

NAMA : QULBI KHUTSI AZZUMI

NIM : 244107020050

NO : 22

KELAS : TI-1D

JOBSHEET 12

PERCOBAAN 1

1. Buat project baru bernama Rekursif, dan buat file Java dengan nama Percobaan1

Percobaan1.java U

2. Buat fungsi static dengan nama faktorialRekursif(), dengan tipe data kembalian fungsi int dan memiliki 1 parameter dengan tipe data int berupa bilangan yang akan dihitung nilai faktorialnya

```
static int faktorialRekursif(int n) {  
    if (n == 0) {  
        return (1);  
    } else {  
        return (n * faktorialRekursif(n - 1));  
    }  
}
```

3. Buat lagi fungsi static dengan nama faktorialIteratif(), dengan tipe data kembalian fungsi int dan memiliki 1 parameter dengan tipe data int berupa bilangan yang akan dihitung nilai faktorialnya.

```
static int faktorialIteratif(int n) {  
    int faktor = 1;  
    for (int i = n; i >= 1; i--) {  
        faktor = faktor * i;  
    }  
    return faktor;  
}
```

4. Buatlah fungsi main dan lakukan pemanggilan terhadap kedua fungsi yang telah dibuat sebelumnya, dan tampilkan hasil yang didapatkan.

```
Run | Debug  
public static void main(String[] args) {  
    System.out.println(faktorialRekursif(5));  
    System.out.println(faktorialIteratif(5));  
}
```

5. Jalankan program tersebut. Amati apa yang terjadi!

```
Page 15030  
120  
120  
PS D:\KULIAH
```

PERTANYAAN

1. Apa yang dimaksud dengan fungsi rekursif?

Fungsi terdapat perintah untuk memanggil fungsi itu sendiri (dirinya sendiri). Dengan demikian, proses pemanggilan fungsi akan terjadi secara berulang-ulang

2. Bagaimana contoh kasus penggunaan fungsi rekursif

Hitung faktorial, deret bilangan, fibonacci

3. Pada Percobaan1, apakah hasil yang diberikan fungsi faktorialRekursif() dan fungsi faktorialIteratif() sama? Jelaskan perbedaan alur jalannya program pada penggunaan fungsi rekursif dan fungsi iteratif!

Sama, namun jalannya alur berbeda. Alur Fungsi faktorialRekursif() memiliki nilai 5, lalu jika n sama dengan 0 maka nilai akan mengembalikan nilai 1, jika tidak sama dengan 0 maka akan mengeksekusi $n * \text{faktorialRekursif}(n-1)$

PERCOBAAN 2

1. Pada project Rekursif, dan buat file Java dengan nama Percobaan2

Percobaan2.java 1,0

2. Buat fungsi static dengan nama hitungPangkat(), dengan tipe data kembalian fungsi int dan memiliki 2 parameter dengan tipe data int berupa bilangan yang akan dihitung pangkatnya dan bilangan pangkatnya

```
static int hitungPangkat(int x, int y) {  
    if (y == 0) {  
        return (1);  
    } else {  
        return (x * hitungPangkat(x, y - 1));  
    }  
}
```

3. Buatlah fungsi main dan deklarasikan Scanner dengan nama sc. Buatlah dua buah variabel bertipe int dengan nama bilangan dan pangkat. Tambahkan kode berikut ini untuk menerima input dari keyboard

```
public static void main(String[] args) {  
    Scanner sc = new Scanner(System.in);  
    int bilangan, pangkat;  
    System.out.print(s:"Bilangan yang dihitung: ");  
    bilangan = sc.nextInt();  
    System.out.print(s:"Pangkat: ");  
    pangkat = sc.nextInt();  
}
```

4. Lakukan pemanggilan fungsi hitungPangkat yang telah dibuat sebelumnya dengan mengirimkan dua nilai parameter

```
System.out.println("Hitung: " +bilangan+ " ^ " +pangkat );  
System.out.println("Hasil: "+hitungPangkat(bilangan, pangkat));
```

5. Jalankan program tersebut. Amati apa yang terjadi!

```
Bilangan yang dihitung: 9  
Pangkat: 2  
Hitung: 9 ^ 2  
Hasil: 81
```

PERTANYAAN

1. Pada Percobaan2, terdapat pemanggilan fungsi rekursif hitungPangkat(bilangan, pangkat) pada fungsi main, kemudian dilakukan pemanggilan fungsi hitungPangkat() secara berulang kali. Jelaskan sampai kapan proses pemanggilan fungsi tersebut akan dijalankan!

