COSC 120 **Final Exam** Spring 2012

Name: _____

Write all of your responses on the exam paper. If you need extra space, please use the back.

Part 1: Short Answer (4 Points Each)

1. What are the main differences between a compiler and an interpreter?

2. What are the three main types of programming errors and give a short explanation of how you find these errors.

3. What is the difference between pass by reference, pass by value and pass by pointer? Discuss both the syntactical and memory usage differences for each of these.

4. Write a typedef statement that will allow the user to use the data type `Jack` in place of a float array with 105 elements in it.

5. Write a function header for a void function called `DoSomething` that takes in a one-dimensional array of doubles as a parameter and locks out the changing of this array.

6. Write the declaration of a two-dimensional array of doubles where the array would have 10 rows and 25 columns.

7. Write a function header for a value-returning function, which returns a boolean, called `DoThis` that takes in a one-dimensional array of long integers as a parameter and allows the array to be changed.

8. Given the declaration `int *myarray;` Write a single line of code that will make this into an array of 200 integers. Also write the line of code that will free the memory used by this array.

9. Explain the difference between public, protected and private.

10. What is tail recursion? Give an example from class that was tail recursive and an example that was not tail recursive.

Part 2: Functions (15 Points Each)

1. Write a function that will take an input of a positive integer n and repeat the following process until n is equal to 1. If n is even replace n by n/2 and if n is odd replace n by 3n+1. The function must print out the sequence of numbers along with a count of the number of numbers in the sequence.

2. A generalized Fibonacci sequence is one where we could use more than the last two terms, our first terms can be different from 1 and we can multiply the terms by non-zero constants. For example, the sequence created by $F_n = 2F_{n-1} - F_{n-2} + 3F_{n-3}$ and starting with 2, 1, 4 is 2, 1, 4, 13, 24, 47, 109, … . Write a recursive function that will return the n^{th} generalized Fibonacci number for the above sequence. The input to the function should be the single integer n.

Part 3: Arrays (15 Points Each)

1. Write a function that will bring in array of doubles of unknown size containing positive numbers and a sentinel value of a negative number to designate the end of the array. Have the function return the average of the positive numbers in the array. The header for the function is below.

```
int Average(double arr[])
```

2. Below we have the function header for a Linear Search function. Complete the code so that the function will do a linear search on the array of size, size and determine if the target is in the array. If the target is in the array return the index of the target and if the target is not in the array have the function return –1.

```
int LinearSearch(int arr[], int size, int target)
```

3. Below we have the function header for a Bubble Sort function. Complete the code so that the function will bubble sort the array of size, size.

```
void BubbleSort(int list[], int size)
```

Part 4: Characters & Strings (15 Points Each)

1. The headers of 4 string processing functions are below. `CountLowerCase` is to count the number of lowercase letters, `CountLowerCase` is to count the number of uppercase letters, `CountDigits` counts the numbner of digits and `CountSpaces` counts the number of spaces in the input string. Write the code for each of the functions below. You may not change the headers. For extra credit write a single function that will do all four functions by adding the ability to pass the appropriate function as a parameter.

```
void CountLowerCase(char str[])
{









}

void CountUpperCase(char str[])
{









}
```

```
void CountDigits(char str[])
{









}

void CountSpaces(char str[])
{









}
```

