CSE110 Review Questions (Solutions)

Prepared by Ryan Dougherty

Introduction to Classes

Question 1 Which of the following enforces Encapsulation?
a) Make instance variables private
b) Make methods public
c) Make the class final
d) Both a and b
e) All of the above
Answer: D

Question 2 Use the following class to answer the questions below:

```
public class Store {
        private int quantity;
        private double price;

        public Store(int q, double p) {
                quantity = q;
                price = p;
        }

        public int getQuantity() {
                return quantity;
        }

        public void setPrice(double p) {
                price = p;
        }

        public double calcTotal() {
                return price * quantity;
        }
}
```

a) What is the name of the class? Store
b) List all instance variables of the class. quantity, price
c) List all methods of the class. Store(int, double), getQuantity(), setPrice(double), calcTotal()
d) List all mutators in the class. setPrice(double)
e) List all accessors in the class. getQuantity()
f) List which method is the constructor. Store(int, double)
g) Write a mutator for the quantity.
Answer:

```
public void setQuantity(int q) {
        quantity = q;
}
```

h) Write an accessor for the price.
Answer:

```
public double getPrice() {
        return price;
}
```

i) Write a line of code that will create an instance called videoStore that has quantity 100 and a price of 5.99.
Answer:

```
Store videoStore = new Store(100, 5.99);
```

j) Call the calcTotal method with the videoStore object (from part i) to print out the total.
Answer:

```
System.out.println("Total: " + videoStore.calcTotal());
```

Question 3 True or False? If no constructor is provided, then Java automatically provides a default constructor. Answer: True. Java will automatically provide a default constructor if none is given. However, if any other constructor is given, the default constructor can no longer be used.

Question 4 True or False? A method must have at least 1 return statement. Answer: False. Any method with return type void is not required to have a return statement. However, if it does have a return statement, it must not have a value associated with it (i.e. "return 5;" is not allowed for void methods, but "return;" is.).

Question 5 Correct the following class definition if you think it will not work:

```
public class Student {
        private String name, major;

        public Student() {
                name = "???";
                major = "xxx";
        }

        public Student(String n, String m) {
                n = name;
                m = major;
        }

        public String getMajor() {
                return m;
        }

        public String getName() {
                return n;
        }
}
```

Answer: There are problems in the assignment in the constructor, "n = name" and "m = major" should be the other way around. Also, the "return m" in getMajor and "return n" in getName need to be "return major" and "return name", respectively.

Question 6 Implement a class called AsuStudent. The class should keep track of the student's name, number of classes registered, hours spent per week for a class (consider a student devotes the same amount of time for each of his/her classes per week). Implement a toString method to show the name and number of classes registered by a student, a getName method to return the name of the student, a getTotalHours method to return the total number of hours per week, and a setHours method to set the number of hours the student devotes for each class.

Answer:

```
public class AsuStudent {
        private String sName;
        private int classNum, hrPerWeek;

        public AsuStudent(String name, int class, int hr) {
                sName = name;
                classNum = class;
                hrPerWeek = hr;
        }

        public String toString() {
                return sName + " " + classNum + " " + hrPerWeek;
        }

        public String getName() {
                return sName;
        }

        public int getTotalHours() {
                return classNum * hrPerWeek;
        }

        public void setHours(int time) {
                hrPerWeek = time;
        }
}
```