

# Lab 17

## Linked Lists

---

### Objectives of this Lab

1. Practice implementing Linked List methods
  2. Implement a simple book system
- 

### Description

In this recitation we will implement a singly linked list class to get a better understanding of this reference-based data structure. The interface you will implement is very similar to that provided by ArrayLists, but the way in which they are implemented is very different.

### Phase 1 - Build the project

Create a Lab17 project and download the following files into that project.

- [Book.java](#)
- [LinkedBookList.java](#)
- [BookNode.java](#)
- [R20.java](#)

You will also need to download a text file containing books that appeared on the New York Times list of top 10 Fiction Hardback Books. This file contains the books name, author, and number of pages.

- [top10.txt](#)

### Phase 2 - File Descriptions

#### Book.java

Book.java is nothing out of the ordinary - it's just keeps track of the book's title and its author, as well as the number of pages. It has a few getter and setter methods, but overall this class is very straight forward.

## **BookNode.java**

BookNode.java is a little more interesting. A BookNode is an object that stores a Book in its book instance variable, and has a reference to the next BookNode in its next instance variable. If the node is the last one in the sequence, its next variable will be set to null. There are also methods for setting the next node and getting the next node, as well as a method for returning the book that the node contains.

## **LinkedList.java**

Now let's look at the LinkedList class. You'll see that the instance variables for this are very simple. Only a BookNode called head, which refers to the first item, and an int to keep track of the size of the linked list. This is all we need, since each element contains a reference to the next one.

## **R20.java**

R20.java is your test program to run and show the output to the TAs.

## **Phase 3 - Implementation**

Your task is to implement the first add method, which adds a book at the end of the list (not the beginning).

This method will require you to iterate through the list. You can do this either with a while loop or a for loop. If using a for loop, think about the three components that make up the loop:

```
(initialization; are we done?; operation to reach the next item)
```

Many of the operations that we are trying to implement here, can be reasoned through by drawing pictures. Use that to reason how far to go, when to set which pointers, and when to delete pointers.

---

## Turn In

Show your work to your TA before submitting to GitHub. Submit all of your files with correct headers to GitHub here: <https://classroom.github.com/a/Cb0GNgtl>