Java for Smartphone Development

Midterm

You are to finish the exam in two hours and email it to cislabs04@gmail.com by 9:35am Pacific Standard Time or 12:35pm Eastern Standard Time on Oct 22nd 2012. Timestamps on emails beyond 12:30pm will not be graded.

Please answer all questions. Each question is worth 10 points for a total score of 80 points.

Exam is open book and exams. You can use the Internet got reference. Pl. do not copy and paste code snippets from online or from each other as it would be considered cheating. Your solutions will be searched against other submitted solution and Internet references and if a match is found, a grade F will be assigned at the end of semester.

Please submit your answers in a single word document.

Question 1:
Concept Questions:
**a. Explain in your own words the role of JNI in enabling cross platform compatibility between Java and other programming languages?**

Answer:
Java Native Interface acts as a bridge providing API's to interoperate memory context between runtime environments authored using different programming languages such as C or C++. The bridge allows Java programs to access native functions in Operating Systems - such as calls from Java to create a new JFrame or a Socket use JNI. It enables binary compatibility between operating environments authored in different programming languages.

**b. Can interfaces be used to exhibit polymorphism. Explain with a coded example.**

Answer:
Yes.

```
public interface Act {
 void act();
}

class Actor1 implements Act {
 public void act() {
      System.out.println("To be, or not to be");
 }
}
```

```
class Actor2 implements Act {
 public void act() {
      System.out.println("Wherefore art thou Romeo?");
 }
}
public class TryOut {
 public static void main(String args[]) {
      Actor1 hamlet = new Actor1();
      Actor2 juliet = new Actor2();
      tryout(hamlet);
      tryout(juliet);
 }

 private static void tryout(Act actor) {
      actor.act();
 }
}
```

**Question 2:**
**a. How do you transfer input from a form to a JavaBean Variable? Explain your answer and write a short code snippet to demonstrate the concept.**

Answer:
SOLUTION WITH SERVLETS (POSTED EARLIER)

```
<html>
<head>
<title>New Page 1</title>
</head>
<body>
<h2>Login</h2>
<p>Please enter your username and password</p>
<form method="GET" action="/htmlform/LoginServlet">
<p> Username  <input type="text" name="username" size="20"></p>
<p> Password  <input type="text" name="password" size="20"></p>
<p><input type="submit" value="Submit" name="B1"></p>
</form>
<p> </p>
</body>
</html>
```

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class LoginServlet extends HttpServlet{
```

```
public void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
  response.setContentType("text/html");
  PrintWriter out = response.getWriter();
  String name = request.getParameter("username");
  String pass = request.getParameter("password");
  out.println("<html>");
  out.println("<body>");
  out.println( name + "  " + pass);
  out.println("</body></html>");
  }
}

SOLUTION WITH JAVABEAN
//Calling a private method
// JSP file

<HTML>
    <HEAD>
        <TITLE>Calling a Private Method</TITLE>
    </HEAD>
    <BODY>
        <H1>Calling a Private Method</H1>
        <jsp:useBean id="bean1" class="beans.Message" />
        The message is: <jsp:getProperty name="bean1" property="message" />
        <BR>
        <jsp:setProperty name="bean1" property="message" value="Hello again!"
/>
        Now the message is: <jsp:getProperty name="bean1" property="message"
/>
    </BODY>
</HTML>


package beans;
import java.io.Serializable;
public class Message implements Serializable
{
    private String message = "Hello from JSP!";
    public void setMessage(String m)
    {
        this.message = m;
    }
    public String getMessage()
    {
        return privateMessage();
    }
    private String privateMessage()
```

```
    {
        return this.message;
    }
    public Message()
    {
    }
}
```

**b. Explain how the hidden text fields can be used with Java?**

Answer:

```
<input type="hidden" id="age" name="age" value="23" />
<input type="hidden" id="DOB" name="DOB" value="01/01/70" />
```

Generally used for tracking Session ID, variables values that need to be persisted for form processing, or caching cookie values.

**Question 3:**

**a. Write a function to compute the square root of a number to a precision of 4.**

Answer:

```
public class Sqrt {

      public static float sqrt(float k)
      {
            float sq=0.0f,b=0.0f,precision=0.00001f,chsq=0.0f;
            sq=k;p=sq*sq;
            while(p-k>=precision)
            {
                  b=(sq+(k/sq))/2;
                  sq=b;
                  chsq=sq*sq;
            }
            return sq;
      }

      public static void main(String [] args)
      {
            System.out.printf("%7.4f",(sqrt(75333.0f)));
      }
}
```

**b. Can a serialized object be transferred over a network? Write a complete class to show how this might be achieved.**

Answer:

Yes a serialized object can be transferred over a network. A class demonstrating how it might be done.

