CSC2001F Data Structure 1 Assignment Report

Jing Yeh
YHXJIN001

To start the program:

Two versions of the main programs are included in the compressed archive file, named KnowledgeBaseArrayApp.java and KnowledgeBaseBinarySearchTreeApp.java respectively. To run, type in the command ”make” to compile the .java files and then ”make runArray” or ”make runTree”

```
make
make runArray
make runTree
```

which should display a text-based menu like this:

```
The Common Sense Library: Where you can find all the common sense knowledge on
Earth!

How could we assist you?
[1] Load a knowledge base from a file
[2] Add a new statement to the knowledge base
[3] Look up definition of a term
[4] Look up how confident we are about the definition of a term
[5] Save the knowledge base to a file;
[q] Quit
```

Overall OOP Design

Summary:

The following classes are used:

1. *KnowledgeBase*, which is an abstract class extended by *KnowledgeBaseArray* (The traditional array version of the program) and *KnowledgeBaseBinarySearchTree* (The unbalanced binary search tree version of the program)
2. The *Entry* datatype is a representation for each line of statement, with the **term**, **sentence**, and **confidence score** stored separately within the datatype's class definition.
3. A *UserInterface* class is used to display the text-based user interface for both *KnowledgeBaseArrayApp* and *KnowledgeBaseBinarySearchTreeApp*, thereby improving code resusability.

The KnowledgeBase abstract class

The KnowledgeBase abstract class defines the signatures of common methods shared between KnowledgeBaseArray and KnowledgeBaseBinarySearchTree, such as search, insert, save, etc (Refer to javadoc for more details).

Implementation of the traditional array

The specification states that the KnowledgeBase must be able to store at least 100 000 statements. Therefore, an array of size 100 100 is initiated in the constructor of KnowledgeBaseArray. The extra 100 space is allocated to allow rooms for adding individual statements.

addFile(), insert(), search(), save() methods all implement the standard, for-loop linear search algorithm, as sorting the array is prohibited.

Implementation of the binary search tree

The binary search tree follows the convention of placing data ranging in magnitude from left to right. Data are compared based on the value of **term** carried by each EntryNode, and using the built-in java string compareTo method.

The tree is not balanced, and therefore has a worst case complexity of $O(N)$. However, the worst case unlikely to happen as the size of the dataset is huge and relatively spread out (The average case is $O(logN)$, which is more likely). The node of the tree, **EntryNode** inherits directly from the Entry datatype. This is done to enhance code reusability and for reducing unnecessary instantiation of Entry.

addFile(), insert(), and getSize() all implement the standard recursive algorithms, with the exception that save() on top of that uses a slightly modified version of in-order traversal technique.

Implementation of Creativity

Several methods are added to enhance user experience. For example, users can save the knowledge base they've created into a txt file, with the save() method that uses BufferedWriter for better optimisation.
Protections against error inducing inputs are also included in the code.

