**CSCI 301  
Computer Science 2  
Fall, 2018**

|  |  |  |
| --- | --- | --- |
| https://web.stcloudstate.edu/bajulstrom/icons/book069.gif | **Programming Project 6 Due Thursday, October 11** | https://web.stcloudstate.edu/bajulstrom/icons/book069.gif |

TITLE

**PROCESSING A LINKED LIST RECURSIVELY**

INTRODUCTION

As we saw in [Project 4](https://web.stcloudstate.edu/bajulstrom/cs301/projects/p4f18.html), a **concordance** of a text is an alphabetical listing of the words that appear in the text with the number of times each word appears. Concordances summarize the frequencies of words in texts and are used in statistical analyses of authors' works, to determine authorship of disputed works, and to see how an author's writing changes over time.

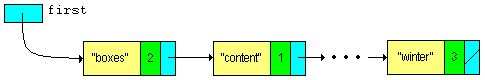
DESCRIPTION

Modify the Concordance class in the earlier project to perform as many operations as possible **recursively**. For this project, the necessary operations are these:

* The default constructor
* The destructor
* insert(word) - Insert word into the concordance. If word is already in the concordance, increment its count.
* length() - Return the length of the invoking concordance; that is, the number of distinct words that it contains.
* Output - Overload the "<<" operator to write the invoking concordance to an output stream.
* get\_node(word,count,link) - Return a pointer to a new Node that contains a word, its count, and the pointer value link.

Iplement the destructor and the operations to insert a word, determine the length of the concordance, and write out the contents of a concordance **recursively**. Thus each of these requires **two functions**: a public member function that can be called by the client program and a private recursive member function, called by its public function, that does the actual work.

The problem itself---building and reporting a concordance of a text read from an input file---remains as described in [Project 4](https://web.stcloudstate.edu/bajulstrom/cs301/projects/p4f18.html), whose entire main program can be re-used here.



HAND IN

See [About Programming Assignments](https://web.stcloudstate.edu/bajulstrom/cs301/handin.html) for a description of what to hand in: design document, user document, code, tests, and summary.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | https://web.stcloudstate.edu/bajulstrom/icons/note04.gif | Be sure that your **full name**, the **project number**, and the **due date** appear on every part of the project; print on **one** side of each page; and **staple** the pages together. | https://web.stcloudstate.edu/bajulstrom/icons/note04.gif |  |