

Grade: Great job! 9/10. Please see comments below.

If one of the following is true, you will NOT get credits.

- ☐ The project is late.
- ☐ The algorithm or class design is missing.
- ☐ The project has errors.
- ☐ There are no comments (Javadoc format required) at all in your code.
- ☐ Wrong files are submitted.
- ☐ A project is a copy and modification of another student's project. (Both will receive 0.)

Software Development Project Rubric: Design is worth 20%; the rest is worth 80%.

Analysis

Note: There will be no credit if the software doesn't meet customer specification at all.

Does the software meet the exact customer specification?

Does the software read the exact input data and display the exact output data as they are shown in sample runs?

Does each class include all corresponding data and functionalities?

Design

Note: There will be no credit if the design is missing.

Is the design (a UML class diagram) an efficient solution?

Is the design created correctly?

Code

Note: There will be no credit if there are syntactic errors.

Are there errors in the software?

Are code conventions and name conventions followed?

Does the software use the minimum computer resource (computer memory and processing time)?

Is the software reusable?

LinkedList:

```
/**
 * Concatenates two different strings from a LinkedList.
 */
```

```
public LinkedList concat(LinkedList string) throws ListIndexOutOfBoundsException
{ //newly made Character
```

This method should combine the content of this Linked string with the content of the other linked string that is passed into this method, return a new linked string with it. In your method, this linked string is modified through its alias concat. The insertion/deletion methods we made in class change this list all the time. Therefore, they are not appropriate used for concat and substring method.

```
    LinkedList concat = new LinkedList();
    concat = this;
    for(int i=0; i < string.length(); i++)
    {
        concat.insertAtBack(string.charAt(i));
    }
    return concat;
}
```

Please see an example:

```
public LinkedList concat(LinkedList other){
    if(this != null && other != null){
        String content = "";
        Node<Character> current = this.head;
        while(current != null){
            content += current.getItem();
            current = current.getNext();
        }

        current = other.head;
        while(current != null){
            content += current.getItem();
            current = current.getNext();
        }

        return new LinkedList(content);
    }
    return null;
}
```

Debug

Are there bugs in the software?

Documentation

Note: There will be no credit if comments are not included.

Are there enough comments included in the software?

Class comments must be included before a class header.

Method comments must be included before a method header.

More comments must be included inside each method.

All comments must be written in Javadoc format.