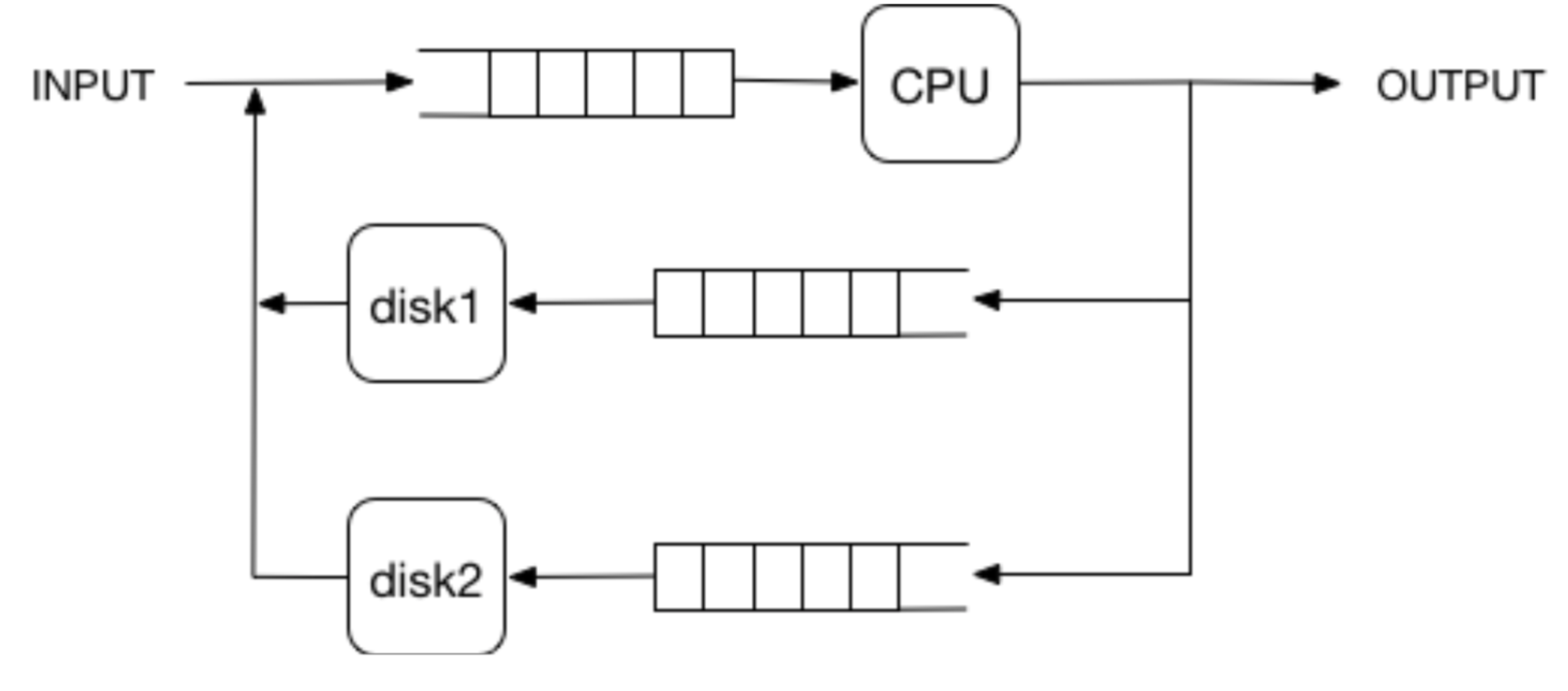
Matthew Cohen

Lab 1

Discrete Event Simulator

The purpose of this project is to simulate how processes move throughout a computing system using a scheduler. The program parses a required config file to load the associated values to use in the simulation. The config file should be in the following format or the program terminates without beginning the simulation. The config format should include the following names; seed, init\_time, fin\_time, arrive\_min, arrive\_max ,quit\_prob, cpu\_min, cpu\_max, disk\_min, disk\_max, disk2\_min, disk2\_max. Each name should have a corresponding integer value (expect quit\_prob) separated by a space.

The program simulates a CPU scheduler by storing process/events by using queues as the data structure. A priority queue is used to hold the events as they come into the system and before they are sent to a device according to their event type. The priority queue is sorted by the lowest time to ensure that each event gets processed accordingly. The CPU and both disks also have their own FIFO queues to hold events when the CPU or disk device is busy and to be worked on when they become free. The event simulation can be summed up by looking at the illustration below to understand the proper flow of the program. 

I decided to design the CPU as its own class to abstract and hide its inner workings, so all the main loop has to do is pass an event to the CPU and it knows how to handle it accordingly. I took this same approach to design the disk class. I found that taking the OOP approach to design this project made the main simulation loop easier to understand and follow the passing of events to the different devices.

I first tested the program by making sure invalid config file configurations cause the program to immediately terminate. After a valid configuration file is confirmed to be present, I began changing its values to see what type of results it generated. I was then able to read the stat file to make sure the changes in the config file produced values that made sense.