

Homework 3 – Due Oct. 6th 23:59, KST

Instructions: Complete the implementation and turn it in before the due date. Any deviations from the instructed deliverable format will result in a deduction of grade. DO NOT COPY OTHER'S WORKS!

In this assignment, you will be implementing a data structure using only an array as the underlying storage. Do not use or build nodes to implement a linked list. It should support the following four major modification operations. See HW3.java for more instructions.

- Front insertion/deletion: Inserting/deleting a new element to/at the front of the array.
- Rear insertion/deletion: Inserting/deleting a new element to/at the end of the array.

In addition to these modifiers, you should also implement the following operations as well.

- Return front: Return the first element.
- Return rear: Return the last element.
- Size: Returns the number of elements stored in the array.
- Get at: Return the element at the given index.
- Reverse: Reverse the order of stored elements.
- Clear: Delete everything.

Rubric: Grading will be based on, but not limited to, the following criteria.

- Documentation (20 points): For each of the four modifier methods, you should provide a header comment that provides a big-O time complexity analysis (with respect to the current number of elements stored) of your implementation. In addition to the big-O's, provide a brief explanation of how you arrived at that conclusion.
- Correctness (70 points): Your implementation should behave as specified above. It should be error-free and use appropriate features of Java, such as generics. This score will be determined by a grading script that I prepare.
- Additional question (10 points): Write a paragraph discussing possible approaches to minimize the complexity of your methods. What type of data structures would you use? What specialized algorithms can you think of?

Deliverable: A single HW3.java source file. DO NOT provide a zip file. Just submit a single Java file with no package structure.