***Java Programming***

***Section 4.3 practice***

**1. Linear Recursion Class**

public class Linear {

// Method to compute factorial using recursion

public static double factorial(double n) {

// Base case: factorial of 1 or less is 1

if (n <= 1) {

return 1;

} else {

// Recursive case: n \* factorial(n - 1)

return n \* factorial(n - 1);

}

}

public static void main(String[] args) {

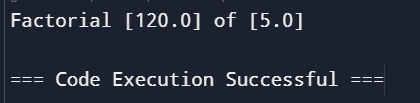
double d = 5.0;

double result = factorial(d);

System.out.printf("Factorial [%.1f] of [%.1f]%n", result, d);

}

}



**2. Non-Linear Recursion Class**

public class NonLinear {

// Method to compute Fibonacci value using recursion

public static double fibonacci(double n) {

// Base cases: Fibonacci of 0 or 1 is the number itself

if (n < 2) {

return n;

} else {

// Recursive case: fibonacci(n - 1) + fibonacci(n - 2)

return fibonacci(n - 1) + fibonacci(n - 2);

}

}

public static void main(String[] args) {

double d;

// Check if arguments are provided

if (args.length > 0) {

d = Double.parseDouble(args[0]);

} else {

d = 5.0;

}

// Print Fibonacci values for the range from 0 to d

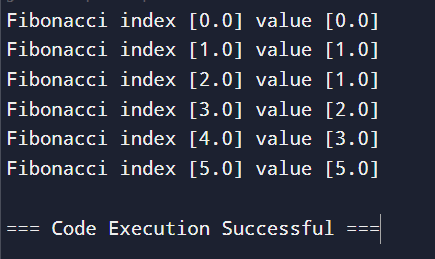
for (int i = 0; i <= d; i++) {

System.out.printf("Fibonacci index [%.1f] value [%.1f]%n", (double) i, fibonacci(i));

}

}

}



**3. Factorial of 7**

public class Main {

// Linear class with factorial method

public static class Linear {

public static double factorial(double n) {

// Base case: factorial of 1 or less is 1

if (n <= 1) {

return 1;

} else {

// Recursive case: n \* factorial(n - 1)

return n \* factorial(n - 1);

}

}

public static void main(String[] args) {

double d = 5.0;

double result = factorial(d);

System.out.printf("Factorial [%.1f] of [%.1f]%n", result, d);

}

}

// NonLinear class with fibonacci method

public static class NonLinear {

public static double fibonacci(double n) {

// Base cases: Fibonacci of 0 or 1 is the number itself

if (n < 2) {

return n;

} else {

// Recursive case: fibonacci(n - 1) + fibonacci(n - 2)

return fibonacci(n - 1) + fibonacci(n - 2);

}

}

public static void main(String[] args) {

double d;

// Check if arguments are provided

if (args.length > 0) {

d = Double.parseDouble(args[0]);

} else {

d = 5.0;

}

// Print Fibonacci values for the range from 0 to d

for (int i = 0; i <= d; i++) {

System.out.printf("Fibonacci index [%.1f] value [%.1f]%n", (double) i, fibonacci(i));

}

}

}

public static void main(String[] args) {

// Run Linear example

Linear.main(args);

// Run NonLinear example

NonLinear.main(args);

}

}

A screen shot of a computer program

Description automatically generated