**CS 162 Worksheet 10**

**Templates**:

1. Given the following outline of code, use the vector class from the STL to add a new integer, 10, to a vector of integers, v.

#include<iostream>

#include<vector>

using namespace std;

int main(){

}

1. How would you create a templated class called dynarray and provide the same functionality as push\_back() in the vector templated class from the STL?

template<class element>

class dynarray {

 private:

 element \*a;

 int size;

public:

 void add(const element &item);

};

template<class element>

dynarray<element>::add(const element &item){

}

1. Now, write code/pseudocode that has the functionality of adding the element to the front in this class.

**Linked List**

1. What is a singly linked list? Draw a picture of it.
2. What would a doubly linked list and circular linked list look like?

**Compare and contrast: Singly Linked List vs. Array**

1. How are memory assigned for both data structures? Contiguous vs. non-contiguous?
2. In order to store the same amount of elements (e.g., 100 integers), which data structure would consume more memory and why?
3. Advantage(s) of linked list over array:
4. Drawbacks of linked list:

**Complexity Analysis**

1. Suppose an array, arr, and a singly linked list, lst, both have x elements,
* If we want to access the element at index y (y<x), what is the time complexity for both using big O, and why?
* If we want to insert an element in the middle, what is the time complexity for both using big O, and why?
* If we want to delete an element in the middle, what is the time complexity for both using big O, and why?