

Submit your code and any other answers to Canvas. The lab is open book.

Files to submit

Question 1

DoublyLinkedList.java

Question2

CircularlyLinkedList.java

Assignment

1. Implement a method called `middle` for finding the middle node of a doubly linked list with header and trailer sentinels by “link hopping,” and without relying on explicit knowledge of the size of the list. In the case of an even number of nodes, report the node slightly left of center as the “middle.” Your method should return a reference to the middle node as output. Implement a main method for testing your method. Initialize your linked list to contain Integers 1,2,3,4. Once you find the middle node, display its element and make sure it is 2. Repeat for a doubly linked list that contains 1,2,3,4,5 and make sure the middle node you found contains 3.

2. Given a circularly linked list `L` containing an even number of nodes, implement a main method that splits `L` into two circularly linked lists of half the size. You can use the size variable of the linked list. Test your method for a linked list that contains Integers 1,2,3,4,5,6. Print the contents of the linked lists before and after the split.