Proposal

In this project, I am going to use tstl to test a small python library. The python library is called LinkedList_Single, which can be found at this link: https://github.com/mirob2005/Python_Data_Structures/blob/master/LinkedLists/LinkedList_Single.py. This library has a basic data structure single linked list function. Most programs contain one or more than one data structures of this library. So basic data structure is essential basement of the robustness of a program. That is the reason way I am going to test these functions.

LinkedList_Single.py has insert, append, returnIndex, updateIndex, deleteIndex, insertBeforeIndex, insertAfterIndex, deleteData, deleteAlldata, insertAfterEveryData, insertBeforeEveryData, deleteList and copyList functions. These functions can be categorized to inserting, deleting, searching and functional method. Each of the categories can be tested in a same way. I will test all of these functions by tstl.

The general idea of testing has generally two steps. The first thing I will do is testing all the functions and make sure all of them work correctly. The second thing is inputting some different types to the functions. They should be thrown out error or compatible with all the data types. Also you can mix some different types into the functions and make sure if they are compatible. For testing, it is a good way to check if the outputs match the varieties of inputs.

It is more complex to test linked list than other data structure, for example queue, and stack. Instead of checking inserting and deleting, the linklist should also be tested with inserting head, inserting tail, and deleting a specific item in the link list. Also, I will test if the list is empty, can I still delete an item. Although link list can be expanded all the items are connected by links. So what if I insert one hundred items, or thousand items? What will happen?

To be more specific, for insert functions, the library provides insert to head, tail and insert to some specific index. So I will test these insert function one by one. A linked list is a data structure consisting of a group of nodes that together represent a sequence and all the components connect with each other with links. Linked list. So after inserting, the components of the list must also connect with others correctly. The delete functions are the same as insert function. After deleting items from a link, the properties of the link must be contained. Deleting actions can be occurred anywhere in the links.

The linked list also provides some functional methods, for example, copyList function. Firstly, I will test this kinds of the correctness of functions. Will they running correct? And for compyList function, I will copy a list to a queue, and copy into a stack. Check whether linked list can be copy a list to another data structure.

After testing all the functions, I will check the robustness of the code. I will try to run the linked list a hundred of times or a thousand of times and see if this code can contain thousands times running.