

Assignment 5 Analysis

Ray Ding

If you worked in a partnership:

- What is your programming partner's name and which of you submitted the program files to Gradescope? Briefly discuss the pair programming experience, including:

Xiyao Xu, we both submitted program files to Gradescope.

- How well did you apply the techniques of pair programming (as explained on this page) to this assignment?

We try to complete this assignment separately, but we discuss frequently if we encounter problems. This is the competition and collaboration mode.

- What percentage of the program code did you write using these techniques?

I wrote all the Java class files with these techniques. However, I let Xiyao write all the test files.

- About how many hours did you spend completing the programming and testing portion of this assignment? Evaluate your programming partner.

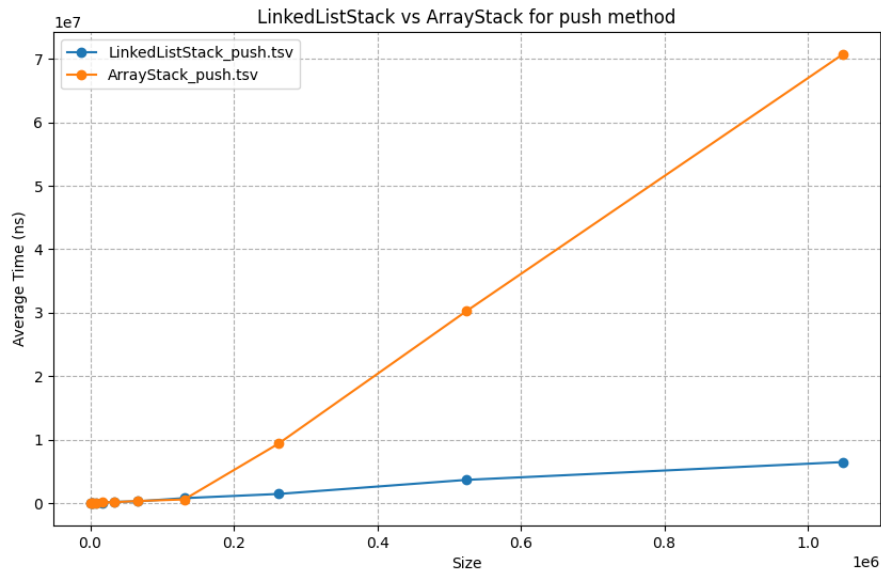
I spent 8 hours to write them. And spend another 4 hours for the revision after code review and writing analysis document.

- Do you plan to work with this person again?

Yes. He's motivated for labs and assignments.

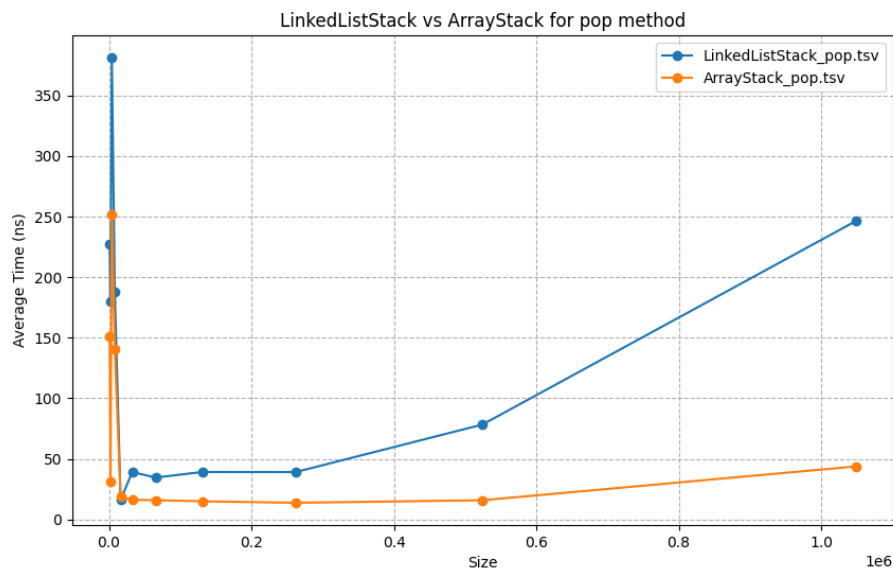
For everyone:

- Compare the running time of the push method. What is the growth rate of the method's running time for each stack class, and why?



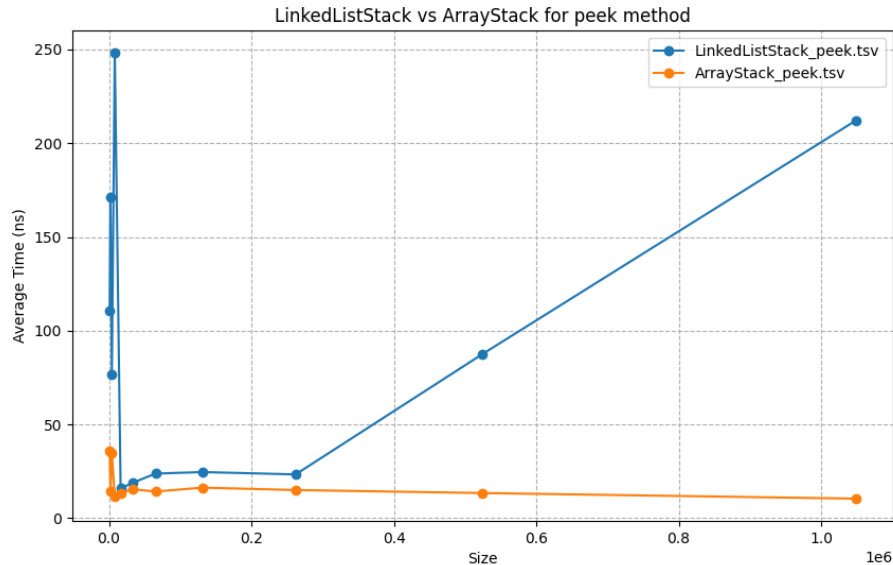
I test the push method inside the for loop with iteration of i from 0 to size - 1. The running time increase linearly with the size of the data. The push method itself should both be $O(1)$. The for loop iterate N times, so the total running time is $O(N)$ for both. Interestingly, as the size increases, the ArrayStack implementation's time cost grows linearly due to array resizing, while the LinkedListStack maintains a constant running time for the push operation.

- Compare the running time of the pop method. What is the growth rate of the method's running time for each stack class, and why?



The running time for both stacks are $O(1)$. This is because pop method only involves the first element in the stack, regardless of the stack's size.

- Compare the running time of the peek method. What is the growth rate of the method's running time for each stack class, and why?



The running time for both stacks are $O(1)$. This is because peek method only involves the first element in the stack, regardless of the stack's size.

- Based on your timing experiments, which stack class do you think is more efficient for using in your WebBrowser application? Why?

Based on the performance data collected, the ArrayStack implementation seems to be the more suitable choice for the WebBrowser application. All three core methods—push, peek, and pop—demonstrate $O(1)$ time complexity, which is desirable for stack operations. When examining the pop and peek methods, ArrayStack consistently outperforms LinkedListStack across various stack sizes, indicating a more efficient and stable behavior for these operations. While the push method does exhibit a performance hit due to array resizing when the stack grows beyond its initial capacity, such scenarios are unlikely to be a frequent concern in a typical web browsing session. It would require an exceedingly large number of webpages—over 16,384, for instance—to be visited in the same session before resizing becomes a factor. Considering these points, ArrayStack appears to be the optimal choice for managing web browser navigation.