Testing

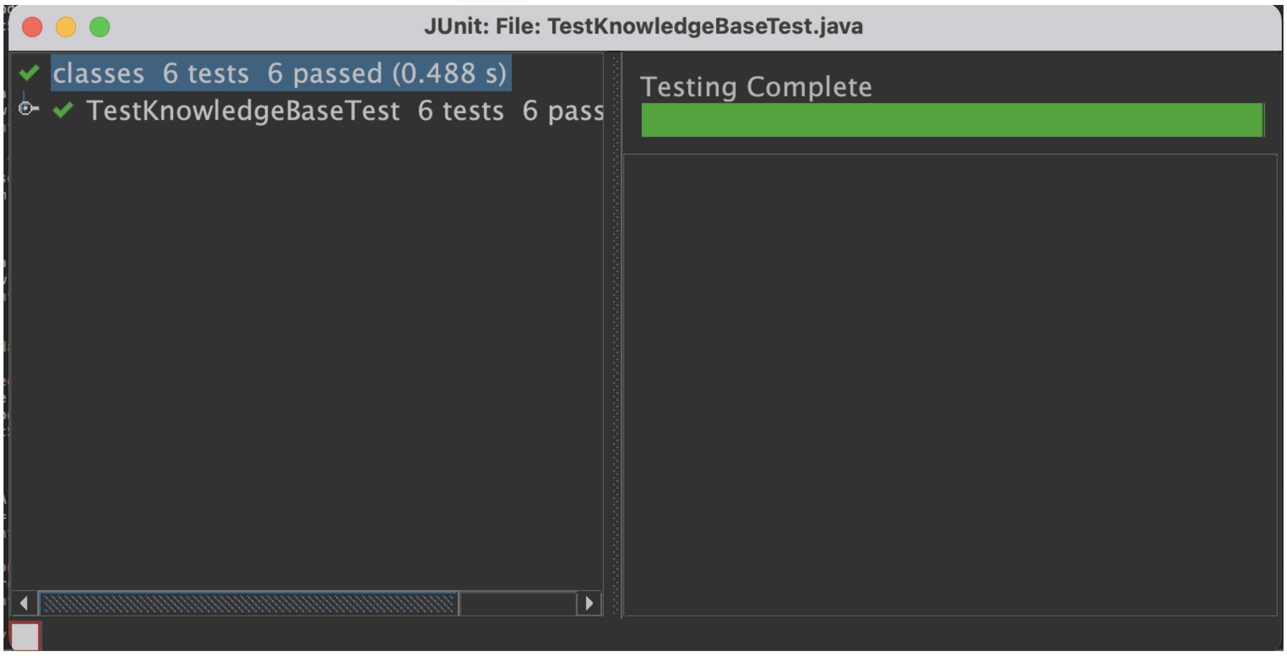
The *TestKnowledgeBaseTest.java* is run to test populating, updating and searching the *KnowledgeBaseArray* and
KnowledgeBaseBinarySearchTree.

Testing Protocol Explanation:

1. **Insertion Tests:**
    - Verify that the knowledge base can correctly insert entries, whether using an array or a binary search tree.
    - Check the length/size of the knowledge base to ensure it reflects the number of inserted entries.
2. **Update Tests:**

