1 Short Answer (10 Points Each)

1. Create a method called `countLetters` that takes in a string and a character as parameters and returns the number of times that character is in the string.

   Solution:

```
public static int countLetters(String phrase, char target) {
    int count = 0;

    for (int i = 0; i < phrase.length(); i++) {
        if (phrase.charAt(i) == target)
            count++;
    }

    return count;
}
```

2. Create the method called `factorial` that takes in a long integer parameter n and returns a long with the value of $n!$. Remember that $n! = n \cdot (n-1) \cdot (n-2) \cdots 2 \cdot 1$ if $n > 0$ and $0! = 1$. You may assume that the value of the parameter is not negative.

   Solution:

```
public static long factorial(long n) {
    long fact = 1;

    while (n > 1) {
        fact *= n;
        n--;
    }

    return fact;
}
```

3. What are the three types of loops? For each type state if they are precondition or postcondition and what type of control they use.

   Solution:

Type	Check	Control
While	Precondition	Conditional
Do While	Postcondition	Conditional
For	Precondition	Count

4. Answer the following,

   (a) What is a sentinel value?

   Solution: A special value or values that control program flow, usually used with loops to stop the repetition. Normally it is a value that is either input or changed inside the body of a loop.

   (b) What is priming a loop?

   Solution: When using a precondition loop it is setting the condition so that you are guaranteed to enter the loop the first time. This usually means that a postcondition loop would be more appropriate.

5. Write a method called RollCount that will simulate the rolling of three die (6 sided). The method should roll the die as many times as necessary until all the die have the same value. The method should return the number of rolls it needed to do until all the die were equal.

Solution:

```
public static long RollCount() {
    long count = 0;
    int roll1;
    int roll2;
    int roll3;

    do {
        roll1 = (int) (Math.random() * 6 + 1);
        roll2 = (int) (Math.random() * 6 + 1);
        roll3 = (int) (Math.random() * 6 + 1);
        count++;
    } while (roll1 != roll2 || roll2 != roll3 || roll1 != roll3);

    return count;
}
```

2 Program Traces (15 Points Each)

1. For each of the given inputs, write the output of the program.

```java
import java.util.Scanner;

public class Exam2_Trace1 {

    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("n = ");
        int n = kb.nextInt();

        do {
            if (n % 2 == 0) {
                System.out.println("*** A ***");
                n--;
            } else {
                System.out.println("*** B ***");
                if (n % 3 == 0) {
                    System.out.println("*** C ***");
                    n = n / 2;
                } else {
                    System.out.println("*** D ***");
                    n--;
                    n = n % 5;
                }
            }
            System.out.println("n = " + n);
        } while (n > 0);
    }
}
```

(a) `n = 18`

Solution:

```
*** A ***
n = 17
*** B ***
*** D ***
n = 1
*** B ***
*** D ***
n = 0
```

(b) `n = 33`

Solution:

```
*** B ***
*** C ***
n = 16
*** A ***
n = 15
*** B ***
*** C ***
n = 7
*** B ***
*** D ***
n = 1
*** B ***
*** D ***
n = 0
```

(c) `n = 31`

Solution:

```
*** B ***
*** D ***
n = 0
```

2. For each input, write the output of the program.

```
import java.util.Scanner;

public class Exam2_Trace2 {

    public static int DoSomething(int a, int b) {
        System.out.println("In DoSomething: " + a + " " + b);
        if (a > b)
            return a / b;
        else
            return a + b;
    }

    public static int DoSomethingElse(int a, int b) {
        System.out.println("In DoSomethingElse: " + a + " " + b);
        if (a == b)
            return a;
        else
            return DoSomething(2 * b, a);
    }

    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("n = ");
        int n = kb.nextInt();
        System.out.print("m = ");
        int m = kb.nextInt();

        int t = DoSomething(m, n);
        int q = DoSomethingElse(m, n);
        int r = DoSomethingElse(t, q);

        System.out.println(n + " " + m + " " + t + " " + q + " " + r);
    }
}
```

(a)

```
n = 7
m = 2
```

Solution:

```
In DoSomething: 2 7
In DoSomethingElse: 2 7
In DoSomething: 14 2
In DoSomethingElse: 9 7
In DoSomething: 14 9
7 2 9 7 1
```

(b)

```
n = 5
m = 20
```

Solution:

```
In DoSomething: 20 5
In DoSomethingElse: 20 5
In DoSomething: 10 20
In DoSomethingElse: 4 30
In DoSomething: 60 4
5 20 4 30 15
```

3 Coding (15 Points Each)

1. Write a program that will ask the user for the number of darts they would like to throw. Have the program simulate the throwing of that many darts on a square ranging from -1 to 1 in both the x and y directions. have the program find the ratio of the number of darts that within the unit circle, and then multiply this ratio by 4 to obtain an approximation to π. Finally have the program output the approximation to π. A couple sample runs are below.

```
Number of darts: 100000
Pi is approximately = 3.14464
```

```
Number of darts: 100000000
Pi is approximately = 3.14134584
```

```java
import java.util.Scanner;

public class Exam2_01 {

    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("Number of darts: ");
        int n = kb.nextInt();
        int count = 0;

        for (int i = 0; i < n; i++) {
            double x = Math.random() * 2 - 1;
            double y = Math.random() * 2 - 1;
            if (x * x + y * y < 1)
                count++;
        }

        double PiApprox = 4.0 * count / n;
        System.out.println("Pi is approximately = " + PiApprox);
    }
}
```

2. Write a program that will ask the user for a phrase and output the number of words that have at least 5 letters in it. An example run is below.

```
Input Phrase: Far out in the uncharted backwaters of the unfashionable end of the
western spiral arm of the Galaxy lies a small unregarded yellow sun.

Number of words with at least 5 letters is 9
```

```java
import java.util.Scanner;

public class Exam2_02 {

    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("Input Phrase: ");
        String str = kb.nextLine();
        str += " ";
        int wordcount = 0;
        int lettercount = 0;

        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);
            if ((ch >= ’a’ && ch <= ’z’) || (ch >= ’A’ && ch <= ’Z’))
                lettercount++;
            else {
                if (lettercount >= 5)
                    wordcount++;

                lettercount = 0;
            }
        }
        System.out.print("Number of words with at least 5 letters is " + wordcount
            );
    }
}
```