**Apna College**

**JAVA Class 1 (Codes)**

Q1. Print numbers from 5 to 1.

public static void printNumbers(int n) { if(n == 0) {

return;

}

System.out.println(n);

printNumbers(n-1);

}

Q2. Print numbers from 1 to 5.

public static void printNumbers(int n) { if(n == 6) {

return;

}

System.out.println(n);

printNumbers(n+1);

}

Q3. Print the sum of first n natural numbers.

class Recursion1 {

public static void printSum(int n, int sum) { if(n == 0) {

System.out.println(sum);

return;

}

sum += n;

printSum(n-1, sum);

}

public static void main(String args[]) { printSum(5, 0);

}

}

Q4. Print factorial of a number n.

**Apna College**

class Recursion1 {

public static void printFactorial(int n, int fact) { if(n == 0) {

System.out.println(fact);

return;

}

fact \*= n;

printFactorial(n-1, fact);

}

public static void main(String args[]) {

printFactorial(5, 1);

}

}

Q5. Print the fibonacci sequence till nth term.

class Recursion1 {

public static void printFactorial(int a, int b, int n) { if(n == 0) {

return;

}

System.out.println(a);

printFactorial(b, a+b, n-1);

}

public static void main(String args[]) {

printFactorial(0, 1, 5);

}

}

Q6. Print x^n (with stack height = n)

class Recursion1 {

public static int printPower(int x, int n) {

if(n == 0) {

return 1;

}

if(x == 0) {

return 0;

}

**Apna College**

int x\_ = printPower(x, n-1);

int xn = x \* x\_;

return xn;

}

public static void main(String args[]) {

int x = 2, n = 5;

int output = printPower(x, n);

System.out.println(output);

}

}

Q7. Print x^n (with stack height = logn)

class Recursion1 {

public static int printPower(int x, int n) {

if(n == 0) {

return 1;

}

if(n % 2 == 0) {

return printPower(x, n/2) \* printPower(x, n/2); }

else {

return x \* printPower(x, n/2) \* printPower(x, n/2); }

}

public static void main(String args[]) {

int x = 2, n = 5;

int output = printPower(x, n);

System.out.println(output);

}

}