CSCI 111  
Midterm  
Spring 2014

<table>
<thead>
<tr>
<th>Question Number</th>
<th>Point Value</th>
<th>Points Awarded</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

Name ___________________________
Student Id ______________________
Lab Section_______________________

- 50 Minutes
- If I can’t read your handwriting I have to count it wrong
- Keep short answers short
- You do not need to put comments in any code you write.
- NO NOTES, PARTNERS, ELECTRONIC DEVICES…
Question one - (40 points) Implement a Book class for a book store as described: A Book has a title, cost, and number in stock. Set the title and cost to values passed to the constructor. Create a “get method” for each instance variable. Create a method that will increase the number in stock by an amount provided. Finally, create a method that will return the total value of the given book in stock (do not worry about formatting this output to a normal currency value). You do not need to provide any comments in your code.

Here is sample code and output from a Driver for your Book class:

```java
public class Book
{
    private int numInStock;
    private String title;
    private double cost;

    public Book(String inTitle, double inCost)
    {
        title = inTitle;
        cost = inCost;
    }

    public String getTitle()
    {
        return title;
    }

    public int getNumInStock()
    {
        return numInStock;
    }

    public double getCost()
    {
        return cost;
    }

    public void increaseStock(int newAmount)
    {
        numInStock += newAmount;
    }

    public double calcStockValue()
    {
        return getNumInStock() * getCost();
    }
}
```

Book book1 = new Book("Extreme Alpinism", 19.95);
book1.increaseStock(5);
book1.increaseStock(2);
System.out.println(book1.getNumInStock() + " copies of " + book1.getName() + " in stock.");
System.out.println("The value of " + book1.getName() + " in stock is $" + book1.calcStockValue());

Output:
7 copies of Extreme Alpinism in stock.
The value of Extreme Alpinism in stock is $139.65
public double calcStockValue()
{
    return numInStock * cost;
}
}
Question Two (20 points) – Given the Driver below, create the classes and method headers needed to get that Driver to compile. Note: You do not need to create any instance variables, nor do you need to put anything in the method bodies EXCEPT a bogus return value (ex: `return 0;`), if one is required. If I put your code into a computer, it should compile. Just make the classes, and make the correct method headers. You do not need to provide any comments in your code.

```java
public class Driver
{
    public static void main(String[] args)
    {
        Marker m1 = new Marker("red");
        Marker m2 = new Marker("blue");

        m1.drawLine(7.3);
        int val = m2.getUses();
    }
}
```

Write your class(es) here:

```java
public class Marker
{
    public Marker(String s)
    {
    }

    public void drawLine(double d)
    {
    }

    public int getUses()
    {
        return 0;
    }
}
```
Question Three (15 points) – Convert the following switch statement to an if-else statement. You do not need to provide any comments in your code.

```java
switch (size)
{
    case 6:
        price = 44.99;
        break;
    case 7:
        price = 49.99;
        break;
    case 8:
        price = 54.99;
        break;
    case 9:
        price = 59.99;
        break;
    case 10:
        price = 64.99;
        break;
    default:
        price = -1;
        System.out.println("size error");
        break;
}

if (size == 6)
{
    price = 44.99;
}
else if (size == 7)
{
    price = 49.99;
}
else if (size == 8)
{
    price = 54.99;
}
else if (size == 9)
{
    price = 59.99;
}
else if (size == 10)
{
    price = 64.99;
}
```
else
{
    price = -1;
    System.out.println("size error");
}
**Question Four (25 points)** – Answer the following short answer questions. If it asks for the method declaration, no method body is needed, just the first line: public…

a. (3 points) Write the method declaration for a public method that takes two integers as parameters and returns a boolean value.
   
   ```java
   public boolean method(int n1, int n2)
   ```

b. (2 points) Call your method from above on `instance1` with any input and store the result in a variable you declare.
   
   ```java
   boolean var = instance1.method(3, 2);
   ```

c. (2 points) What is the maximum value that the following line of code could print?
   
   ```java
   System.out.println((int) (Math.random() * 10 + 1));
   ```

10

d. (4 points) In the `University` class below, write the method declaration for the following `getCost` method:
   
   ```java
   double tuition = University.getCost(782.42, new Student(int credits));
   ```

   ```java
   public class University
   {
       public static double getCost(double val, Student student);
   }
   ```

e. (2 points) In general, should instance variables be public or private? Why?
   
   private, so users cannot access/modify information without permission.

f. (3 points) Create an instance of the `SpringBreak` class using a constructor that takes a string as the only parameter (use any string). Also assign your instance to a variable you declare.
   
   ```java
   SpringBreak sb = new SpringBreak("s");
   ```

g. (2 points) What does `n` equal after this switch statement is executed?
   
   ```java
   int n;
   int x = 1;
   switch(x)
   {
       case 2:
         n = 2;
       case 1:
         n = 1;
       default:
         n = 0;
   }
   ```

0

h. (4 points) What does the following code print?
   
   ```java
   int n = 7;
   System.out.println(n++); 7
   System.out.println(n); 8
   System.out.println(~n); 7
   System.out.println(n); 7
   ```
i. (3 points) What does a constructor do (they do at least two things)?
   Creates a new instance of the class in memory and initializes the variables to their desired initial values.