CS3000: Algorithms & Data — Summer 2025 — Laney Strange

APP 4

Due: May 20th, 2025 @ 11:30am via Gradescope

Name: Sample Solution

- APPs will be assigned towards the end of roughly two lectures each week. You'll put together a solution to a short problem that we'll all use in the following lecture. We'll have time set aside to do these in class, or you can work on your own.
- You may handwrite your solutions, or typeset them in LATEX or another system.
- APPs will be graded on completeness. They must be submitted by 11:30am (just before lecture) on the due date. They will not be accepted late, but we drop 3 of them (out of 8 total).
- Collaboration is strongly encouraged for APPs!

Problem 1.

Show how to implement a queue using two stacks. Describe how you would implement the procedures ENQUEUE and DEQUEUE using only Stack operations PUSH, POP, and/or STACK-EMPTY from two stacks (no need to write full pseudocode, though). Can your two-stack solution be implemented with the same asymptotic run-time as the original queue structure?

Solution:

Call the two stacks S_1 and S_2 . Here's how we'd implement the two key procedures:

- ENQUEUE: Push the new element onto *S*₁.
- DEQUEUE: First, check if S_2 is empty. If not, pop and return the top of S_2 . If so, pop all the elements off S_1 and push them onto S_2 . Now the oldest element will be on top of S_2 . Pop and return it.

The ENQUEUE operation as described takes O(1) time, and the DEQUEUE takes O(1) in the best case (S_2 is non-empty) and O(n) in the worst case (when we move everything from S_1 to S_2 .)