

CSCI 2125

Homework 2: Linked Lists and Lazy Deletion

An alternative to the deletion strategy we discussed with our linked lists is to use lazy deletion. To delete an element, we merely mark it deleted (using an extra bit field). The number of deleted and nondeleted elements in the list is kept as part of the data structure. If there are as many deleted elements as nondeleted elements, we traverse the entire list, performing the standard deletion algorithm on all marked nodes.

1. List the advantages and disadvantages of lazy deletion. **(20 points)**
2. Modify the MyLinkedList implementation we wrote together in lecture to implement the standard linked list operations using lazy deletion. **(60 points)**
3. Write a JUnit 5 tester class to prove that your code functions properly **(20 points)**
4. **BONUS (20 points):** Modify the MyLinkedList with lazy deletion to be a doubly linked list, and where the iterator has methods “hasPrior()” and “getPrior()”

Submission

You will add, commit, and push your program to Gitlab. Label your homework folder "HW2" in your repository. If you attempt the bonus, make a subdirectory called “bonus” which contains a FULL implementation that can be run and tested.