

Kim Cooperrider
CS1550
Fall 2013
Project 1

This project was a thorough and much-needed reintroduction to the C programming language for me. It was certainly slow to begin coding and required more time than I had originally planned for, but I feel confident now that I am familiar enough with C to perform well on the next project.

My initial development phase began by plotting out my program's structs. I designed my node struct to contain the number of children contained, a boolean indicating whether or not the node is a leaf node, an array of pointers to node children, an array of pointers to course lists, and a pointer to a nextLeaf node. I also designed a struct called item, loosely based around the LinkedList item but with more course-relevant data. Leaf-level nodes point to an item that begins their respective LinkedLists of courses.

Next, I worked through the pseudocode referenced by the course homepage, but found it difficult at times because I prefer to work with 0-indexed arrays. Ultimately, I rewrote most of the code in a way that made more sense to me, although the pseudocode was helpful in getting started with the fundamental functions.

A major challenge I faced concerned the input buffer and strings. At times, I struggled with forgetting to clear the buffer after a print statement; this small error did cost me significant development time. Additionally, once I completed the basic functions of the system (insert, split, etc) and moved onto the integration of course data, I had some issues with pointers when reading in data. Fortunately, I was able to overcome these problems by properly allocating space for and copying strings.

Unfortunately, I found myself spending a large amount of time completing the basic functions and getting them to work with course data elements within the linked lists. Though I created some neat structs to map objects with their indices in arrays for searching, I did not have enough time to fully implement all of the remaining functions as I had planned. After scraping through the GPA calculations, I was left with the *top* function, which was more demanding than I had anticipated.

I am confident that if I had better managed my time, I would have finished this project, but I am glad to have completed it as much as I did.