- Insert an entry with term "term1", then insert another entry with the same term "term1" but with a higher confidence score.
- Confirm that the update was successful by searching the entry and checking the updated values.

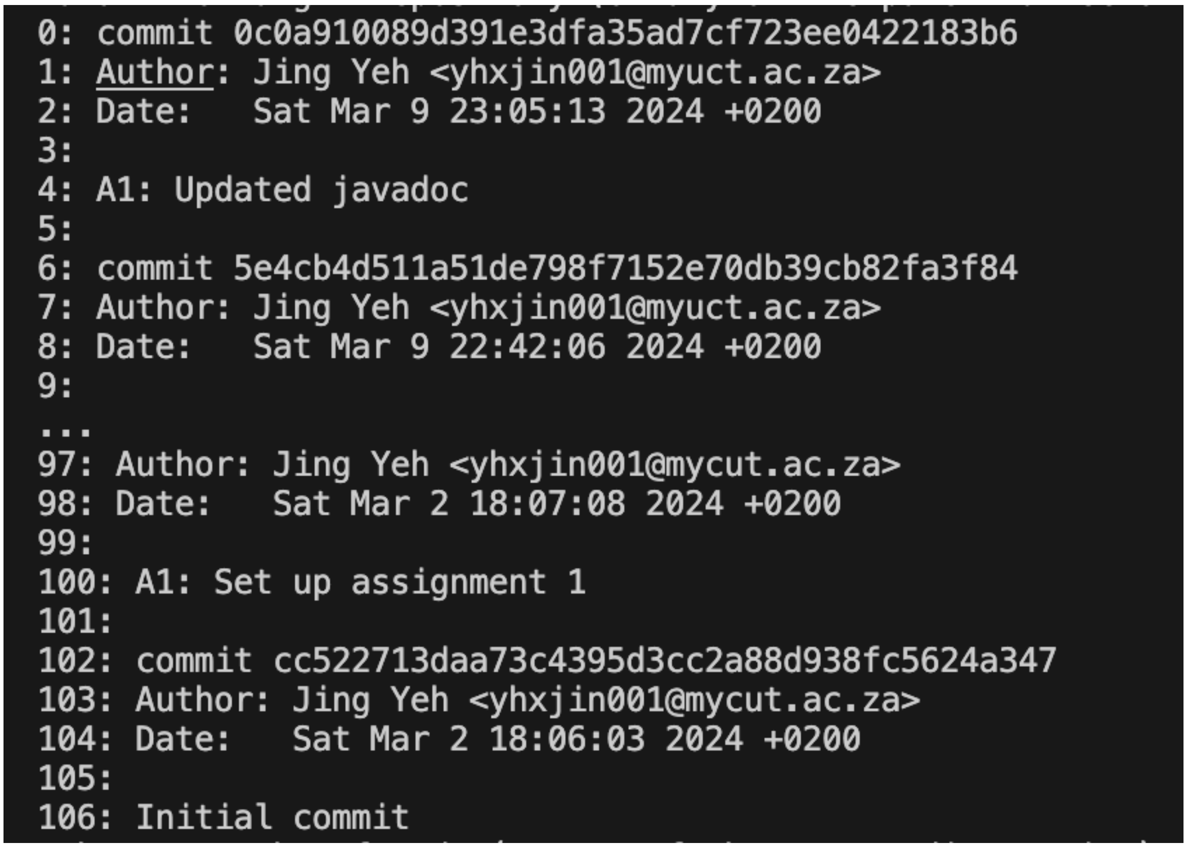
3. **Search Tests:**
   - Test searching for both existing and non-existing entries.
   - Ensure that the correct entry or `null` is returned, depending on whether the term is present.



```
JUnit: File: TestKnowledgeBaseTest.java
✓ classes  6 tests  6 passed (0.488 s)
  ✓ TestKnowledgeBaseTest  6 tests  6 pass
Testing Complete
```

