CSci 1113
Final

Name: ________________________________

Student ID: __________________________

Instructions: Please pick and answer any 10 of the 12 problems for a total of 100 points. If you answer more than 10 problems, only the first 10 will be graded. The time limit is 120 minutes. Please write your answers in the space provided. The exam is open book and notes. You may use electronic devices to ONLY look at either an e-book version or electronic notes. You may not use the internet, compiler or any other outside resources. (If you are typing on your keyboard/input device for anything other than ctrl-F to find words in the e-book or notes, this is probably not acceptable.)

Problem (1) [10 points] Write (in C++) a “Fraction” class with two integers to store the numerator and denominator. Also write a non-default constructor to initialize the fraction. Write the class following the proper coding conventions.
**Problem (2)** [10 points] Suppose there is a “Course” class. For each of the following pieces of code, describe whether the operator overloading can be done with: (1) a friend function only, (2) a member function only (a function inside the “Course” class), or (3) both types of functions would work (with member functions part of the “Course” class).

Provide a sentence justifying your reasoning.

```cpp
Course c;

1: cout << c;

2: if("csci1113" == c)

3: UMNrequirement libEd;
   libEd += c;
```
Problem (3) [10 points] What is the output of the following code:

class A{
protected:
    int x;
public:
    A();
    A(int in);
    int getValue();
    virtual int virtualValue();
};
A::A() {
    x = 1;
}
A::A(int in) {
    x = in;
}
int A::getValue() {
    return x;
}
int A::virtualValue() {
    return x;
}

class B : public A{
public:
    B(int in);
    int getValue();
    virtual int virtualValue();
};
B::B(int in) {
    x = in;
}
int B::getValue() {
    return 2*x;
}
int B::virtualValue() {
    return 2*x;
}

int main() {
    A a = A(5);  B b = B(5);
    A* aptr = &a;
    A* bptr = &b;
    cout << a.getValue() << " " << b.getValue() << endl;
    cout << aptr->getValue() << " " << bptr->getValue() << endl;
    cout << a.virtualValue() << " " << b.virtualValue() << endl;
    cout << aptr->virtualValue() << " " << bptr->virtualValue() << endl;
}
Problem (4) [10 points] What are both the variables and their values inside “thing” from main()? How many constructors are run to create “thing”? 

class A{
    protected:
        int a;
        int b;
        int c;
    public:
        A();
        A(int x);
    };

A::A() {
    a = 1;
    b = 2;
    c = 3;
}

A::A(int x) {
    a = x;
    b = x+1;
    c = x*2;
}

class B : public A{
    private:
        int d;
    public:
        B(int x);
    };

B::B(int x) {
    d = x;
}

int main() {
    B thing = B(5);
}
Problem (5) [10 points] Suppose you have a class “Linker” as shown below. Write (in C++) code below the lines in main to change this from a chain into a ring by adding the dashed arrow link in the following picture:

class Linker {
    public Linker* p;
};

int main() {
    Linker start;
    // setup pointers, add code below
}
Problem (6) [10 points] Write (in C++) a “<” operator for the “Date” class below to indicate whether a date comes before another date. Write both the function declaration (inside the class) along with the operator’s function definition (how it works).

class Date {
public:
    int month;
    int day;
    int year;
};

int main()
{
    Date cppDayFinal;
    Date cppNightFinal;
    // initialize ^^ (both)
    if(cppNightFinal < cppDayFinal)
    {
        cout << "FIRST" << endl;
    }
}
Problem (7) [10 points] Make a copyArray() function that takes as input two things: (1) a dynamically created array of “Complex” numbers (assume there exists a “Complex” class), and (2) the size of this array. This function should return a copy of the dynamically created array (i.e. changing the returned array should not effect the array passed in).
**Problem (8)** [10 points] Find 3 errors in the code below. Assume that the code is completely shown except for #includes and “using namespace std”. For each error, identify whether it is a runtime error, syntax error or logic error. You must also precisely describe why you think the part of code you identify is an error.

```cpp
class Temperature {
    double* days;
    int size;
    Temperature();
};

Temperature::Temperature()
{
    days = new int[31];
    size = 31;
}

int main() {
    Temperature dec();
    // do something with dec
}
```
Problem (9) [10 points] What is the output of this code:

```cpp
for(int i=0; i < 5; i++) {
    for(int j=0; j < 8; j++) {
        if(j%3 == 0) {
            cout << "X";
        }
        else if(i == 2 && j < 4) {
            cout << "X";
        }
        else if( (i == 0 || i == 4) && j > 4) {
            cout << "X";
        }
        else {
            cout << " ";
        }
    }
    cout << endl;
}
```
Problem (10) [10 points] Write a recursive function that takes as input 3 things: an integer array, the array’s length and a number. Have this recursive function display all the indexes in the array that have the value of this number. You may pass more arguments into the recursive function in addition to the three above if you wish.

Example usage:

```c
int x[] = {1, 2, 3, 1, 2, 1, 1};
recursivePrint(x, 7, 1);
// above couts: 0, 3, 5, 6
```
Problem (11) [10 points] Write (in C++) a function to compute the circumference of a trapezoid. The formula for this is: $b_1 \cdot b_2 \cdot h \cdot \left( \frac{1}{\sin(A)} + \frac{1}{\sin(B)} \right)$. Note: the “cmath” library has a sin() function.
Problem (12) [10 points] Read a sentence from the keyboard, then re-display all parts of that sentence that are in quotes. You can assume there will be an even number of quotes.

Example input:
My "friend" ate "something" that tasted "unknown".

Example output:
frend something unknown