Proses pemanggilan fungsi akan berhenti ketika y sama dengan 0

2. Tambahkan kode program untuk mencetak deret perhitungan pangkatnya. Contoh :
hitungPangkat(2,5) dicetak 2x2x2x2x2x1 = 32

```
static int hitungPangkat(int x, int y) {  
    if (y == 0) {  
        System.out.print(s:"1");  
        return (1);  
    } else {  
        System.out.print(x+ " x ");  
        return (x * hitungPangkat(x, y - 1));  
    }  
}
```

Bilangan yang dihitung: 2
Pangkat: 5
Hitung: 2 ^ 5
2 x 2 x 2 x 2 x 2 x 1
Hasil: 32

PERCOBAAN 3

1. Pada project Rekursif, dan buat file Java dengan nama Percobaan3

Percobaan3.java 1,0

2. Buat fungsi static dengan nama hitungLaba(), dengan tipe data kembalian fungsi double dan memiliki 2 parameter dengan tipe data int berupa saldo investor dan lamanya investasi. Pada kasus ini dianggap laba yang ditentukan adalah 11% per tahun. Karena perhitungan laba adalah laba * saldo, sehingga untuk menghitung besarnya uang setelah ditambah laba adalah saldo + laba * saldo. Dalam hal ini, besarnya laba adalah 0.11 * saldo, dan saldo dianggap 1 * saldo, sehingga 1 * saldo + 0.11 * saldo dapat diringkas menjadi 1.11 * saldo untuk perhitungan saldo setelah ditambah laba (dalam setahun).

```
static double hitungLaba(double saldo, int tahun){  
    if (tahun == 0 ) {  
        return (saldo);  
    }else{  
        return (1.11 * hitungLaba(saldo, tahun-1));  
    }  
}
```

3. Buatlah fungsi main dan deklarasikan Scanner dengan nama sc. Buatlah sebuah variabel bertipe double dengan nama saldoAwal dan sebuah variabel bertipe int bernama tahun. Tambahkan kode berikut ini untuk menerima input dari keyboard. Lakukan pemanggilan fungsi hitungLaba yang telah dibuat sebelumnya dengan mengirimkan dua nilai parameter

```

Run | Debug
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    double saldoAwal;
    int tahun;

    System.out.print(s:"Jumlah saldo awal: ");
    saldoAwal = sc.nextDouble();
    System.out.print(s:"Lamanya investasi (Tahun): ");
    tahun = sc.nextInt();

    System.out.print("Jumlah saldo setelah " + tahun + " tahun : ");
    System.out.print(hitungLaba(saldoAwal, tahun));
}

```

4. Jalankan program tersebut. Amati apa yang terjadi!

```

Page(F50564C638C7B1D157CE88B9C498C02D) (reunat.java)
Jumlah saldo awal: 10000
Lamanya investasi (Tahun): 1
Jumlah saldo setelah 1 tahun : 11100.000000000002

```

PERTANYAAN

1. Pada Percobaan3, sebutkan blok kode program manakah yang merupakan “base case” dan “recursion call”!

- Base case

```

if (tahun == 0 ) {
    return (saldo);
}

```

- Recursion call

```

}else{
    return (1.11 * hitungLaba(saldo, tahun-1));
}

```

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

```

ptionmessages -cp C:\Users\ASUS\AppData\Roaming\Co
Jumlah saldo awal: 10000
Lamanya investasi (Tahun): 3
Jumlah saldo setelah 3 tahun : 13676.310000000005

```

Fase ekspansi:

- $1.11 * \text{hitungLaba}(10000, 3-1)$
- $1.11 * \text{hitungLaba}(10000, 2-1)$
- $1.11 * \text{hitungLaba}(10000, 1-1)$

Hasil: 10000

Fase substitusi

- Hasil = 10000
- $1.11 * 10000 = 11100 \rightarrow \text{ekspansi } 3-1$

- $1.11 * 11100 = 12321 \rightarrow \text{ekspansi 2-1}$
- $1.11 * 12321 = 1367631 \rightarrow \text{ekspansi 1-1}$

