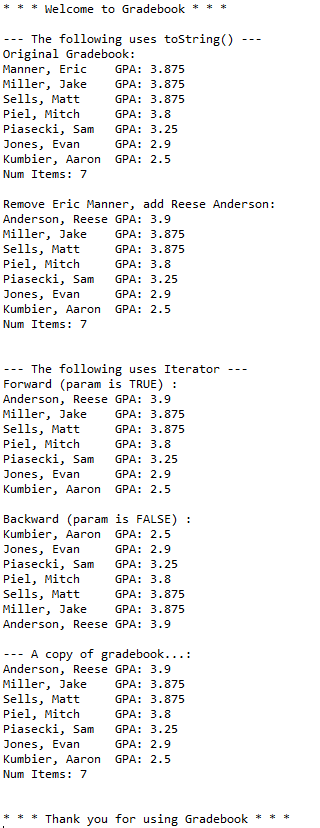
Assignment 9 Reflection:

All of the requirements for Assignment 9 are complete.

My biggest issue was figuring out the algorithm to copy a DLSortedList. However, I was just overthinking and realized it was pretty simple. It took me a little while to decide how to implement the add method as well, but I didn’t struggle all that much.

Was there an easier way to implement the add method? I feel like it should be based on your comment indicating that a header and trailer make add/remove implementations simpler.

**Gradebook.java**



**Test Scenarios:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| gpa | |  |  | | --- | --- | | Test | Output | | -1 | 0 | | 3.5 | 3.5 | | 4.2 | 0 | |

* A gradebook contains Piasecki, Sam GPA: 3.25 . Remove Piasecki, Sam GPA: 2
  + Sam Piasecki remains on the list because the GPAs do not match
* Add a student to the front of the gradebook
  + Successful
* Add a student to the end of the gradebook
  + Successful
* Print gradebook without any students created
  + Successful
* Add student with same gpa but different name
  + Students with same gpa are sorted by last name, first
* Test iterator(true)
  + Gradebook prints high gpa to low, as it should.
* Test iterator(false)
  + Gradebook prints low gpa to high, as expected.