1 (15 points)

Write the method `sumSeries` that given an integer argument `n` will return as a double the following sum:

\[ \frac{1}{n} + \frac{2}{n-2} + \frac{3}{n-4} + \ldots + \frac{n-2}{2} + \frac{n-1}{1} \]

Do not write a full program with input and output, just the method.

Hints/Notes:
- You need only a single loop, what kind of loop do you need?
- The return value is a double, make sure that as the sum is computed you are computing double division and not integer division.
- You don't need to check for valid values of `n`, assume it's an integer > 0.

Solution:

```java
public static double sumSeries(int n) {
    double retValue = 0;
    for ( int term = 0; term < n; ++term )
        retValue += (double)(term + 1) / (n - term);
    return retValue;
}
```
2 (15 points)

Write the method `sumDiagonal` that, given an integer two dimensional array `mat` representing a square matrix, will return as an integer the sum of the diagonal values.

For example:

```java
int[][] mat = {{2, 3, 5},
               {6, 7, 9},
               {2, 6, 7}};

int sd = sumDiagonal(mat);  // sd set to 16
```

Do not write a full program with input and output, just the method.

Hints/Notes:
•  `mat` is a square matrix, which means that the number of rows is the same as the number of columns.
•  How can the method tell the number of rows (and columns) in the array?
•  You need only a single loop, what kind of loop do you need?
•  You don't need to check for valid shape of `mat`, assume it is a perfect square matrix.

Solution:

```java
public static int sumDiagonal(int[][] mat) {
    int retValue = 0;
    for (int index = 0; index < mat.length; ++index)
        retValue += mat[index][index];
    return retValue;
}
```
3 (20 points)

(a) What will be the output of the following program:

```java
public class FinalExam {
    public static void main(String[] args) {
        func("The string");
        func("3.72");
        func(25);
    }

    public static void func(String arg) {
        System.out.println("Process a string.");
    }

    public static void func(double arg) {
        System.out.println("Process real value.");
    }

    public static void func(int arg) {
        System.out.println("Process integer.");
    }
}
```

**Solution:**

```
Process a string.
Process a string.
Process integer.
```

(b) Given the following class and subclass:

```java
1. class Parent {
2.    void func(int i) {
3.    }
4. }
5. class Child extends Parent {
6.    void func(char c) {
7.    }
8. }
9.    void func(int i) {
10. }
11. }
```

Indicate the line number of the method(s) in Child that is (are) (if none, leave blank):

**Overrides:** 9
**Overloads:** 7
3 (continued)

(c) What will be the output of the following program:

```java
public class FinalExam { 
    public static void main(String[] args) { 
        String[] values = {"Boston", "Miami", "Chicago"};
        try { 
            for (String val : values) { 
                func(val, 'o'); 
                System.out.println("Processed " + val);
            }
        } catch (Exception ex) { 
            System.out.println("Error: " + ex.getMessage());
        } 
        finally { 
            System.out.println("Processed values");
        }
    }

    public static void func(String arg, char search) throws Exception{
        boolean found = false;
        for (char c : arg.toCharArray())
            if (c == search) {
                found = true;
                break;
            }
        if (!found)
            throw new Exception(search + " not found in " + arg);
    }
}
```

Solution:

Processed Boston
Error: o not found in Miami
Processed values
3 (continued)

(d) What, if anything is illegal in the following code:

```java
public static void main(String[] args) {
    Child c = new Child();
}

public class Parent {
    protected int pVal;
    public Parent(int p) {
        pVal = p;
    }
}

public class Child extends Parent {
    protected int cVal;
    public Child(int c, int p) {
        super(p);
        cVal = c;
    }
}
```

Solution:

No default constructor available for `Child`, “Child c = new Child();” is therefore illegal.

