All students should read chapter §5.3.3 (up to and including the section *Back-of-an-Envelope Calculations*) and §5.3.5 for this assignment as well as future lectures.

As usual, question 1 should get double coverage, the remaining members of the group should do one of the remaining two questions. It **would be wise** to be sure solutions to both programming questions exit for review.

1. Read and summarize (in writing, so there is something for your group members to review) `this paper` for your group. If you liked CSCI220 (*Data Structures*), you'll like this paper. This may require a little bit of research into data structures with which you aren't already familiar. You don't have to provide detailed implemention or theory explanations, a good "this is how it works" will suffice.
2. From your group's solutions to `lga-nes.pdf` choose a solution for SiS **that you did not write** and do the following.
   (a) Make the following changes to the SiS simulation:
      i. Let a be the inventory three days **after** an order is placed. Track $\bar{a}$, and the standard deviation of a using Welford's equations (Theorem 4.1.2). Report both at the end of the simulation.
      ii. Now, let b be the inventory three days **before** an order is placed. Like a, track and report b's mean and standard deviation using Welford's equations.

      Even better, use your group's `binner` from `lga-nes.pdf` to show density histograms for these measures.

   Be careful to calculate a and b only when orders are made!

   (b) Finally, report your experience back to your learning group: Was the initial construction of the simulation easy to understand and easy to modify for this question's specific needs? **If not** be specific about what could have been done differently in the original code. **If so** comment on how scalable the design is, you made a small modification to the simulation — would making substantial changes (another 10 event types and another 15 variables tracked) give you pause? Why?
3. From your group's solutions to `lga-nes.pdf` choose a solution for SSQ **that you did not write** and do the following.
   (a) Make the following changes to the SSQ simulation:
      i. Allow for a command line provided parameter $n > 2$ that specifies the number of servers (this is no longer a **single** server queue!). All servers follow the usual rules (non-preemptive, one job at a time, feedback for completed jobs stays the same) and jobs are allocated to servers from a single "waiting" queue of jobs. As in an SSQ, no server should be idle when there are jobs waiting to be completed.
      ii. Now, let V be the probability that at least half ($\geq \frac{n}{2}$) the servers are busy. Measure this probability two different ways:
         a. **Randomly sample** this probability 500 times (give or take a few samples) while the simulation runs. Use Welford's equations (Theorem 4.1.2) to track and report the average and standard deviation at the end of the simulation as V_{rs}.
         Clarification: "random sample" does not mean sampling every $\frac{\tau}{500}$ time units. And **do not** place 500 "`SampleVrs`" events into the `eventList` at the beginning of the simulation — look back to your group's `lga-uniform-arrivals.pdf` solutions (or Example 3.1.2 of the text) for a hint for an "exponentially more efficient" way to sample this probability.
         b. Using Welford's **sample path** equations (Theorem 4.1.4) track this probability every time the number of busy servers changes in the simulation. As for V_{rs}, report the average and standard deviation at the end of the simulation as V_{sp}.

         Even better, use your group's `binner` from `lga-nes.pdf` to show density histograms for these metrics.

   (b) Finally, report your experience back to your learning group: Was the initial construction of the simulation easy to understand and easy to modify for this question's specific needs? **If not** be specific about what could

have been done differently in the original code. **If so** comment on how scalable the design is, you made a small modification to the simulation — would making substantial changes (another 10 event types and another 15 variables tracked) give you pause? Why?