

C++ Exercise: Linked List Set: Part 1

Michael Bartling

October 16, 2016

1 Introduction

For one of your assignments, you implemented an API for a set type using arrays. Although this implementation has excellent time complexities it unfortunately suffers from the copy problem; it turns out some of the slowest computations are memcopy and memmove. Fortunately, we have another data structure in our software-engineering arsenal, the ordered doubly-linked list!

2 Your Task

In this lab you will create a basic LinkedListSet container (don't worry, we have provided skeleton code for you). Your job is to write 5 key functions to create, interact with, and destroy a LinkedListSet object. Specifically you will implement the following:

```
// Is the Set empty
bool isEmpty(void);

//Default Constructor
LinkedListSet(void);

//Destructor frees up any allocated space
~LinkedListSet(void);

//Scan through the list using cursor n
// looking for element x
// T can be any type that supports <, >, ==
bool is_element(const T& x, Node*& n);

// Insert an element
void insert_element(const T& x);
```

Notice our implementation is templated, this means we can use our LinkedListSet for any type that can be ordered! For example: int, char, floats, strings, ducks, etc.

NOTE: This is an individual assignment, however you are encouraged to form groups to discuss concepts/techniques/algorithms. Just please do not submit identical code.

3 Helpful Hints

- Valgrind magic still works on c++
- use new and delete instead of malloc and free. unlike malloc and free, new and delete don't suck (as much).
- We recommend you use a sentinel (dummy) node whos next always points to the first element in the list. This way you don't have to worry about ordering the insertion of the first element or two :).
- There is an example C++ vector type provided in the C++ lecture folder on canvas
- The way the header file is laid out, we declare the functions inside the class but define them outside the class. The funky syntax of the definitions basically say we are defining the method blah in the LinkedListSet class scope.
- Feel free to add extra helper functions to the LinkedListSet class. However, anything you add that should not be accessible outside the class itself must be private.