Problem Discussed in Class

- Square Root
- Square Root to Precision
- Trailing Zeroes
- Factorial
- Binomial Coefficient
- Pascal's Triangle

```java
package com.prateek;

public class SquareRoot {

    //Integer Part
    static int squareRoot(int n){
        int i = 0;
        while(i*i <= n){
            i = i + 1;
        }
        return i-1;
    }
    // Upto P places
    static double squareRoot(int n,int p){

        double i = 0;
        double inc = 1;

        for(int work = 0; work <= p; work++){
            // finalise the digit at current place
            while(i*i <= n){
                i = i  + inc;
            }
            //one step backward
            i = i - inc;
            inc = inc/10;
        }
        return i;
    }
    //Given a number N, find trailing zeroes in n!
    static int findTrailingZeroes(int n){
        //Implement this function (Live Assignment)
        int ans = 0;
        int p = 5;
        while(n/p > 0){
            ans = ans + n/p;
            p = p*5;
        }
        return ans;
    }

    //LIVE Assignment 5 Mins (10.25)
    static int factorial(int n){
        int ans = 1;
```

```
        for(int i=1; i<=n; i++){
            ans = ans * i;
        }
        return ans;
    }
    static int binomialCoeff(int n,int r){
        int num = factorial(n);
        int t1 = factorial(n-r);
        int t2 = factorial(r);
        int denom = t1 * t2;

        int ans = num/denom;
        return ans;
    }
    //Pascal Triangle
    static void pascalTriangle(int n) {

        for (int i = 0; i < n; i++) {
            for (int j = 0; j <= i; j++) {
                System.out.print(binomialCoeff(i, j) + " ");
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        int n = 10;
        System.out.println(squareRoot(n));
        double ans = squareRoot(n,4);
        System.out.println(String.format("%.4f",ans));

        System.out.println(Math.sqrt(10));
        System.out.println(Math.ceil(10.3));
        System.out.println("Traling Zeroes"+ findTrailingZeroes(1000));
        System.out.println("NCR 5C3 = " + binomialCoeff(5,3));
        pascalTriangle(5);

    }
}
```

Homework (Additional ToDo Problems)

Q1. Inverted Pyramid Pattern (N=5)

```
* * * * *

 * * * *

  * * *

   * *

    *
```

```
    *
   * *
  * * *
 * * * *
* * * * *
```

Q2. Find Prime Factorisation of a Number N.

N = 15 Output : 3 X 5

N = 100 Output : 2 X 2 X 5 X 5