**Code Analysis Question 1**

Question: Outline any issues/concerns with the implemented code and explain why the block below does not emit “bow-wow”:

Animal d = new Dog();

Console.Write(d.speak(0));

**Issues/concerns with implemented code**

1. The speak methods for Cat and Dog classes will issue a compiler warning
2. Both the Cat and Dog classes with the speak method do not override the base class’(Animal) speak method. It will only hide it since new is implied. Therefore since the speak method is hidden instead of overridden, when the speak method is accessed via the Animal class reference, Animal.speak() will be called instead of Cat.speak() or Dog.speak().
3. There is no polymorphism for the Cat and Dog classes here so the Animal.speak() method will not be overridden if the Cat class is accessed through the Animal class reference as shown above with the Dog class
4. There is an input parameter mismatch for the speak method. The speak method in the base class, Animal, takes in int x as an input while the speak method in the Dog class takes in short x as an input.

**Why the block does not emit “bow-wow”**

1. The code block above will emit “silence” and not “bow-wow” because the Animal class speak method is used as a reference instead of Dog
2. The input parameters between the speak methods for Animal and Dog are mismatches data types, as Animal takes in a int x while Dog takes in short x.
3. The speak method in Dog does not override the speak method in Animal. Since there is no override keyword, the Animal class will be only hidden and because the Animal class is used as a reference instead of the Dog class, d.speak(0) will output “silence” instead of “bow-wow” since the Animal.speak() method is used as the reference class