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 two C++ member functions for the class below: (1) a function that returns a pointer to the "size" variable; (2) a function that takes an integer input and returns a pointer to the element in "nums" with the input index.

```
class Problem1 {
public:
    int size;
    double* nums;
    Problem1(int n);
};

Problem1::Problem1(int n) {
    size = n;
    numes = new double[size];
}
```

Problem (2) [10 points] Write (int C++) a destructor for the following class:

```
class Problem2 {
public:
    int* a;
    int b;
    char* c;
    char* d;
    Problem2();
};

Problem2::Problem2() {
    b = 0;
    a = &b;
    c = new char[9001];
    d = (c+3);
}
```

Problem (3) [10 points] Suppose there exists a "Person" class below. Make a child class called "Rockclimber" that inherits from this person class that stores two additional pieces of information: (1) the equipment being used and (2) climbing skill. Then make and use a constructor to create these four people (in main() or something):

```
(1)  name=Max, height=1.778, equipment="shoes, chalk", climbing skill="5.10"
(2)  name=Tim, height=1.803, equipment="harness", climbing skill="5.10"
(3)  name=Allie, height=1.803, equipment="none", climbing skill="5.8"
(4)  name=Liam, height=1.854, equipment="harness, chalk", climbing skill="5.11c"
```

```
class person{
protected:
    string name;
    double height;
};
```

Problem (4) [10 points] What is the output of the following code? You *must* show work to get full credit.

```cpp
class Problem4{
public:
    Problem4 operator+(int x);
    friend int operator+(int x, Problem4 y);
    int operator*(Problem4 x);
};

Problem4 Problem4::operator+(int x) {
    Problem4 result;
    return result;
}

int operator+(int x, Problem4 y) {
    return 2;
}

int Problem4::operator*(Problem4 x) {
    return 1;
}

int main()
{
    Problem4 p1, p2;
    cout << p2*(p1+2) + (9+p2);
}
```

Problem (5) [10 points] Consider the code in main() below. Write code (in C++) to close as many memory leaks as possible. If there is still some memory leaking, describe why you cannot take care of it.

```
int** pp;
int* p1, p2;
int x = 2;

p2=new int;
pp=new int*;
*pp = p2;
p2 = new int;
**pp = *p2;
p1 = *pp;
p2 = new int;
```

Problem (6) [10 points] What is the output of this code:

```cpp
class TA {
public:
    virtual void highFive();
    void greet();
};

void TA::greet() {
    cout << "Let's high five!" << endl;
    highFive();
}

void TA::highFive() {
    cout << "Nice!\n";
}

class jerkface : public TA {
public:
    void greet();
    void highFive();
};

void jerkface::greet() {
    cout << "Give me some skin, bro!" << endl;
    highFive();
}

void jerkface::highFive() {
    cout << "Psych!\n";
}

int main() {
    TA* csci1113[3];
    csci1113[0] = new TA();
    csci1113[1] = new TA();
    csci1113[2] = new jerkface(); // <--- Mina
    for(int i=0; i < 3; i++) {
        csci1113[i] -> greet();
    }
}
```

Problem (7) [10 points] What is the output of this code? You *must* show work for full credit.

```cpp
class Problem7 {
public:
    int i;
    double d;
    Problem7 magic();
};

Problem7 Problem7::magic() {
    Problem7 result;
    result.i=d;
    result.d=i;
    return result;
}

Problem7 magic(Problem7 x) {
    x.i++;
    x.d = x.d/2;
    return x;
}

int main() {
    Problem7 x;
    x.i = 2;
    x.d = 7.5;
    x=magic(x.magic());
    cout << x.i << " " << x.d << endl;
    x=magic(x.magic());
    cout << x.i << " " << x.d << endl;
}
```

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.

```
class Problem8 {
private:
    Problem8* ptr;
public:
    Problem8();
    ~Problem8();
    Problem8(Problem8 x);
};

Problem8::Problem8() {
    ptr = new Problem8[10];
}

Problem8::~Problem8() {
    delete [] ptr;
}

Problem8::Problem8(Problem8 x) {
    ptr = new Problem8[10];
    for(int i=0; i < 10; i++) {
        ptr[i] = x.ptr[i];
    }
}

int main()
{
    Problem8 x;
    x.ptr[0] = x.ptr[9];
}
```

Problem (9) [10 points] What is the output of this code:

```cpp
for(int i=0; i < 5; i++) {
    for(int j=0; j < i; j++) {
        for(int k=0; k < j; k++) {
            cout << "_";
        }
        cout << "x";
    }
    cout << endl;
}
```

Output:

```

x
x_x
x_x__x
x_x__x___x
```

Problem (10) [10 points] Wrtie (in C++) a $recursive$ function that takes as input a string and an integer. This function should return a string containing only characters that have an index a multiple of this integer (always including index 0). You may assume the input string contains at least one character.

```
Example usage:
cout << skipPrint("abcdefghijk", 4) << endl; // shows "aei"
//                 01234567890
cout << skipPrint("hi there!", 2) << endl; // shows "h hr!"
```

Problem (11) [10 points] Write a "shift()" function that takes three integers as input. You should then move the values one variable to the right. In other words, the second input should have the first input's value. The third input should have the second input's value. The first input should have the third input's value.

```
Example usage:
int a=1, b=2, c=3;
shift(a,b,c);
cout << a << b << c << endl; // shows "312"
int x=4, y=8, z=5;
shift(x,y,z);
cout << x << y << z << endl; // shows "548"
```

Problem (12) [10 points] Assume you have a c-string (char array) called "sentence". Output the first letter of every word. You may assume there will only be one space between words.

```
Example sentence:
    char sentence[] = "I saw Susie sitting in a shoe shine shop.";

Example output:
    IsSsiasss
```