**Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

For each of the below questions, write a short sentence or two to express (in your own words) your answer. Keep the answers short, but use complete, correct, English sentences.

If it helps to clarify the questions, feel free to mentally prefix all the questions with the phrase "According to the video…"

1. After you’ve watched all the videos, please answer this question:  
   Of all the videos that you watched, if you could pick one video to be re-recorded by the instructor outside of class which would you choose? Why?  
   (Keep in mind the recording outside of class will omit any pauses from the instructor answering student questions, have less hemming and hawing, etc, and generally be more concise)

|  |
| --- |
| < Write your answer here > |

**VIDEO: How To Use My Videos**

1. When viewing the videos in your web browser, where are the video-playback controls located?

|  |
| --- |
| < Write your answer here > |

1. List out at least three controls that you’ll find on the web page, and what each one does.

|  |
| --- |
| < Etc. > |

1. How can you download the .MP4 video file (so that you can watch it in a media player program on your local computer)?

|  |
| --- |
|  |

1. List out at least three features that the VLC Media Player has, and what each one does.

|  |
| --- |
|  |

**VIDEO: What is a project?**

1. How many files might a typical program be made of?

|  |
| --- |
|  |

1. What can you think of a project as?  
   What is the primary purpose of a project?

|  |
| --- |
|  |

1. Give some examples of different types of files that you might store inside a project?

|  |
| --- |
|  |

1. What is a Visual Studio Solution?

|  |
| --- |
|  |

1. Give some examples of different things that a Solution might contain.

|  |
| --- |
|  |

1. THIS IS REALLY IMPORTANT:  
   When you’re working with a Project/Solution in Visual Studio, WHICH FILE SHOULD YOU OPEN?

|  |
| --- |
|  |

1. What problem will you run into if you open a C# file directly?

|  |
| --- |
|  |

**VIDEO: How to create a simple console application**

1. Briefly explain two separate ways to start the process of creating a new project.

|  |
| --- |
|  |

1. If you’re working at school and you have trouble getting your program to compile and run on the H: (network) drive, where should you try saving the project?

|  |
| --- |
|  |

1. How do you tell Visual Studio to display line numbers?

|  |
| --- |
|  |

1. What is the difference between “Start **With** Debugging” and “Start With**out** Debugging”

|  |
| --- |
|  |

1. How does Visual Studio indicate that your file has a compile-time/syntax error?

|  |
| --- |
|  |

**VIDEO: How to download and use a simple console application**

1. What is the key thing to do after you’ve downloaded the .ZIP archive?

|  |
| --- |
|  |

1. If you don’t extract the files from the .ZIP archive and instead you open the C# file from within Visual Studio anyways (while the file is still inside the .ZIP archive) what problem will you run into?

|  |
| --- |
|  |

1. How do you get Visual Studio to display the Solution Explorer?

|  |
| --- |
|  |

**VIDEO: How Basic console I/O ("Everything you need to know for this class, and nothing more”)**

1. What does Console.WriteLine do?

|  |
| --- |
|  |

1. How does Console.Write differ from Console.WriteLine?

|  |
| --- |
|  |

1. How does one produce a line of output in Java (and potentially in C#). Assuming that int x = 3; int y = 7; has been declared, list the code here:

|  |
| --- |
|  |

1. What is the better way to print out variables in C#. Assuming that int x = 3; int y = 7; has been declared, list the code here:

|  |
| --- |
|  |

1. When printing out variables, what does {0} refer to? {1}?

|  |
| --- |
|  |

1. Before getting input from the user what should the program first do?

|  |
| --- |
|  |

1. What is the line of C# code that will get whatever the user has typed?  
   (Make sure that your code stores that input into a variable)

|  |
| --- |
|  |

1. What is the line of C# code that will convert the input from text into a integer?

|  |
| --- |
|  |

1. If the user types in a non-integer value, what will the value of **out x** be?

|  |
| --- |
|  |

1. What is the C# source code can you use to check if the user actually typed in an integer (and display a message either repeating that value, or telling the user that they didn’t type a number in)?

|  |
| --- |
|  |

1. What is the C# source code that will attempt to convert user input into a real number (into a double value)?

|  |
| --- |
|  |

**VIDEO: Expression Evaluation (Order of operations)**

1. Describe in your own words what the first thing that we do when evaluating an expression:

|  |
| --- |
|  |

1. We then repeatedly do two steps.  
   What is the step 1? What is step 2?

|  |
| --- |
|  |

1. When you see a number like 3.0, what is it’s data type?

|  |
| --- |
|  |

1. When you see a number like 3 (without the .0 / without anything after the decimal point), what is it’s data type?

|  |
| --- |
|  |

1. How do we figure out which operator goes next?

|  |
| --- |
|  |

1. Once we’ve identified which operator will be evaluated next,   
   what are the three steps in actually doing / evaluating an operator?

|  |
| --- |
|  |

1. In the precedence table that was built for you, which operator goes first?

|  |
| --- |
|  |

1. What does ‘left to right associativity’ mean?

|  |
| --- |
|  |

1. How is the assignment operator unusual?

|  |
| --- |
|  |

1. In the expression that gets evaluated in the video, what “operator” goes first?

|  |
| --- |
|  |

1. Within that thing that gets evaluated first, what is the first operator that we evaluate?

|  |
| --- |
|  |

1. Why is it always safe to convert an integer into a double?

|  |
| --- |
|  |

1. Can you put an assignment operator inside a larger expression?

|  |
| --- |
|  |

1. In the video you saw many steps needed to evaluate the expression.   
   Does the computer actually do all these steps, or are these just for teaching purposes?

|  |
| --- |
|  |