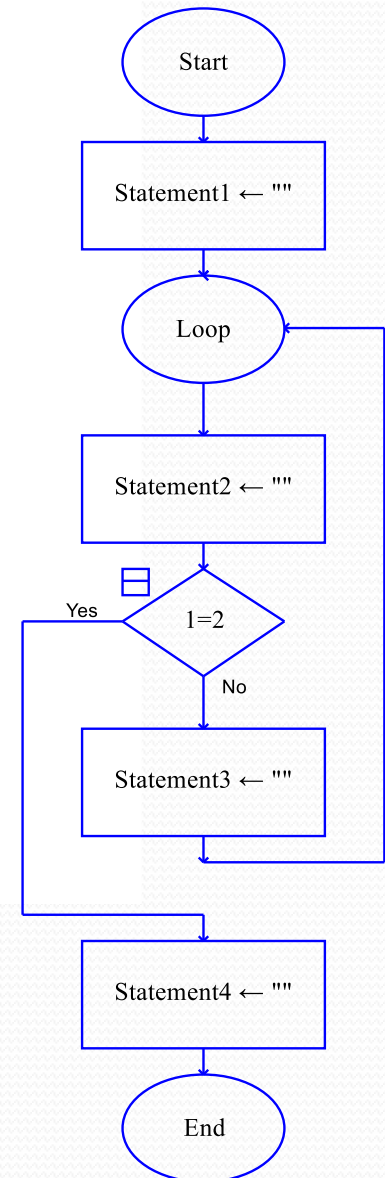


Algoritma & Pemrograman

Pertemuan 4: Struktur Kontrol Pemrograman :
Loop Bersarang (Nested Loop)

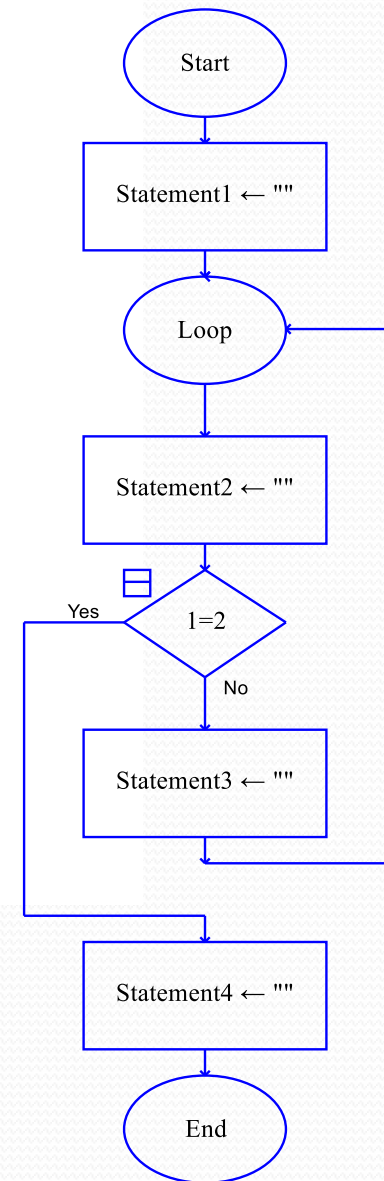
Perulangan

- **Statement1** dieksekusi tepat 1 kali sebelum loop dimulai
- **Statement2** akan selalu dieksekusi setidaknya 1 kali karena ia berada sebelum statement decision
- Jika ekspresi decision bernilai “Yes”, maka loop berakhir dan kontrol dilanjutkan ke **statement4**
- Jika ekspresi decision bernilai “No”, maka kontrol berlanjut ke **statement3** dan **statement3** yang dieksekusi berikutnya. Lalu kontrol kembali ke atas (ke statement Loop) yang memulai kembali proses perulangan
- Perhatikan bahwa **statement2** dijamin dieksekusi setidaknya 1 kali. Perhatikan juga bahwa **statement3** ada kemungkinan tidak akan pernah dieksekusi sama sekali



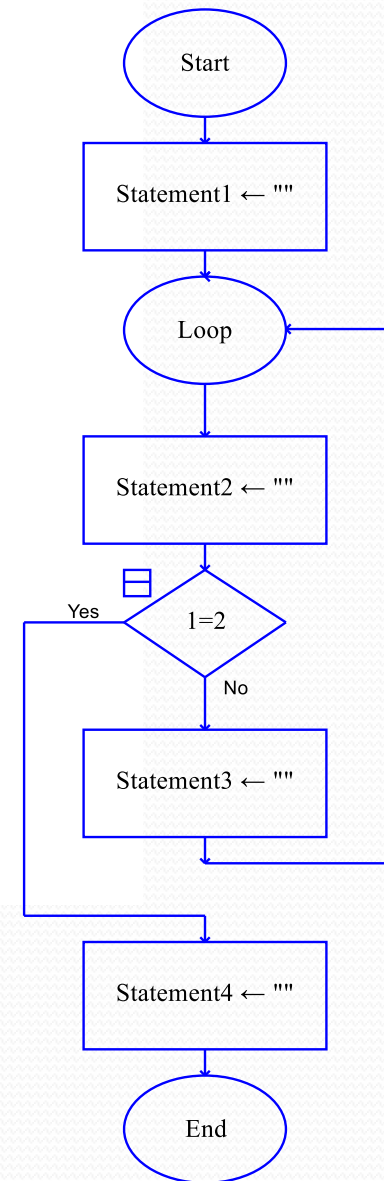
Perulangan (lanjutan)