TUGAS

1. Buatlah program untuk menampilkan bilangan n sampai 0 dengan menggunakan fungsi rekursif dan fungsi iteratif. (DeretDescendingRekursif).

```

DeretDescendingRekursif.java > DeretDescendingRekursif > deretIteratif(int)
1  import java.util.Scanner;
2
3  class DeretDescendingRekursif {
4      public static void deretRekursif(int n) {
5          if (n < 0) {
6              return;
7          }
8          System.out.print(n + " ");
9          deretRekursif(n - 1);
10     }
11
12     static void deretIteratif(int n) {
13         for (int i = n; i >= 0; i--) {
14             System.out.print(i + " ");
15         }
16     }
17
18     public static void main(String[] args) {
19         Scanner input = new Scanner(System.in);
20         System.out.print(s:"Banyak deret: ");
21         int n = input.nextInt();
22
23         System.out.println(x:"Deret dengan Rekursif:");
24         deretRekursif(n);
25
26         System.out.println(x:"\n\nDeret dengan Iteratif:");
27         deretIteratif(n);
28     }
29 }

```

Hasil:

```

Page 1/30
Banyak deret: 10
Deret dengan Rekursif:
10 9 8 7 6 5 4 3 2 1 0

Deret dengan Iteratif:
10 9 8 7 6 5 4 3 2 1 0

```

2. Buatlah program yang di dalamnya terdapat fungsi rekursif untuk menghitung penjumlahan bilangan. Misalnya f = 8, maka akan dihasilkan $1+2+3+4+5+6+7+8 = 36$ (PenjumlahanRekursif).

```

1  import java.util.Scanner;
2
3  public class PenjumlahanRekursif {
4      static int hitungPenjumlahan(int jumlah) {
5          if (jumlah == 1) {
6              return 1;
7          } else {
8              return jumlah + hitungPenjumlahan(jumlah - 1);
9          }
10     }
11     static String deret (int nilai){
12         if (nilai == 1) {
13             return "1";
14         }else{
15             return deret(nilai-1)+ "+" +nilai;
16         }
17     }
18     public static void main(String[] args) {
19         Scanner input = new Scanner(System.in);
20         System.out.print(s:"Masukkan angka yang dijumlahkan: ");
21         int nilai = input.nextInt();
22
23         if (nilai < 1) {
24             System.out.println(x:"angka tidak valid");
25         }else{
26             String deret = deret(nilai);
27             int hasil = hitungPenjumlahan(nilai);
28             System.out.println("Deret: " +deret);
29             System.out.println("Hasil: " +hasil);
30         }
31     }
32 }

```

Hasil:

```

Masukkan angka yang dijumlahkan: 8
Deret: 1+2+3+4+5+6+7+8
Hasil: 36

```

3. Sepasang marmut yang baru lahir (jantan dan betina) ditempatkan pada suatu pembiakan. Setelah dua bulan pasangan marmut tersebut melahirkan sepasang marmut kembar (jantan dan betina). Setiap pasangan marmut yang lahir juga akan melahirkan sepasang marmut juga setiap 2 bulan. Berapa pasangan marmut yang ada pada akhir bulan ke-12? Buatlah programnya menggunakan fungsi rekursif! (Fibonacci). Berikut ini adalah ilustrasinya dalam bentuk tabel.

Bulan ke-	Jumlah Pasangan		Total Pasangan
	Produktif	Belum Produktif	
1	0	1	1
2	0	1	1
3	1	1	2
4	1	2	3
5	2	3	5
6	3	5	8
7	5	8	13
8	8	13	21
9	13	21	34
10	21	34	55
11	34	55	89
12	55	89	144

```

Fibonacci.java > Fibonacci > main(String[])
1  import java.util.Scanner;
2
3  public class Fibonacci {
4      static int hitungMarmut(int bulan){
5          if (bulan == 1) {
6              return 1;
7          }
8          if (bulan == 2) {
9              return 1;
10         }
11         return hitungMarmut(bulan-1) + hitungMarmut(bulan-2);
12     }
13     Run | Debug
14     public static void main(String[] args) {
15         Scanner input = new Scanner(System.in);
16
17         System.out.print(s: "Masukkan bulan ke-" );
18         int bulan = input.nextInt();
19
20         if (bulan < 0) {
21             System.out.println(x: "bulan tidak valid");
22         } else {
23             int pasanganMarmut = hitungMarmut(bulan);
24             System.out.println("Jumlah pasangan marmut pada bulan ke-" + bulan + " sebanyak: " + pasanganMarmut);
25         }
26     }
27 }

```

Hasil:

```

Page 1/555 15:55:13 15:55:13 15:55:13 15:55:13 (F:\code\java\Gut_13\Main_11
Masukkan bulan ke-12
Jumlah pasangan marmut pada bulan ke-12 sebanyak: 144

```