**INFORMATIKA**

**UNIVERSITAS SANATA DHARMA**

**PRAKTIKUM STRUKTUR DATA LINIER**

**Rekursif**

1. **Tujuan praktikum:**
   1. Mahasiswa mampu memahami mendebug metode rekursif
2. **Dasar Teori**

Biasanya, method memanggil method yang lain. Contoh, method main() memanggil method square()

Method rekursif adalah method yang memanggil dirinya sendiri

Rekursif diperlukan karena :

* Beberapa masalah lebih mudah dipecahkan menggunakan rekursif.
* Contoh kasus:
  + Pelacakan directory atau sistem file.
  + Pelacakan pada pohon
  + Beberapa algoritma Pengurutan Data

Untuk membuat method rekursif, maka harus mencari :

* **Base case**: harus memiliki nilai dasar yang dapat dipecahkan tanpa harus melakukan rekursif.
* **Making Progress**: untuk kasus yang dipecahkan secara rekursif, maka pemanggilan rekursif harus dibuat hingga menuju ke kondisi base case.

**C. Tugas**

1. Terdapat method rekursif sebagai berikut :

public static int paijo (int x)

{

if (x == 0)

return 0;

return 2 \* paijo (x - 1) + x \* x;

}

* 1. Apa tujuan method tersebut ? *(5 poin)*
  2. Base case (*5 poin*)
  3. Making progress (*5 poin*)
  4. Stack untuk setiap langkah (*15 poin*)

1. Terdapat method rekursif sebagai berikut :

public static void upAndDown (int n)

{

System.out.print ("\nLevel: " + n);

if (n < 4)

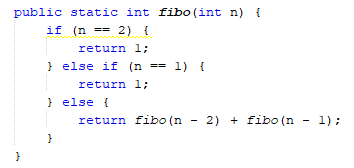
upAndDown (n+1);

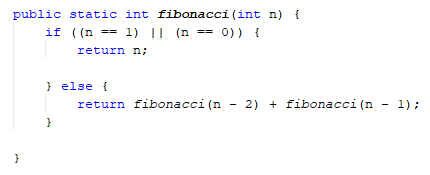
System.out.print ("\nLEVEL: " + n);

}

* 1. Apa tujuan method tersebut ? *(5 poin)*
  2. Base case (*5 poin*)
  3. Making progress (*5 poin*)
  4. Stack untuk setiap langkah (*15 poin*)

1. **Buatlah stack masing-masing untuk kedua program rekursif fibonacci dibawah ini ketika mencetak bilangan Fibonacci ke-6:**





**Apa perbedaan kedua program dalam stack?**

*(2 x 15 poin)*

To be a winner, all you have to do is to give all you have !! **@**