**Laporan Praktikum**

**Fungsi Rekursif**



Disusun oleh:

Tomi Martino Affandi

2341720245

D-IV Teknik Informatika - 1E

Jl.Soekarno Hatta No.9, Jatimulyo, Kec.Lowokwaru, Kota Malang Jawa Timur 65141

Phone : (0341)404424, 404425

E-email : Polinema.ac.id

**1. Tujuan**

* Mahasiswa memahami konsep fungsi rekursif
* Mahasiswa mampu mengimplementasikan fungsi rekursif dalam kode program

**2. Praktikum**

**2.1 Percobaan 1**

public class Percobaan1\_27 {

    static int faktorialRekursif(int n) {

        if (n == 0) {

            return (1);

        }else{

            return (n \* faktorialRekursif(n - 1));

        }

    }

    static int faktorialIteratif(int n){

        int faktor = 1;

        for (int i = n; i >= 1; i--) {

            faktor = faktor \* i;

        }

        return faktor;

    }

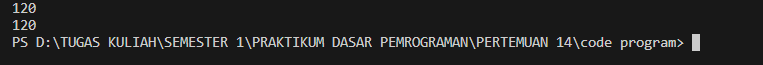
    public static void main(String[] args) {

        System.out.println(faktorialRekursif(5));

        System.out.println(faktorialIteratif(5));

    }

}

****

**Pertanyaan**

1. Fungsi rekursif adalah fungsi yang dapat memanggil dirinya sendiri sehingga proses nya akan berulang ulang(looping)

2. penghitungan angka berpangkat dapat menggunakan fungsi rekursif

3. hasilnya sama namun alurnya berbeda, pada fungsi rekursif akan melakukan “fase ekspansi terlebih dahulu lalu di substitusikan sementara pada fungsi iteratif akan melakukan perulangan biasa(for loop)

**2.2 Percobaan 2**

import java.util.Scanner;

public class Percobaan2\_27 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Bilangan yang dihitunng: ");

        int bilangan = sc.nextInt();

        System.out.print("Pangkat: ");

        int pangkat = sc.nextInt();

        System.out.println(hitungPangkat(bilangan, pangkat));

    }

    static int hitungPangkat(int x, int y){

        if (y == 0) {

            return 1;

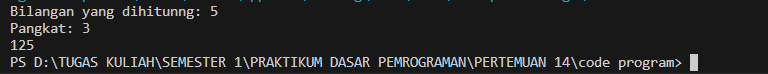
        }else {

            return (x \*hitungPangkat(x, y -1));

        }

    }

}



**Pertanyaan**

1. proses pemanggilan fungsi dijalankan sebanyak value dari y atau pangkat

2.

import java.util.Scanner;

public class Percobaan2\_27 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Bilangan yang dihitung: ");

        int bilangan = sc.nextInt();

        System.out.print("Pangkat: ");

        int pangkat = sc.nextInt();

        System.out.print(hitungPangkat(bilangan, pangkat));

    }

    static int hitungPangkat(int x, int y) {

        if (y == 0) {

            System.out.print("1 = ");

            return 1;

        } else {

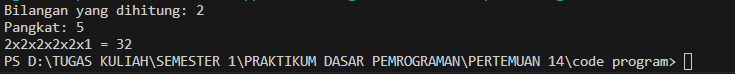
            System.out.print(x + "x");

            return x \* hitungPangkat(x, y - 1);

        }

    }

}



**2.3 Percobaan 3**

import java.util.Scanner;

public class Percobaan3\_27 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Saldo awal: ");

        double saldoAwal = sc.nextDouble();

        System.out.print("lama investasi (tahun): ");

        int tahun = sc.nextInt();

        System.out.print("Jumlahh saldo setelah " + tahun + " tahun :");

        System.out.print(hitungLaba(saldoAwal, tahun));

    }

    static double hitungLaba(double saldo, int tahun){

        if (tahun == 0) {

            return (saldo);

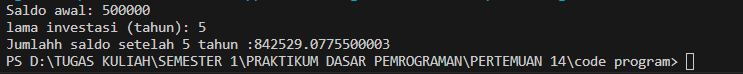
        }else{

            return (1.11 \* hitungLaba(saldo, tahun - 1));

        }

    }

}



**Pertanyaan**

1.

Base case :

 if (tahun == 0) {

            return (saldo);

Recursion call :

else{

            return (1.11 \* hitungLaba(saldo, tahun - 1));

2. Jabarkan trace fase ekspansi dan fase subtitusi algoritma perhitungan laba di atas jika diberikan nilai **hitungLaba(100000,3)**

Fase Substitusi :

=(1.11 \*(1.11\*(1.11\*100000)))

=(1.11 \*(1.11\*111.000))

=(1.11 \*123.210)

=136.763,1

**Tugas**

1.

import java.util.Scanner;

public class DereteDesendingRekursif {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukan angka: ");

        int angka = sc.nextInt();

        System.out.println(deretDescending(angka));

    }

    static int  deretDescending(int n){

        if (n == 0) {

            return 1;

        }else {

            System.out.println(n +" ");

            return deretDescending(n - 1);

        }

    }

}

2.

import java.util.Scanner;

public class PenjumlahanRekursif {

        static int jumlahKan(int m, int n){

        System.out.print(n);

        if (n >= m)

            return n;

        System.out.print(" + ");

        return n + jumlahKan(m, n + 1);

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan bilangan: ");

        int n = sc.nextInt();

        System.out.println(" = " + jumlahKan(n, 1));

        sc.close();

    }

}

3.

import java.util.Scanner;

public class CekPrimaRekursif {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukan angka: ");

        int angka = sc.nextInt();

        if (cekPrima(angka, angka - 1)) {

            System.out.println(angka + " adalah angka prima.");

        } else {

            System.out.println(angka + " bukan angka prima.");

        }

    }

    static boolean cekPrima(int m, int n){

        if (m <= 1) {

            return false;

        }else if (n == 1) {

            return true;

        }else if (m % n == 0) {

            return false;

        }else {

            return cekPrima(m, n -1);

        }

    }

}

4.

import java.util.Scanner;

public class Fibonaci {

    static int fib(int n) {

        if (n <= 1)

            return n;

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

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Bulan ke: ");

        int n = sc.nextInt();

        System.out.println("Jumlah Marmot: " + fib(n));

        sc.close();

    }

}