```
public class DataXfer {
      //Write to socket
   private Socket a1;
   DataXfer () {
      a1 = new Socket();
   }

   void WriteToSocket()
   {
      FileOutputStream fos = new FileOutputStream("date.out");
      ObjectOutputStream oos = new
ObjectOutputStream(a1.getOutputStream());
      Date date = new Date();
      a1.
      oos.writeObject(date);
      oos.flush();
      oos.close();
      fos.close();
   }
   //read from socket
  void readFromSocket()
  {
      ObjectInputStream ois = new ObjectInputStream(a1.getInputStream()
);
      Date date = (Date) ois.readObject();
      System.out.println("The date is: "+date);
      ois.close();
      fis.close();
  }
}
```

**Question 4:**
**a. What is the role of response variable in JSP? Why is thread safety important in JSP and how it is achieved?**
Answer:

Response is a process to responding against a request. Response Object in JSP is used to send information, or output from web server to the user. Response Object sends output in form of stream to the browser. This can be redirecting one file to another file, response object can set cookie, set ContentType, Buffer size of page, caching control by browser, CharSet, expiration time in cache.
Response object tells browser what output has to use or display on browser, and what stream of data contain PDF, html/text, Word, Excel.

Thread safety is critical in JSP to avoid data corruption between request and response pairs. All JSP's are synchronized by default (as managed by the runtime environment).

**b. Explain the meaning of deadlock and how can it be avoided? An explanation for this question would suffice (coding is not required).**
**Answer:**

Deadlock occurs when two or more threads are waiting on common resources that are not exclusively owned
Thread A locks resource X and now requires resource Y
Thread B has resource Y locked and now requires resource X

Deadlock avoidance can be done by detecting the order in which locks are acquired and making code changes to ensure deadlock (as defined in definition) does not occur.

**Question 5:**
**Assuming that the following classes have been defined:**

```
public class A
{
      public static void method1()
      {
            System.out.println("A1");
      }
}

public class B extends A
{
      public static void method2()
      {
            System.out.println("B2");
      }
}
public class C extends B
{
      public static void method1()
      {
            System.out.println("C1");
      }
}
```

**And assuming the following objects have been defined:**

```
A a = new A();
B b = new B();
B other2 = new C();
```

**In the table below, indicate in the right-hand column the output produced by the statement in the left-hand column. If the statement**

**causes an error, fill in the right-hand column with either the phrase "compiler error"or "runtime error" to indicate when the error would be detected.**

| Statement | Output |
|---|---|
| a.method1(); | ___A1_____ |
| a.method2(); | ___Error_____ |
| b.method1(); | ___A1_____ |
| b.method2(); | ___B2_____ |
| other2.method1(); | ___A1_____ |
| other2.method2(); | ___B2_____ |

**Question 6:**
**In this question you will write two classes: Port and Ship. A Port is a city with a harbor. A Ship is an ocean-going vessel that travels between Ports. For the purposes of this question, we will assume that the world is flat, with every position specified by two coordinates x and y. The distance between two points (x1, y1) and (x2, y2) is given by the usual formula: square root of [(x1 - x2)2 + (y1 - y2)2].**
**A Port, when created, has a position. (Reminder: a position consists of two numbers.) It also has methods for finding out its position:**

- **getX() returns its x coordinate.**
- **getY() returns its y coordinate**

**A Ship, when created, has a name. In addition, it has these methods:**
**setPosition() takes as parameters an x coordinate and a y coordinate, and makes the Ship's position be (x, y).**
**setDestination() takes a Port as parameter and makes that the Ship's destination.**
**getDistance() returns the distance between the Ship and its destination Port.**
**toString() is the usual kind of toString, and returns a String giving the Ship's name and its distance from its destination.**

Answer:

