Project 3

Design

Diagram

Description automatically generated

Data structures are difficult to understand at times. I had to refresh my knowledge on linked list and incorporate doubly linked list for this project. The image above shows the original concept after reviewing with the power points and videos. Initially to start the project made small pseudo code for the double linked list. With some time to think about the Javadoc and pseudo code I started implementing the project.

Test Case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Prompt | input | Output | Excepted Output | Screenshot |
| Add  Cow in basic to front | cow | cow | cow | Graphical user interface, application  Description automatically generated |
| Get the last from the sorted list | n/a | shore | shore | Graphical user interface, application  Description automatically generated |
| Iterator on the basic until monkey | n/a | Monkey  The pointer was at monkey | Monkey  The pointer was at monkey | Graphical user interface, application  Description automatically generated |
| Iterate previous in sorted linked list | n/a | true | true | Graphical user interface, application  Description automatically generated |

Junit

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, text, website

Description automatically generated

Learning

Experience

Previous()returns the current node, below or the next before decrementing. Always had to make sure the image of each node adding from one node to the other. This is for the Iterator class, next was not hard to visualize. Previous() need to increment first to return the data. That was the initial challenge. Handle exceptions nothing out of the ordinary, I have been implementing exception for the last 2 project. Generic classes had to be in the background of implementation especially when call that generic class. Implementing double linked list was the trickiest part for myself. With the original image in design was at the for front of my mind to keep thing working. Setting node to the next or previous depending on adding and removing. After some trial and error, I manage to get the doubly linked list functioning. Finally, comparators where the first time utilized the class. I mainly used the compare method to compare two values from nodes.