# Data Structures & Concurrency : Lab 1: Week beginning 11thSeptember 2023

# Using Java Collections Framework

1. Write a program that creates a LinkedList of String objects.

(i) Add the following names to the list, using the add method (or addLast method) :

Kay, Jay, May, Fay

(ii) Use an iterator to insert “Ray” after May.

(iii) Then delete Jay

(iv) Use an iterator to display all the names on the list

From the beginning of the list

From the end of the list

(v) Add Kay again. Use an iterator to display all the names. Note that duplicates are allowed.

2. Insert the names given above in Q1 in a set. Use the two different Set implementations – HashSet and TreeSet. Try inserting the same name twice. Note that duplicates are not allowed.

Use an iterator to display the names in the sets.

What is the output for

1. HashSet implementation?

2.TreeSet implementation?

What do you notice about the output?

3. Write a program that reads text from a file and breaks it up into individual words.

See the source code “filehandling.java” for code to read from a text file, hamlet.txt on the desktop, and break the text into words.

Put hamlet.txt on the desktop to run this.

Insert the words into a TreeSet. At the end of the input file, print all words, followed by the size of the resulting set. Use an iterator.

This program determines how many unique words a text file has.

What happens if you use a HashSet instead of a TreeSet?

4. Add code to the program you wrote for Q.3 so that a Map is used. The key is the word and the value is the line number.

If you store just one line number for each word, then the last occurrence of the word will be in the map. E.g. the word “to” is repeated in a few lines.

To store all the line numbers on which a word occurred, you could use a List or a Set of line numbers – which would be better? Update the code to do this.

Also perform a search on the Map for a key value – what method allows you to search for a word?

**Explanation of Outputs:**

1. **HashSet Output**: No duplicates; order is not guaranteed. The output may appear in any order.
2. **TreeSet Output**: No duplicates; elements are sorted in natural order (alphabetically).

**Using HashSet vs TreeSet:**

* Using a HashSet results in no guaranteed order, while a TreeSet provides sorted output.
* TreeSet is slower for insertion compared to HashSet due to sorting operations, but it allows for ordered traversal.