CNS 1400 Exam #1 Study Guide

The objective of this examination is to provide both you and me with some measure of the learning that you have accomplished up to this point in the semester. Learning to program is much more than just being able to regurgitate facts about a particular programming language, and the mechanics of getting a program to compile and execute. This exam will test your understanding of basic programming principles and your ability to apply those principles correctly to solve difficult computing problems. This exam contains multiple choice questions, short answer questions, and a few questions that require you to solve a computing problem and write a short program.

This exam is OPEN BOOK. You may bring your textbook to class to use as a reference for C# syntax. However, you may not use class notes, copies of slides presented in class, or crib notes of any kind. Foreign students may bring a foreign language-English dictionary to the testing center.

You will be responsible for the material discussed in class as well as the textbook reading assignments and all of the labs and programming assignments.

Topic	**Study Material**
Course Introduction • Demonstrate that you understand how projects and labs are submitted. • Be able to discuss the tools provided to help you with this course. ○ Course Web Site ○ Forum ○ Canvas ○ Textbook • Be able to describe ways in which you can communicate with your instructor and other members of the class. • Be able to explain how the course will be graded.	• Syllabus • Style Guide • Course Web Site • Canvas
Introduction to Programming • Be able to describe the basic rules for naming variables and constants, writing a file prologue, and marking lines in your programs as comments. • Demonstrate that you know how and where data of different types is stored in the computer. • Show that you understand the basic operation of a computer as it executes a program. • Be able to describe the basic elements of a C# program. • Be able to explain the steps in creating, compiling, loading a C# program into memory, and executing it. • Demonstrate that you understand good programming style. • Be able to describe the use of the data segment, the stack and the heap to store program data. • Be able to discuss what it means to desk check your code.	• Fundamentals of Programming (Slides) • Chapter 1 in the textbook • Lab #1

• Be able to create an accurate activity diagram.	
Programs and Data • Be able to create and use proper C# identifiers. • Be able to describe the differences between objects and primitive data. • Be able to discuss the different primitive data types in the C# language and know how and when each is used. • Know the difference between a constant and a variable. • Show that you can write declarations for constants and variables properly in C#. • Demonstrate that you can correctly use Console.WriteLine and Console.ReadLine methods. • Show that you can format data for ouput. • Show that you use the string class to store textual data.	• Classes, Programs and Data (slides) • Chapter 2 in the textbook • Labs #2 and #3
Graphical User Interfaces • Be able to describe the basic components used in a Graphical User Interface program, for example, a Form, A MessageBox, a TextBox and a MenuStrip. • Be able to explain the differences between a procedural program and an event driven program.	• Graphical User Interfaces (slides) • Chapter 3 in the textbook • Labs #4 and #5 • Project #1
Expressions • Demonstrate that you know how to correctly use all of the C# operators. • Show that you know how operator precedence affects expression evaluation. • Be sure that you understand how and when integer division occurs. • Show that you understand how and when data type conversions occur in expression evaluation. • Demonstrate that you can use the string class correctly in a C# program. • Be able to write programs that use expressions to solve a problem.	• Expressions (Slides) • Chapter 4 in the textbook • Labs # 6 and #7 • Projects #2