**44-542 Object Oriented Programming Exam 02 Part 01 (60 Points)**

**Multiple Choice (2 pts each):** Select the BEST correct answer. If you select more than one option, the entire question will be counted as incorrect.

1. Which of the following is true of default methods?
   1. default methods only appear in interfaces; they are not allowed in abstract or concrete classes
   2. a default method includes the implementation of the method.
   3. you can have at most one default method in an interface
   4. all of the above are true
   5. only a) and b) are true
2. Classes \_\_\_\_\_ classes; interfaces \_\_\_\_\_ interfaces; classes \_\_\_\_\_ interfaces
   1. implement, implement extend
   2. implement, extend, implement
   3. extend, extend, implement
   4. extend, implement, extend
   5. extend, extend, extend
3. A reference variable’s type cannot be that of an interface.
   1. true
   2. false
4. The \_\_\_\_\_ interface can be used to extend the usage of the enhanced for loop.
   1. **Iterator**
   2. **Enhanced**
   3. **Comparable**
   4. **Iterable**
5. **InputMismatchException** is an example of a(n)
   1. **IOException**
   2. **RuntimeException**
6. Which of the following is true of checked and unchecked exceptions?
   1. all subclasses of **IOException** are checked exceptions
   2. all subclasses of **RuntimeException** are unchecked exceptions
   3. both of the above are true
   4. none of the above are true
7. **FileNotFoundException** is an example of a(n)
   1. **IOException**
   2. **RuntimeException**
8. In a **try-catch**, **catch** has \_\_\_\_\_ parameter(s).
   1. 0
   2. 1
   3. 2
   4. 3
9. To throw an exception in a method, use the keyword \_\_\_\_.
   1. exception
   2. throw
   3. throws
10. The \_\_\_\_\_ class provides static methods that can be used to operate on collections, such as array lists, for example.
    1. **Collection**
    2. **Collections**
    3. **Comparable**
11. In white-box testing, the testing is from the point of view of the user.
    1. true
    2. false
12. Recall that the header for method **main** is

**public static void main(String[] args)**

Assume that we have a valid Java program named **MyProgram**. This program has no errors in it and is guaranteed to run correctly. We have compiled the program, and we now want to run it at the command line. We navigate to the directory containing the **.class** file, and we enter

**java MyProgram "Hello Leonard"**

Which of the following describes the result of this statement?

* 1. this is an invalid statement; an error will be produced
  2. the argument **Hello Leonard** is stored in **args[0]**
  3. the argument **Hello Leonard** is stored in **args[1]**
  4. the argument **Hello** is stored in **args[0]** and **Leonard** is stored in **args[1]**
  5. the argument **Hello** is stored in **args[1]** and **Leonard** is stored in **args[2]**

1. In a UML diagram, which of the following is the correct syntax for attributes?
   1. visibility attributeName: attributeType
   2. visibility attributeType: attributeName
   3. attributeName: attributeType visibility
2. Suppose we have a **Student** class with a subclass named **GraduateStudent**. Consider the following code segment:
3. **Student stu = new Student(…);**
4. **GraduateStudent gradStu = new GraduateStudent(…);**
5. **stu = gradStu;**

Statement 3 above is legal and is an example of

* 1. inheritance
  2. late-binding polymorphism
  3. polymorphic substitution

1. Which of the following is true of abstract methods.
   1. an abstract method is declared with the keyword **abstract**
   2. an abstract method has no method body – that is, it is not implemented in the superclass
   3. an abstract method must be overridden by all subclasses, except for subclasses also declared as abstract
   4. all of the above are true
   5. Only a) and b) are true

**The remaining questions refer to the Herbivore-Mammal-Carnivore class diagram on a separate handout. If you did not receive this handout, ask your instructor for a copy.**

**Note that AbstractMammal is an abstract class; it is the only abstract class in the class diagram.**

1. Is the following statement legal or illegal?

**HoovedMammal hm = new Herbivore();**

* 1. legal
  2. illegal

1. Is the following statement legal or illegal?

**Carnivore c = new Carnivore();**

* 1. legal
  2. illegal

1. Is the following statement legal or illegal?

**Carnivore c = new Lion();**

* 1. legal
  2. illegal

1. Consider the following code segment, and then determine which of the choices is correct.

**Horse horse01 = new Horse();**

**Horse horse02 = new QuarterHorse();**

**QuarterHorse horse03 = horse02;**

* 1. this code segment will not compile
  2. this code segment will compile, but it will not run correctly
  3. this code segment will compile and will also run correctly, without errors

1. Is the following statement legal or illegal?

**Horse h = new QuarterHorse();**

* 1. legal
  2. illegal

1. Is the following statement legal or illegal?

**QuarterHorse qh = new QuarterHorse();**

* 1. legal
  2. illegal

1. Is the following statement legal or illegal?

**QuarterHorse h = new Horse();**

* 1. legal
  2. illegal

1. Is the following statement legal or illegal?

**AbstractMammal am = new AbstractMammal();**

* 1. legal
  2. illegal

1. Consider the following code segment, and then determine which of the choices is correct.

**Mammal mammal01 = new Lion();**

**Lion mammal02 = new Lion();**

**mammal02 = (Lion) mammal01;**

* 1. this code segment will not compile
  2. this code segment will compile, but it will not run correctly
  3. this code segment will compile and will also run correctly, without errors

1. In the class diagram, **Lion** extends **Carnivore**.
   1. true
   2. false
2. In the class diagram, **Dog** implements **Carnivore**, **Mammal**, **Herbivore**.
   1. true
   2. false
3. Consider the following code segment, and then determine which of the choices is correct.

**HoovedMammal hooved01 = new HoovedMammal();**

**Horse hooved02 = new Horse();**

**hooved02 = (Horse) hooved01;**

* 1. this code segment will not compile
  2. this code segment will compile, but it will not run correctly
  3. this code segment will compile and will also run correctly, without errors

1. In the class diagram, **HoovedMammal** extends **AbstractMammal**.
   1. true
   2. false
2. According to the class diagram, a **PitcherPlan** is-a **Carnivore**.
   1. true
   2. false
3. According to the class diagram, a **HoovedMammal** is-a **Herbivore**.
   1. true
   2. false