Jeff Kohler

CIS269

Homework 2

1.

ex 10 p47

Assume the following class declaration:

template<class T>

class genClass {

...

char aFunction(...);

... };

What is wrong with this function definition?

char genClass::aFunction(...) { ... };

It should look like this:

char genClass<T>::aFunction(...) { ... };

2.

ex 13 p47

template<class T, int size = 50>

class genClass {

T storage[size];

..................

void memberFun() {

............

if (someVar < size) { ...... }

............

} };

Transform the declaration which uses an integer variable size as a parameter to template to a declaration of genClass, which does not include size as a parameter to template and yet allows for flexibility of the value of size. Consider a declaration of genClass's constructor. Is there any advantage of one version over another?

template<class T, int size>

class genClass {

T storage[size]

By having the parameter in the template itself, it can be generated dynamically at compile time. This features affords us a great deal of flexibility that to implement in the constructor is burdensome and requires a great deal of error handling. Also as you can tell in my example of this I had to set some arbitrary upper limit to the size, and we could set any size smaller than that, but then had to add error handling for it to perform correctly. Overall the template is a better way to do it, and aside from needing the STL I can't think of a downside of using the template.

3.

Create a class template for polynomials of degree 2 (ax^2 + bx + c). Add addition, and multiplcation capability. Include a driver file, and test cases. Submit the code, and test cases

1x^2 + 5x + 1

3x^2 + -10x + 15

Multiplied: 3x^4 + 5x^3 + -32x^2 + 65x + 15

Added: 4x^2 + -5x + 16

1.5x^2 + 2.5x + 3.5

3.5x^2 + -5.5x + 2.5

Multiplied:” 5.25x^4 + 0.5x^3 + 2.25x^2 + -13x + 8.75

Added: 5x^2 + -3x + 6

If you need a refresher, here is how to multiply two polynomials: http://www.webmath.com/polymult.html

4.

Submit a screenshot of your github directory with at least 2 branches