

Computer Science I
Prof. Bolton
Exam 2

Name: _____
Net ID: _____

This exam contains 6 pages (including this cover page) and 6 questions. Total of points is 100.

Conditions: You are permitted the following resources: R= {writing utensils, the text, your notes, a compiler, and the exam itself}. Please: No other items / resources are permitted, no web, no interaction with persons,

Note: The pages are double-sided. **PLEASE WRITE your name** on all pages!

I, _____, understand the above statements and agree to follow these terms, and upon my honor, I swear that the answers provided are of my design and of my effort alone (other than resources R). I have not received nor viewed exam answers from any source but myself.

< *sign* > _____

TRUE / FALSE. WRITE T OR F.

1. (10 points) True or False

Statement	T or F
The const key word is used to declare variables to be immutable (unchangeable)	
Object Oriented Programming is centered around objects that encapsulate data and methods.	
Class member variables are by default public .	
When a function is defined within the class declaration it is referred to as inline .	
A constructor is a member function that is automatically called when a class object is created.	
A destructor is a member function that may be explicitly called when a class object is destroyed.	
Every variable is assigned a memory location which can be retrieved using the * operator.	
The ++ operator cannot be used on pointers.	
The this keyword is a pointer to the object through which a member function is called.	
Each instance of a class has its own copy of a Static variable.	
A friend is a function that is not member of a class, but has access to the private members.	

Identify the Errors. Identify ALL errors (conceptual, syntactic, ...). Explain WHY each is an error.

2. (15 points) The following function is intended to compute the factorial of n assuming n is an integer greater than or equal to 1. Identify (by line) and explain error.

```
line 1: int fact(int &n) {  
line 2: int f = 0;  
line 3: if(n < 0)  
line 4:     sum = 1;  
line 5: else{f = n * fact(n-1);}  
line 6: return f;  
line 7: }
```

3. (15 points) The following class is intended to serve as a Node Class for a linked list which stores an int at each node. Identify (by line) and explain error.

```
line 1: class Node {  
line 2: int data;  
line 3: double next;  
line 4: // methods  
line 5: Node getNext() const{ return next };  
line 6: Node setNext(const Node *n) const{ next = &n; };  
line 7: void setData( double i ){i = next;}  
line 8: int getData(){return data; }  
line 9: }
```

CODING. Write code snippets as instructed. No need to include preprocessor directives. Just provide the code snippet requested.

4. (20 points) Recursion.

A. Consider the following statement: A recursive algorithm is an algorithm that solves a problem, by performing some computational steps and solving a smaller instance of the same problem. Define a recursive function **int sq(int x)** that squares a positive integer using the following recurrence: $x^2 = (x - 1)^2 + 2x - 1$. **Hint.** this equation can be interpreted as a means to compute a sequence of squares. One can compute the next square in the sequence x^2 in terms of the previous square in the sequence $(x - 1)^2$. Another Hint: find the base case as the recurrence is essentially given.

5. (20 points) Assume you are designing software to process Order Data from various suppliers. Each order contains the following information: orderNumber, customerNumber, itemList, and numItems. Also assume you plan to create a list of orders using chaining (a linked list of OrderNodes). Assume that the itemList is an array of variable length. Implement an OrderNode class in C++ that has appropriate member variables, 1 constructor that initializes all member variables, and 1 destructor. (Make any reasonable assumptions as needed; document them.)

6. (20 points) Define a *recursive*, member method `OrderNodes::deleteFromBack(...)` which deletes the last `OrderNode` from a linked list of `OrderNodes`. If there is only one node in the chain, the method will **do nothing**; otherwise, the method will traverse to the end of the chain and remove the last node **appropriately**. (You must determine what parameters are necessary.)