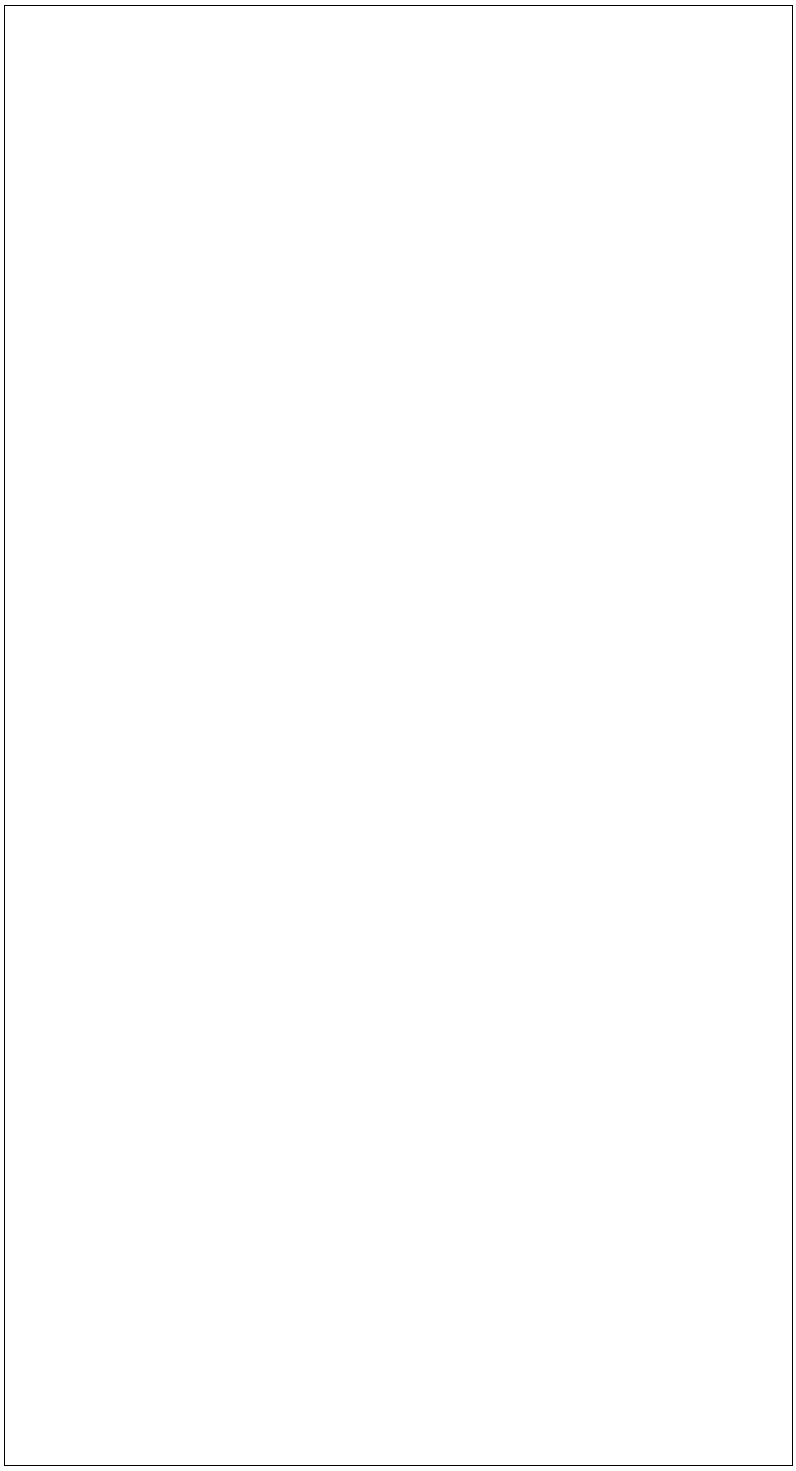
2. Write a program that will take an entire line of input from the user and store it in a string data type (not a character array) called str1. Then have the program construct a new string called str2 and store in it the reverse of the string str1. Finally have it print out both str1 and atr2.

Part 5: Structures & Pointers (15 Points Each)

1. Create a struct named student that stores the student's first and last names, a student ID, an array of 10 exam scores, and the number of exams taken. Also create a function, SearchID that will take in an array of students, the size of the array and a target ID and do a linear search on the array for the student with that ID. If the ID is found the function should return the index of the student with the ID and if the ID is not found the function should return -1.

2. Write the output of the following program. If an output is a memory location state the number being stored in that memory location. Also, mark memory locations that are the same by drawing a line between them. Furthermore, show any places were there are memory leaks, if any.

```
#include <iostream>
using namespace std;

int main()
{
    int i;
    int *j;
    int *k;
    i = 3;
    j = new int;
    *j = 5;
    k = j;

    cout << i << endl;
    cout << &i << endl;
    cout << *j << endl;
    cout << j << endl;
    cout << *k << endl;
    cout << k << endl;
    cout << endl;

    delete j;

    cout << i << endl;
    cout << &i << endl;
    cout << *j << endl;
    cout << j << endl;
    cout << *k << endl;
    cout << k << endl;
    cout << endl;

    k = new int;
    j = new int;

    *k = 17;
    *j = 15;
    k = j;

    cout << i << endl;
    cout << &i << endl;
    cout << *j << endl;
    cout << j << endl;
    cout << *k << endl;
    cout << k << endl;
    cout << endl;
    delete j;
    delete k;
    return 0;
}
```



Part 6: Objects (15 Points Each)

1. Create a class declaration and implementation for a student class. The student class must have data members of the first name, last name, student ID, major and number of credits the student is currently taking. The class should have a default constructor, copy constructor, two functions that return the formal name and informal name of the student, also overload the assignment operator.

2. Write header files and member function implementations for the following. Creates a Car class that includes protected members that identify the car's model (ex. Chevy), year (ex. 2010),and miles driven (ex. 30000). The public members consist of a constructor that has values for each private member data and a default constructor that sets model to "unknown", year to 0 and miles to 0. It has a member function that asks the user for the current year and then determines the average number of miles the car has been driven in a year. Also create a Truck class that inherits the Car class but includes data members for the truck bed length and width (ex. 6ft, 8ft, …) and the towing capacity (ex. 2 tons, 8 tons, …). The public members consist of a constructor that has values for each data member and a default constructor that sets all numeric values to 0. The truck class should also include a member function that prints out the weight per square foot of the bed. This would be calculated by taking the towing capacity and dividing by the area of the bed.