Computer Science 211

Data Structures and Algorithms

Fall, 2016

Community College of Philadelphia Computer Science

Lab Report – Week *[4]* - *[Stack Programming Assignment]*

[Leo Vergnetti]

[CSCI 211] [Fall]

Assignment Analysis and Design

The assignment was to create software to utilize a stack to reverse the order of a String array. I designed three classes to do this: Stack class, StackElement class, and StackProject class. See attached UML for detailed information on each. The StackElement class stores the String data, and a pointer to the next Element (if any) in the list. It also contains accessors and mutators for these datamembers. The Stack class contains a pointer to the next (if any) element in the list, as well an int for size of the list. It includes a null constructor, and a String constructor, as well as methods to push elements to the stack, pop them from the stack, and print the contents of the stack. Finally the StackProject class demonstrates how this class is used. It creates an array of String objects, and a new Stack object and prints the array from the project (not the Stack class). In order to clearly illustrate how the class works, the null argument constructor is called, and, the each element in the array is added to the stack via a push method inside of a while loop. The stack's print method is called, showing each item in the stack, and finally the pop method is called inside of a while statement adding each element in the stack back into the array. Finally, the array's print method is called showing the Array in it's new (reversed) order.

Assignment Code

Source code included in attached file.

Assignment Testing

The testing for this class consisted of ensuring it worked with various string arguments. Care had to be taken when the list became empty, else the size may become negative. A decision had to be made for what to do when pop was called while the list was empty, currently a null String is returned, indicating the lack of a list.

Assignment Evaluation

This assignment working with linked lists was a little easier than the State project assignment, though I'm not sure if that's due to growing prowess with Linked Lists, or the nature of this assignment in general. In either case all seems to be working well, and the Lab illustrated the advantages of lists very nicely. For example, the String array had to be explicitly decided upon (the number of elements) however the list can keep growing indefinitely, making it much more dynamic.