```
class Port {
	int xCoordinate, yCoordinate;


	Port(int xCoordinate, int yCoordinate) {
		this.xCoordinate= xCoordinate;
		this.yCoordinate= yCoordinate;
	}

	public int getX() {
		return this.xCoordinate;
	}

	public int getY() {
		return this.yCoordinate;
	}
```

```
	public void display() {
		System.out.println("X Coordinate: " + getX());
		System.out.println("Y Coordinate: " + getY());
	}
}

class Ship {
	String name;
	int xCoordinate, yCoordinate;
	Port destination;


	Ship(String name) {
		this.name = name;
	}

	public void setPosition (int xCoordinate, int yCoordinate){
		this.xCoordinate = xCoordinate;
		this.yCoordinate = yCoordinate;
	}

	public void setDestination (Port destination){
		this.destination = destination;
	}

	public double getDistance() {
		double distance;
		distance = Math.sqrt((destination.getX() - xCoordinate) + (destination.getY() -
	yCoordinate));
		return distance;
	}

	public String toString() {
		return (this.name + " is " + Double.toString(getDistance()) + " miles away from
destination.");
	}
}

class Driver {
	public static void main (String[] args) {
		Port port = new Port(10, 4);
		Ship ship = new Ship("Java Ship");
		ship.setPosition(5, 1);
		ship.setDestination(port);
		System.out.println(ship.toString());
	}
}
```

**Question 7:**
**In this question, you must write part of a "main" method to solve a problem using two classes that you can assume are provided already.**

**Here are the classes that are provided. The contents of the methods are omitted, because you don't need to know how they work.**

```
class Person {
 public int howManyChildren() { ... }
  // returns an integer that is the number of children the Person has
```

```
   public String getName() { ... }
       // returns the name of the Person
}

class PersonReader {
   public PersonReader() { ... }
  // a constructor - creates its own Reader object, and whatever else it
needs

   public Person readPerson() { ... }
   // reads enough input data to make an object of the class Person, and
returns that object. (This is like
   // Format's readLine( ) method, which reads and returns a String.
}
```

**Complete the program below. It is intended to read the data about three people, storing each in an object of type Person. Then it prints the name of the Person with the largest number of children. Your code should handle case where two or more parents have the same number of children.**

```
public class WhoHasTheMostChildren {
   public static void main (String[] args) {

….....

   }
}
```

Answer:

```
public static void main (String[] args) {
       PersonReader reader = new PersonReader();
       Person[] persons = new Person[3];

       int largestNoChildren = 0;

       for (int i = 0; i < 3; i++) {
              persons[i] = reader.readerPerson();
              if (persons[i].howManyChildren > largestNoChildren)
                     largestNoChildren = persons[i].howManyChildren();
       }

       for (int i = 0; i < 3; i++) {
              if (persons[i].howManyChildren() = largestNoChildren)
                     System.out.println("Largest number of children: " + persons[i].getName());
       }
}
```

**Question 8:**

**a. For each code segment below, determine how many times the body of the loop is executed. Write one of the following answers after each: 0, 1, infinite, or > 1. Note that ″> 1″ means more than once but not infinite.**

**(a) for(int x=1; x<10; x++){**

```
      System.out.println(x);
}
```

```
(b) int x=1;
while(x<10){
      System.out.println(x);
}
```

```
(c) int x=1;
do{
      x = x*2;
} while(x>=8);
```

```
(d) int x=10;
while(x<10){
      System.out.println(x);
      x=x-1;
}
```

```
(e) int x=1;
    while(x!=10){
      x = x*2;
    }
```

Answer:

(a) >1
(b) infinite
(c) 1
(d) 0
(e) infinite

b.

**Complete the method sum that takes two arrays of integers as arguments. It returns null if either argument is null or if the arrays are not the same length. Otherwise it returns a new array, each of whose elements equals the sum of the corresponding elements in the input arrays. The input arrays should not be changed by the method.**

Answer:

```
class Summation {
      public static int[] sum(int[] one, int[] two)
      {
            if (one == null || two == null)
                  return null;
            if (one.length != two.length)
                  return null;
            int[] result = new int[one.length];
            for (int i = 0; i < one.length; i++)
                  result[i] = one[i] + two[i];
```

```
            return result;
        }
    }
```