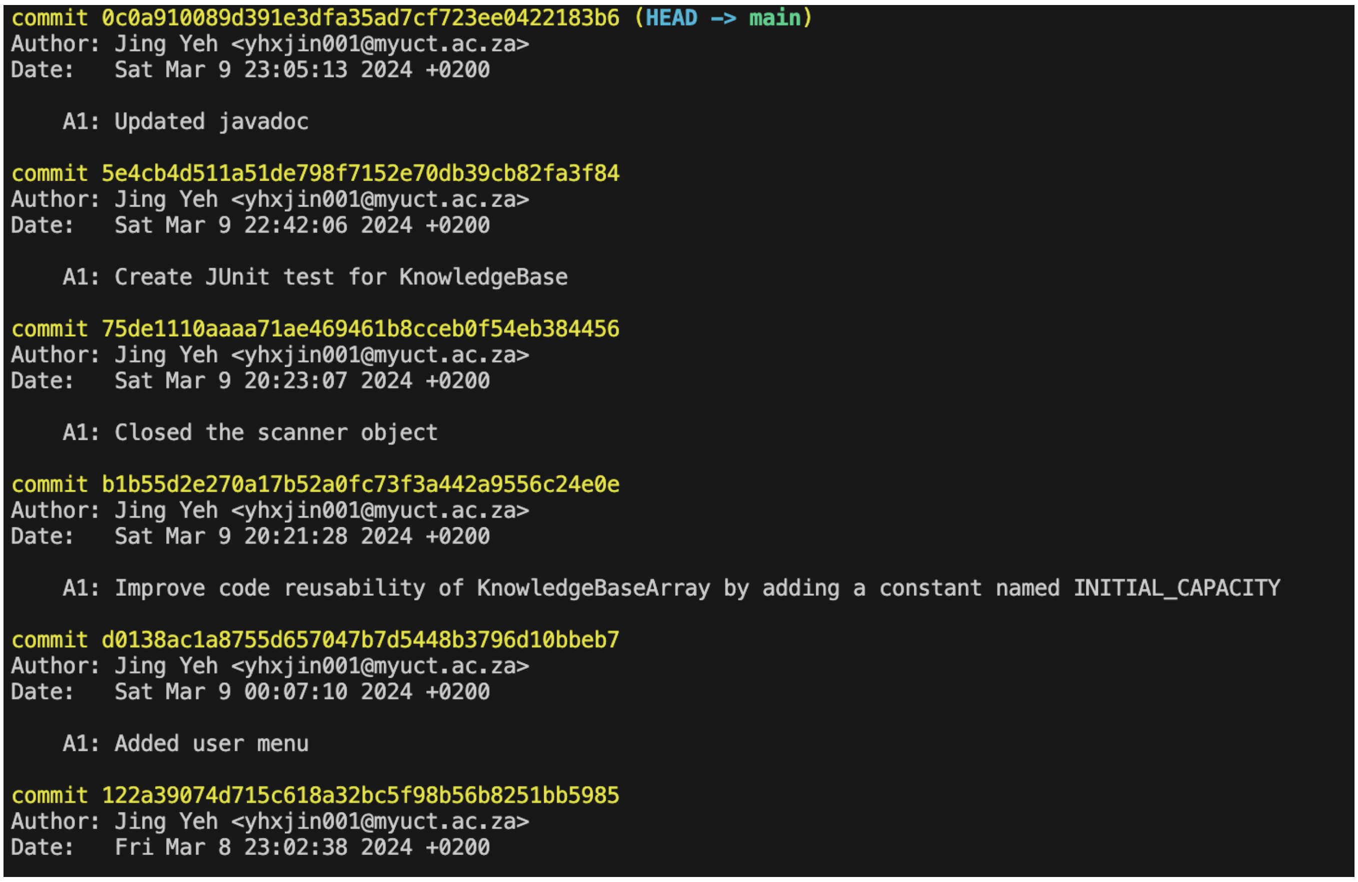
Git log

The first and last 10 lines as specified is included here:



```
0: commit 0c0a910089d391e3dfa35ad7cf723ee0422183b6
1: Author: Jing Yeh <yhxjin001@myuct.ac.za>
2: Date:   Sat Mar 9 23:05:13 2024 +0200
3:
4: A1: Updated javadoc
5:
6: commit 5e4cb4d511a51de798f7152e70db39cb82fa3f84
7: Author: Jing Yeh <yhxjin001@myuct.ac.za>
8: Date:   Sat Mar 9 22:42:06 2024 +0200
9:
...
97: Author: Jing Yeh <yhxjin001@mycut.ac.za>
98: Date:   Sat Mar 2 18:07:08 2024 +0200
99:
100: A1: Set up assignment 1
101:
102: commit cc522713daa73c4395d3cc2a88d938fc5624a347
103: Author: Jing Yeh <yhxjin001@mycut.ac.za>
104: Date:   Sat Mar 2 18:06:03 2024 +0200
105:
106: Initial commit
```

However, here's one that more clearly demonstrates usage of git.



```
commit 0c0a910089d391e3dfa35ad7cf723ee0422183b6 (HEAD -> main)
Author: Jing Yeh <yhxjin001@myuct.ac.za>
Date:   Sat Mar 9 23:05:13 2024 +0200

    A1: Updated javadoc

commit 5e4cb4d511a51de798f7152e70db39cb82fa3f84
Author: Jing Yeh <yhxjin001@myuct.ac.za>
Date:   Sat Mar 9 22:42:06 2024 +0200

    A1: Create JUnit test for KnowledgeBase

commit 75de1110aaaa71ae469461b8cceb0f54eb384456
Author: Jing Yeh <yhxjin001@myuct.ac.za>
Date:   Sat Mar 9 20:23:07 2024 +0200

    A1: Closed the scanner object

commit b1b55d2e270a17b52a0fc73f3a442a9556c24e0e
Author: Jing Yeh <yhxjin001@myuct.ac.za>
Date:   Sat Mar 9 20:21:28 2024 +0200

    A1: Improve code reusability of KnowledgeBaseArray by adding a constant named INITIAL_CAPACITY

commit d0138ac1a8755d657047b7d5448b3796d10bbeb7
Author: Jing Yeh <yhxjin001@myuct.ac.za>
Date:   Sat Mar 9 00:07:10 2024 +0200

    A1: Added user menu

commit 122a39074d715c618a32bc5f98b56b8251bb5985
Author: Jing Yeh <yhxjin001@myuct.ac.za>
Date:   Fri Mar 8 23:02:38 2024 +0200
```