

Perbandingan atau Perbedaan Rekursif vs Perulangan (Looping) pada Java

Muhammad Azka Raki

NIM. 2311016110005

Definisi:

- **Rekursif:** Metode pemanggilan fungsi secara berulang dengan cara memanggil dirinya sendiri sampai mencapai kondisi tertentu (base case).
- **Looping:** Instruksi berulang dalam suatu blok kode menggunakan struktur seperti for, while, atau do-while, yang akan berhenti setelah kondisi tertentu terpenuhi.

Perulangan (Looping):

- Kelebihan

1. Mudah dipahami dan mudah dilakukan debugging ketika ada perulangan yang salah.
2. Dapat melakukan nested loop atau yang disebut dengan looping bersarang
3. Proses lebih singkat karena perulangan terjadi pada kondisi yang telah disesuaikan
4. Jarang terjadi overflow karena batasan dan syarat perulangan yang jelas
5. Umumnya lebih cepat, terutama pada jumlah iterasi besar, karena eksekusinya langsung tanpa memanggil fungsi berulang.
6. Lebih efisien dalam penggunaan memori karena tidak menggunakan stack tambahan untuk setiap iterasi.

- Kekurangan

1. Tidak dapat menggunakan batasan berupa fungsi
2. Perulangan dengan batasan yang luas akan menyulitkan dalam pembuatan program perulangan itu sendiri.

Rekursif:

- Kelebihan

1. Sangat mudah untuk melakukan perulangan dengan batasan yang luas dalam artian melakukan perulangan dalam skala yang besar.
2. Dapat melakukan perulangan dengan batasan fungsi
3. Cocok untuk masalah yang dapat dibagi menjadi sub-masalah yang lebih kecil (misalnya, pencarian biner, factorial, Fibonacci), sehingga dapat membuat kode lebih mudah dibaca untuk algoritma seperti ini.

- Kekurangan

1. Tidak bisa melakukan nested loop atau looping bersarang
2. Biasanya membuat fungsi sulit untuk dipahami, hanya cocok untuk persoalan tertentu saja
3. Memerlukan stack yang lebih besar, sebab setiap kali fungsi dipanggil, variabel lokal dan parameter formal akan ditempatkan ke stack dan ada kalanya akan menyebabkan stack tak cukup lagi (*Stack Overrun*)

4. Setiap kali fungsi rekursif dipanggil, tumpukan (stack) baru dibuat, yang dapat menyebabkan konsumsi memori lebih tinggi. Hal ini bisa menyebabkan *StackOverflowError* jika rekursinya terlalu dalam.
5. Proses agak berbelit-belit karena terdapat pemanggilan fungsi yang berulang-ulang dan pemanggilan data yang ditumpuk
6. Biasanya lebih lambat dibandingkan looping karena overhead dalam pemanggilan fungsi.

Kapan Harus Menggunakan Rekursif atau Looping

- Gunakan Rekursif ketika masalah dapat dibagi menjadi sub-masalah serupa dan memiliki kondisi batas yang jelas, misalnya:
 - Masalah berbasis pohon atau graf, seperti traversal pohon.
 - Algoritma *divide and conquer*, seperti merge sort atau quick sort.
 - Algoritma matematis tertentu seperti menghitung faktorial atau deret Fibonacci.
- Gunakan Looping ketika iterasi jumlah tetap diperlukan, atau ketika efisiensi memori dan performa penting, misalnya:
 - Pengulangan sederhana seperti mencetak deret angka atau iterasi array.
 - Kasus di mana jumlah iterasi diketahui di awal.