**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: Inheritance: Conceptual overview**

1. What is the purpose behind inheritance? (Give a very brief, intuitive explanation)

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

1. It is bad to copy-and-paste the Car code and then tweak it in order to create a more specialized subclass/subcategory (such as, for example, a SpyCar). Briefly, intuitively explain one reason why this is bad / will introduce problems later on.

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

1. In order to tell C# that a given class extends another, existing class, what do add to the class declaration?

|  |
| --- |
|  |

**VIDEO: Inheritance: vocabulary**

1. A **SpyCar** has a copy of what?   
   (This will be covered in more detail in the next video)

|  |
| --- |
|  |

1. What are the two pairs of words we use to talk about the Car & SpyCar classes? Make sure that you’re clear which word describes Car and which word describes SpyCar, and which two words typically go together in which pair:

|  |
| --- |
|  |

1. If you have a class, then create a subclass of it, and you then create a further subclass of the firsdt subclass?   
   What advantage(s) would this give you?

|  |
| --- |
|  |

**VIDEO: Inheritance: Inheriting data**

1. What data fields does an instance of the SpyCar class have? For each field, explain where it came from / why the SpyCar has it.

|  |
| --- |
|  |

1. Can a derived class directly access a data field that was declared to be private in a base class?  
   If not, then how does the derived class make use of it (explain this briefly and intuitively)?

|  |
| --- |
|  |

1. What does the C# keyword **protected** do? How is this different from **private**?

|  |
| --- |
|  |

1. Can an instance of the **Car** class ever access the fields of the derived **SpyCar** class? Explain why this does (or does not) make sense, briefly and intuitively.

|  |
| --- |
|  |

**VIDEO: Inheritance: Constructors**

1. In C#, how do you call a base-class constructor? If there are multiple (possibly overloaded) constructors in the base class, how do you indicate which one you want to use?

|  |
| --- |
|  |

1. How can you set things up so that all instances of the **SpyCar** class always has, say, zero backseats.

|  |
| --- |
|  |

1. If you don’t specify a base-class constructor (to use in a derived class’s constructor), which one will C# use?

|  |
| --- |
|  |

1. How can you provide a default constructor in the base class without duplicating code amongst the constructors?

|  |
| --- |
|  |

1. Building off your answer for the previous question, what is a very typical pattern when a class provides multiple constructors?

|  |
| --- |
|  |

1. Why do you have to copy-and-paste the parameters for the derived class’s constructor(s)?

|  |
| --- |
|  |

**VIDEO: Polymorphism: Overriding Methods**

1. When you declare a method to be **virtual**, what does that mean?

|  |
| --- |
|  |

1. In order to override a method in a subclass, what do you need to copy from the superclass?

|  |
| --- |
|  |

1. In addition to what was listed in the prior question, what other keyword do you need to add to the subclass’s method?

|  |
| --- |
|  |

1. When **playerCar** is set to refer to the **c1** object, and you call **Print**, what happens? Why?

|  |
| --- |
|  |

1. When **playerCar** is set to refer to the **sp** object, and you call **Print**, what happens? Why?

|  |
| --- |
|  |

1. What is the name for this technique?

|  |
| --- |
|  |

1. In C#, if you wanted the **SpyCar** to invoke (to call) the base-class version of **Print** (i.e., **Car.Print**), what snippet of C# code would you use?

|  |
| --- |
|  |

1. What is the name of the technology that allows (parts of) your C# program to run in parallel?

|  |
| --- |
|  |

**VIDEO: Polymorphism: An Array Of Objects With An Overridden Method**

1. What output will be produced by the loop that iterates through the **garage** array and calls the Print method on all the non-null slots? Please summarize briefly in English what will be printed, and then make sure to explain WHY this output is produced.

|  |
| --- |
|  |

1. When adding objects to an array, are you required to create a new object for the array? If not, how would you add, say, the **c2** car object into the array

|  |
| --- |
|  |

**VIDEO: Named Constants**

1. What is a “magic number”?

|  |
| --- |
|  |

1. In terms of printing out a magic number, what is a better alternative to simply printing the number directly?

|  |
| --- |
|  |

1. What modifier can be used to mark an integer (or double, or short, or char, etc) as a named constant?

|  |
| --- |
|  |

1. What is the “readonly” modifier used for?

|  |
| --- |
|  |

1. What happens if I try to change the value of a named constant (after it’s been created)?

|  |
| --- |
|  |

1. How do people (traditionally) capitalize named constants?

|  |
| --- |
|  |

1. Where do you normally put named constant declarations?

|  |
| --- |
|  |

1. Copy over the C# source code that declares a new class named ComputerScreen, including it’s two named constants:

|  |
| --- |
|  |

1. Copy over a snippet of C# source code that will make use of the two constants in a class that’s separate from the ComputerScreen class?

|  |
| --- |
|  |