AP Computer Science A – Unit 05 Writing Classes Name: ____________________

Writing Classes MCQ, Part B

This quiz has 12 questions.

1. Consider the following class. The method `getTotalSalaryAndBonus` is intended to return an employee’s total salary with the bonus added. The bonus should be doubled when the employee has `10` or more years of service.

```
public class Employee {

   private String name;
   private double salary;
   private int yearsOfService;

   public Employee(String n,
                   double sal,
                   int years)
   {
       name = n;
       salary = sal;
       yearsOfService = years;
   }

   public double
         getTotalSalaryAndBonus(
                   double bonus)
   {
      /* missing code */
   }
}
```

Which of the following could replace *`/* missing code */`* so that method `getTotalSalaryAndBonus` will work as intended?

Ⓐ
```
if( years >= 10 ) {
   bonus *= 2;
}
return salary + bonus;
```

Ⓑ
```
if( yearsOfService >= 10 ) {
   bonus *= 2;
}
return salary + bonus;
```

Ⓒ
```
return salary + bonus;
```

Ⓓ
```
if( years >= 10 ) {
   bonus *= 2;
}
return sal + bonus;
```

Ⓔ
```
if( yearsOfService >= 10 ) {
   bonus *= 2;
}
return sal + bonus;
```

2. Consider the following class declaration. The `changeWeather` method is intended to update the value of the instance variable `weather` and return the previous value of `weather` before it was updated

```
public class WeatherInfo {

   private String city;
   private int day;
   private String weather

   public WeatherInfo(String c,
                      int d,
                      String w)
   {
      city = c;
      day = d;
      weather = w;
   }

   public String changeWeather(
                      String w)
   {
      /* missing code */
   }

}
```

Which of the following options should replace *`/* missing code */`* so that the `changeWeather` method will work as intended?

Ⓐ
```
String prev = w;
return weather;
```

Ⓑ
```
String prev = weather;
return w;
```

Ⓒ
```
String prev = w;
return prev;
```

Ⓓ
```
weather = w;
String prev = weather;
return prev;
```

Ⓔ
```
String prev = weather;
weather = w;
return prev;
```

Writing Classes MCQ, Part B

3. Consider the following class definition.

```
public class BoolTest {
   private int one;
   public BoolTest(int newOne) {
      one = newOne;
   }
   public int getOne() {
      return one;
   }
   public boolean isGreater(
                BoolTest other)
   {
      /* missing code */
   }
}
```

The `isGreater` method is intended to return `true` if the value of `one` for this `BoolTest` object is greater than the value of `one` for the `BoolTest` parameter `other`, and `false` otherwise. The following code segments have been proposed to replace */* missing code */*.

I. `return one > other.one;`

II. `return one > other.getOne();`

III. `return getOne() > other.one;`

Which of the following replacements for */* missing code */* can be used so that `isGreater` will work as intended?

Ⓐ I only

Ⓑ III only

Ⓒ I and II only

Ⓓ I and III only

Ⓔ I, II, and III

4. Consider the following class definition.

```
public class Gadget {
   private static int status = 0;
   public Gadget() {
      status = 10;
   }
   public static
         void setStatus(int s)
   {
      status = s;
   }
}
```

The following code segment appears in a method in a class other than `Gadget`.

```
Gadget a = new Gadget();
Gadget.setStatus(3);
Gadget b = new Gadget();
```

Which of the following best describes the behavior of the code segment?

Ⓐ The code segment does not compile because the `setStatus` method should be called on an object of the class `Gadget`, not on the class itself.

Ⓑ The code segment does not compile because the `static` variable `status` is not properly initialized.

Ⓒ The code segment creates two `Gadget` objects `a` and `b`. The class `Gadget`'s `static` variable `status` is set to `10`, then to `3`, and then back to `10`.

Ⓓ The code segment creates two `Gadget` objects `a` and `b`. After executing the code segment, the object `a` has a `status` value of `3` and the object `b` has a `status` value of `3`.