- Pada contoh di samping, statement2 dapat dihilangkan (maka decision menjadi statement pertama pada loop yang dieksekusi) atau dapat berupa sekelompok statement lainnya. Loop akan tetap dieksekusi
- Sama halnya dengan Statement3 yang dapat dihilangkan atau diganti dengan sekelompok statement lainnya
- Selain itu, semua statement di atas/di bawah statement Decision dapat berupa statement Loop lain!
- Jika ada statement Loop muncul di dalam statement Loop, ini disebut **“nested loop”**



Perulangan (lanjutan)

- Dapat juga dimungkinkan statement Decision tidak pernah bernilai “Yes”. Pada kondisi ini, kita akan masuk dalam keadaan “loop tak terbatas (*infinite loop*)” yang tidak akan pernah berhenti. Jika ini terjadi, kita perlu secara manual menghentikannya dengan menekan tombol “STOP” pada bar icon di RAPTOR.
- Kita harus menghindari terjadinya infinite loop dengan tidak menulis statement yang menghasilkan infinite loop tersebut.
- Caranya adalah dengan menuliskan satu (atau beberapa statement) pada Loop untuk mengubah satu atau lebih variabel dalam statement Decision sehingga pada akhirnya akan menghasilkan keputusan “Yes”
- Dengan demikian, eksekusi proses akan keluar dari loop dan melanjutkan ke statement lain hingga mencapai End.



Latihan Loop (1)

- Loop dapat digunakan mengulangi bagian code tertentu. Contoh penggunaan:
 - Dalam program menghitung konversi Fahrenheit ke Celcius dan sebaliknya, user dapat mengulangi untuk melakukan konversi lagi, tanpa harus menjalankan ulang program
 - Cara: setelah selesai melakukan konversi, dimunculkan pertanyaan apakah user ingin mengulangi perhitungan? Jika jawabannya “YA” maka program akan melakukan konversi lagi. Jika jawabannya “TIDAK”, maka program akan selesai.

- 
- Flowchart?

Latihan Loop (2)

- Loop dapat digunakan untuk menjumlahkan deret matematika

- Selesaikan: $\sum_{i=0}^{10} i$

$$\sum_{i=0}^{10} 2i + 3$$

$$\sum_{i=0}^5 (i + 2)(i + 5)$$

$$\sum_{i=0}^5 \sum_{j=0}^4 i + j$$

Tracing

- To trace: to follow the footprints, track, or traces of.
- Tracing: proses melacak sesuatu
- Dalam konsep pemrograman, tracing berarti proses melacak setiap perubahan nilai dari tiap variable yang digunakan (atau variable tertentu yang ingin diketahui perubahan nilainya)
- Caranya dengan menuliskan nilai variable dan perubahannya pada setiap loop

Latihan tracing

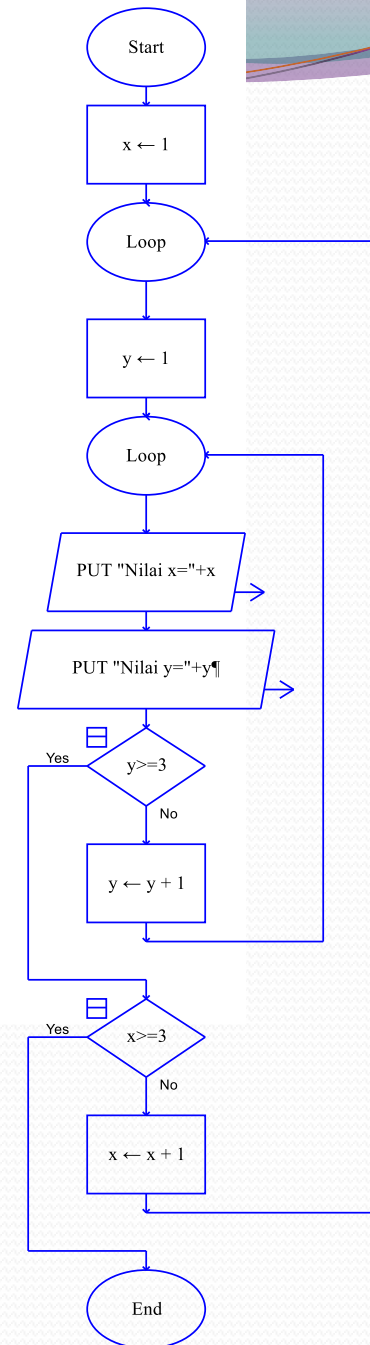
- Lacak perubahan variable pada deret berikut:

$$\sum_{i=0}^{10} 2i + 3$$

Loop Bersarang (*Nested Loop*)

- Loop bersarang adalah loop di dalam loop
- Jika loop biasa adalah melakukan iterasi dengan 1 variabel, maka loop bersarang adalah iterasi dengan 2 atau lebih variabel.

Contoh flowchart loop bersarang dengan 2 variabel: x dan y



Loop Bersarang (*Nested Loop*)

- Contoh penggunaan loop bersarang:
 - Mengisi data ke sebuah tabel
 - Membuat matriks dan operasi terhadap matriks
 - Penjumlahan
 - Pengurangan
 - Perkalian bilangan dengan matriks
 - Perkalian matriks dengan matriks
 - Penjumlahan deret matematika

Latihan

- Buat flowchart untuk menghitung jumlah dari deret berikut:

$$\sum_{i=0}^5 \sum_{j=0}^4 i + j$$

$$\sum_{i=1}^4 \sum_{j=1}^3 \sum_{k=1}^2 i + 2j - k$$

Latihan tracing

- Lacak perubahan variable pada deret berikut:

$$\sum_{i=0}^5 \sum_{j=0}^4 i + j$$

$$\sum_{i=1}^4 \sum_{j=1}^3 \sum_{k=1}^2 i + 2j - k$$