

CMSC204 Assignment 1

Class: CMSC204

Program: Assignment #3

Instructor: Professor Kuijt

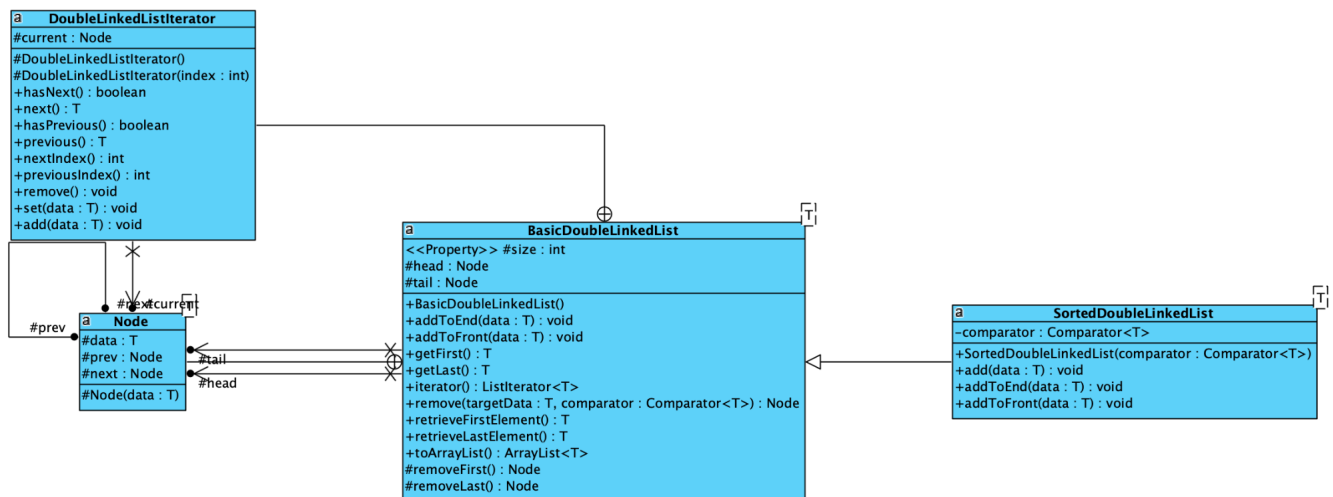
Summary of Description:

Assignment is to write a generic double-linked list class with an iterator, and a generic sorted double-linked list class with an iterator that inherits from your generic double-linked list class. GUI for assignment was provided to help visualize linked list.

Due Date: March 5, 2023

Integrity Pledge: I pledge that I have completed the programming assignment independently. I have not copied the code from a student or any source.

UML:



Screenshots:

Adding to basic list:

Adding to sorted list:

Doubly Linked List

Type of lists
☒ Basic ☐ Sorted

Add to List
 Element to Add:

Retrieve from List (deletes from list)
 Retrieved:

Get from List (doesn't deletes from list)
 Returned:

Remove from List
 To be Removed:

Iterator (upon add, retrieve or remove, restart iterator)
 Returns:

Contents of lists

Basic List	Sorted List
fifth	
fourt	
thir	
Sec	
First	
neg1	

Doubly Linked List

Type of lists
☐ Basic ☒ Sorted

Add to List
 Element to Add:

Retrieve from List (deletes from list)
 Retrieved:

Get from List (doesn't deletes from list)
 Returned:

Remove from List
 To be Removed:

Iterator (upon add, retrieve or remove, restart iterator)
 Returns:

Contents of lists

Basic List	Sorted List
fifth	s2
fourt	s3
thir	s4
Sec	sort1
First	
neg1	

Doubly Linked List

Type of lists
☒ Basic ☐ Sorted

Add to List
 Element to Add:

Retrieve from List (deletes from list)
 Retrieved:

Get from List (doesn't deletes from list)
 Returned:

Remove from List
 To be Removed:

Iterator (upon add, retrieve or remove, restart iterator)
 Returns:

Contents of lists

Basic List	Sorted List
fifth	s2
fourt	s3
thir	s4
Sec	sort1
First	

Doubly Linked List

Type of lists
☐ Basic ☒ Sorted

Add to List
 Element to Add:

Retrieve from List (deletes from list)
 Retrieved:

Get from List (doesn't deletes from list)
 Returned:

Remove from List
 To be Removed:

Iterator (upon add, retrieve or remove, restart iterator)
 Returns:

Contents of lists

Basic List	Sorted List
fifth	s2
fourt	s3
thir	sort1
Sec	
First	

Removing neg 1 from basic and s4 from sorted
 Iterator

Doubly Linked List

Type of lists

☒ Basic ☐ Sorted

Add to List

Element to Add:

Retrieve from List (deletes from list)

Retrieved:

Get from List (doesn't deletes from list)

Returned:

Remove from List

To be Removed:

Iterator (upon add, retrieve or remove, restart iterator)

Returns:

Contents of lists

Basic List	Sorted List
fifth	s2
four	s3
thir	sort1
Sec	
First	

School

Search or jump to...

[Pull requests](#) [Issues](#) [Codespaces](#) [Marketplace](#) [Explore](#)

ArminRezz / CMSC_204 (Public)

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

main CMSC_204 / Assignments / A3 /

ArminRezz commit cmcsc204 A3

..	
javadoc	commit cmcsc204 A3
.DS_Store	commit cmcsc204 A3
BasicDoubleLinkedList.java	commit cmcsc204 A3
BasicDoubleLinkedListTestStudent.java	commit cmcsc204 A3
SortedDoubleLinkedList.java	commit cmcsc204 A3
SortedDoubleLinkedListTestStudent.java	commit cmcsc204 A3

[Give feedback](#)

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#)

Lessons Learned:

What have you learned?

Learned how to implement linked lists, specifically a double linked list and a sorted double linked list.

What did you struggle with?

I struggled with some of the functions thinking of how to rearrange the nodes properly. Found that using a whiteboard was quite helpful for quickly visualizing stuff.

What would you do differently on your next project?

Nothing really probably wouldve spent more time optimizing

What parts of this assignment were you successful with, and what parts (if any) were you not successful with?

Successfull with all parts.