



[**You may use one side of an 8.5×11 inch sheet of paper.**] Show all of your work clearly in the space provided or on the additional page at the end of the exam. If the additional page is used, clearly identify to which exam question it is related. Be sure to **read each problem carefully**. You should answer all 4 questions. Note that the exam is double sided.

1. (15 points) Write a function that will take in a `std::list` of `std::string` objects and return the length of the shortest object. Be sure to pass the objects appropriately and include all of the code necessary for your function to compile.

2. (15 points) Rewrite the following function so that it does exactly the same thing without using any pointers.

```
double something(double* x, double* y, unsigned int* num)
{
    for(unsigned int i=0; i<*num; ++i) {
        *y *= *x;
    }
    return *y;
}
```

3. (10 points) Precisely and concisely describe what one asteriks (*) represented in a histogram generated by the Histogram class in the second lab assignment.

4. Consider the following class definition for complex numbers:

```
#include <iostream>

class Complex {
public:
    Complex();
    Complex(double rI , double im=0.0);
    Complex(const Complex& rhs);
    ~Complex();
    Complex& operator=(const Complex& rhs);
    void display(std::ostream& os) const;
    bool read(std::istream& is);
    double getReal() const;
    Complex add(const Complex& rhs) const;
    Complex operator-(const Complex& rhs) const;
    Complex operator*(const Complex& rhs) const;
    Complex& operator/=(const Complex& rhs); // Notice rhs is passed by value
private:
    double real;
    double imaginary;
};
Complex operator+(const Complex& lhs , const Complex& rhs);
```

(a) (10 points) Write the add member function. (line 13)



(b) (10 points) Write the + operator (line 21).



(c) (15 points) Write the `/=` operator (line 16).



(d) (10 points) The + operator (line 21) was defined outside of the class while the - operator (line 14) was defined inside the class. Explain why this can be useful.

(e) (15 points) Indicate which functions from the `Complex` class listing are called on each line of the following program. You may indicate the function called by writing the line number for the function called in the `Complex` class listing.

Example:

```

Complex num1 , num2; // 5 5 --- --- --- --- --- ---
#include "Complex.h" // --- --- --- --- --- --- --- ---
int main () // --- --- --- --- --- --- --- ---
{ // --- --- --- --- --- --- --- ---
    Complex a; // --- --- --- --- --- --- --- ---
    // b = 0 + i // --- --- --- --- --- --- --- ---
    Complex b(0 ,1); // --- --- --- --- --- --- --- ---
    { // --- --- --- --- --- --- --- ---
        // c = 2.0 + i0.0 // --- --- --- --- --- --- --- ---
        Complex c=2.0; // --- --- --- --- --- --- --- ---
        c = 3.2 + b; // --- --- --- --- --- --- --- ---
        c /= a; // --- --- --- --- --- --- --- ---
        c = b - 7.25; // --- --- --- --- --- --- --- ---
    } // --- --- --- --- --- --- --- ---
    // --- --- --- --- --- --- --- ---
    return EXIT_SUCCESS; // --- --- --- --- --- --- --- ---
} // --- --- --- --- --- --- --- ---

```

For your convenience, here it is again:

```

#include <iostream>

class Complex {
public:
    Complex();
    Complex(double r1 , double im=0.0);
    Complex(const Complex& rhs);
    ~Complex();
    Complex& operator=(const Complex& rhs);
    void display(std::ostream& os) const;
    bool read(std::istream& is);
    double getReal() const;
    Complex add(const Complex& rhs) const;
    Complex operator-(const Complex& rhs) const;
    Complex operator*(const Complex& rhs) const;
    Complex& operator/=(const Complex& rhs); // Notice rhs is passed by value
private:
    double real;
    double imaginary;
};

Complex operator+(const Complex& lhs , const Complex& rhs);

```



Additional work area for any problem. Clearly identify to which problem the work on this page is related.