(e) What will be the output of the following program:

```java
public class E {
    public static void main(String[] args) {
        int[] values = {3, 8, 5, 4, 9};
        System.out.println("Result: " + compute(values, 0, 4));
    }

    public static int compute(int[] values, int low, int high) {
        if (low == high)
            return values[low];
        int mid = (low + high) / 2;
        return compute(values, low, mid) + compute(values, mid + 1, high);
    }
}
```

Solution:

Result: 29

(compute call recursively and computes the sum of the numbers in the array.)
4 (25 points)

Design an abstract Shape class that will be used in a graphical application. Shape is extended by Circle and Rectangle.

The data elements of the Shape class are:
• Two private integer variables, x and y, representing the coordinates of the center of the object.

The methods of the Shape class are:
• A protected constructor that accepts two arguments: the coordinates of the center.
• Public methods to set and get the values of the coordinates of the center.
• The public abstract method isInside, which takes as arguments the coordinates of a point and returns the boolean value true/false if the point is inside the graphical object.

The additional data element of the Circle class is:
• The private integer variable radius.

The methods of the Circle class are:
• A constructor that accepts three arguments: the coordinates of the center and the radius.
• Implementation of the isInside method.

Note: to see if the point is inside the circle, verify if the distance from the point to the center is less than the radius. To compute the distance, assuming that px and py represent the coordinates of the point and cx and cy represent the coordinates of the center, use the formula:
  Math.pow(Math.pow(px – cx, 2) + Math.pow(py – cy, 2)), 0.5)

The additional data elements of the Rectangle class are:
• Two private integer variables, horizontal and vertical, representing the horizontal and vertical size of the rectangle.

The methods of the Rectangle class are:
• A constructor that accepts four arguments: the coordinates of the center and the size of the sides.
• Implementation of the isInside method
4 (continued)

Solution:

```java
public abstract class Shape {
    private int x, y;

    protected Shape(int x, int y) {
        this.x = x;
        this.y = y;
    }

    public int getX() {
        return x;
    }

    public void setX(int x) {
        this.x = x;
    }

    public int getY() {
        return y;
    }

    public void setY(int y) {
        this.y = y;
    }

    public abstract boolean isInside(int x, int y);
}

public class Circle extends Shape {
    int radius;

    public Circle(int x, int y, int radius) {
        super(x, y);
        this.radius = radius;
    }

    public boolean isInside(int x, int y) {
        double d = Math.pow(Math.pow(x - getX(),2) + Math.pow(y - getY(),2), 0.5);
        return d < radius;
    }
}

public class Rectangle extends Shape {
    int horizontal, vertical;

    public Rectangle(int x, int y, int h, int v) {
        super(x, y);
        horizontal = h;
        vertical = v;
    }

    public boolean isInside(int x, int y) {
        return (x > getX() - horizontal / 2) && (x < getX() + horizontal / 2) && (y > getY() - vertical / 2) && (y < getY() + vertical / 2);
    }
}
```
5 (25 points)

Write the definition of the abstract class **Vehicle** that contains:

- a private double variable, **maxSpeed**
- a private double variable, **currentSpeed**
- a protected constructor accepting a double used to initialize the **maxSpeed** instance variable
- an abstract method, **accelerate**, that accepts no parameters and returns nothing.
- a method **getCurrentSpeed** that returns the value of **currentSpeed**
- a method **getMaxSpeed** that returns the value of **maxSpeed**
- a method, **pedalToTheMetal**, that repeatedly calls **accelerate** until the speed of the vehicle is equal to **maxSpeed**. **pedalToTheMetal** returns nothing.

If you were to implement the class **Car** that extends **Vehicle**, which methods would you need to implement (at a minimum)?
5 (continued)

Solution:

```java
public abstract class Vehicle {
    private double maxSpeed;
    private double currentSpeed;

    protected Vehicle(double m) {
        maxSpeed = m;
    }

    public abstract void accelerate();

    public double getCurrentSpeed() {
        return currentSpeed;
    }

    public double getMaxSpeed() {
        return maxSpeed;
    }

    public void pedalToTheMetal() {
        while (currentSpeed < maxSpeed)
            accelerate();
    }
}
```

Car would have to implement a constructor (only constructor in Vehicle is protected) and the abstract method accelerate.