Ⓔ The code segment creates two `Gadget` objects `a` and `b`. After executing the code segment, the object `a` has a `status` value of `3` and the object `b` has a status value of `10`.

Writing Classes MCQ, Part B

5. Consider the following class definition.

```
public class Beverage {
   private int numOunces;
   private static int numSold = 0;
   public Beverage(int numOz) {
      numOunces = numOz;
   }
   public static void sell(int n) {
     /* implementation not shown */
   }
}
```

Which of the following best describes the sell method’s level of access to the `numOunces` and `numSold` variables?

Ⓐ Both `numOunces` and `numSold` can be accessed and updated.

Ⓑ Both `numOunces` and `numSold` can be accessed, but only `numOunces` can be updated.

Ⓒ Both `numOunces` and `numSold` can be accessed, but only `numSold` can be updated.

Ⓓ `numSold` can be accessed but not updated, `numOunces` cannot be accessed or updated

Ⓔ `numSold` can be accessed and updated; `numOunces` cannot be accessed or updated.

6. The following class is used to represent shipping containers. Each container can hold a number of units equal to `uPerC`.

```
public class UnitsHandler {
   private static int totalUnits = 0;
   private static int containers = 0;
   private static int uPerC = 0;
   public UnitsHandler(int containerSize)
   {
      uPerC = containerSize;
   }
   public static void update(int c) {
      containers = c;
      totalUnits = uPerC * containers;
   }
}
```

The following code segment appears in a method in a class other than `UnitsHandler`. Assume that no other code segments have created or modified `UnitsHandler` objects.

```
UnitsHandler large =
   new UnitsHandler(100);
UnitsHandler.update(8);
```

Which of the following best describes the behavior of the code segment?

Ⓐ The code segment does not compile, because it is not possible to create the object `large` from outside the `UnitsHandler` class.

Ⓑ The code segment does not compile, because it attempts to change the values of `private` variables from outside the `UnitsHandler` class.

Ⓒ The code segment does not compile because the update method should be called on the object `large` instead of on the `UnitsHandler` class.

Ⓓ The code segment creates a `UnitsHandler` object called `large` and sets the static variable `uPerC` to `100`. The static variables `containers` and `totalUnits` each retain the default value `0`.

Ⓔ The code segment creates a `UnitsHandler` object called `large` and sets the static variables `uPerC`, `containers`, and `totalUnits` to `100`, `8`, and `800`, respectively.

Name: ____________________

Writing Classes MCQ, Part B

7. Consider the following class, which models a bank account. The `deposit` method is intended to update the account balance by a given amount; however, it does not work as intended.

```
public class BankAccount {

   private String ownerName;
   private double bal;
   private int accountNumber;

   public BankAccount(
          String name,
          double initialBalance,
          int acctNum)
   {
      ownerName = name;
      bal = initialBalance;
      accountNumber = acctNum;
   }

   public void deposit(double a)
   {
      double bal = bal + a;
   }

}
```

What is the best explanation of why the `deposit` method does not work as intended?

Ⓐ The `deposit` method must have a `return` statement

Ⓑ In the `deposit` method, the variable `bal` should be replace by the variable `initialBalance`.

Ⓒ In the `deposit` method, the variable `bal` is declared as a local variable and is different from the instance variable `bal`.

Ⓓ The method header for the `deposit` method should be `public void deposit(a).`

Ⓔ The variable `bal` must be passed to the `deposit` method.

8. Consider the following class declaration.

```
public class Student {

   private String fName;
   private String lName;
   private int age;

   public Student(String fName,
                  String lName,
                  int age)
   {
      fName = fName;
      lName = lName;
      age = age;
   }

   public String toString() {
      return = fName+" "+lName;
   }

}
```

The following code segment appears in a method in a class other than `Student`. It is intended to create a `Student` object and then to print the first name and last name associated with that object.

```
Student s =
   new Student("Priya","Banerjee",-1);
System.out.println(s);
```

Which of the following best explains why the code segment does not work as expected?

Ⓐ The code segment will not compile because an object cannot be passed as a parameter in a call to `println`.

Ⓑ The code segment will not compile because `fName`, `lName`, and `age` are names of instance variables and cannot be used as parameter names in the constructor.

Ⓒ The code segment will not compile because the constructor needs to ensure that `age` is not negative.

Ⓓ The code segment will compile, but the instance variables will not be initialized correctly because the variable names `fName`, `lName`, and `age` refer to the instance variables inside the constructor.

Ⓔ The code segment will compile, but the instance variables will not be initialized correctly because the variables names `fName`, `lName` and `age` refer to the local variables inside the constructor.

Writing Classes MCQ, Part B

9. Consider the following class definition.

```
public class ClassP {
   private String str;

   public classP(String newStr)
   {
      String str = newStr;
   }
```

The `ClassP` constructor is intended to initialize the `str` instance variable to the value of the formal parameter `newStr`. Which of the following statements best describes why the `ClassP` constructor does not work as intended?

Ⓐ The constructor should have a return type of `String`.

Ⓑ The constructor should have a return type of `void`.

Ⓒ The instance variable `str` should be designated `public`.

Ⓓ The variable `str` should be designated `public` in the constructor.

Ⓔ The variable `str` should not be declared as a `String` in the constructor.

10. Consider the following class definition.

```
public class Contact {
   private String contactName;
   private String contactNumber;

   public Contact(String name,
                  String number)
   {
      contactName = name;
      contactNumber = number;
   }

   public void doSomething() {
      System.out.println(this);
   }

   public String toString() {
      return contactName + " " +
             contactNumber;
   }
}
```

The following code segment appears in another class.

```
Contact c = new Contact("Alice",
                        "555-1234");
c.doSomething();
c = new Contact("Daryl", "");
c.doSomething();
```

What is printed as a result of executing the code segment?

Ⓐ `Daryl`

Ⓑ `Daryl 555-1234`

Ⓒ
```
Alice 555-1234
Daryl
```

Ⓓ
```
Alice 555-1234
Daryl 555-1234
```

Ⓔ
```
this
this
```

Writing Classes MCQ, Part B

11. Consider the following class definition.

```
public class Person {
   private String name;
   private int feet;
   private int inches;
   public Person(String nm,
                 int ft, int in)
   {
      name = nm;
      feet = ft;
      inches = in;
   }
   public int heightInInches() {
      return feet * 12 + inches;
   }
   public String getName() {
      return name;
   }
   public String
   compareHeights(Person other)
   {
      if(this.heightInInches() <
         other.heightInInches())
      {
         return name;
      }
      else if(
         this.heightInInches() >
         other.heightInInches())
      {
         return other.getName();
      }
      else return "Same";
      }
}
```

The following code segment appears a method in a class other than `Person`.

```
Person andy = new Person("Andrew",5,6);
Person ben = new Person("Benjamin",6,5);
System.out.println(
   andy.compareHeights(ben));
```

What, if anything, is printed as a result of executing the code segment?

Ⓐ `Andrew`

Ⓑ `Benjamin`

Ⓒ `Same`

Ⓓ Nothing is printed because the method `heightInInches` cannot be called on `this`.

Ⓔ Nothing is printed because the method `compareHeights` in the `Person` class cannot take a `Person` object as a parameter.

12. Consider the following class definition.

```
public class Email {
   private String username;
   public Email(String u) {
      username = u;
   }
   public void printThis() {
      System.out.println(this);
   }
   public String toString() {
      return username +
             "@example.com";
   }
}
```

The following code segment appears in a method in another class.

```
Email e = new Email("default");
e.printThis();
```

What, if anything, is printed as a result of executing the code segment?

Ⓐ `e`

Ⓑ `default`

Ⓒ `e@example.com`

Ⓓ `default@example.com`

Ⓔ Nothing is printed because the class will not compile