

2.1.3 Pertanyaan

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program

Percobaan 1!

Inisialisasi = `int i = 1;`

Kondisi = `i <= 10;`

Update = `i++`

2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Untuk inisialisasi awal dan nanti nya akan di bandingkan dengan input variable nilai, lalu variable tertinggi / terendah akan di Ganti dengan nilai variable nilai apabila perbandingan nya merupakan true.

Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Nilai tertinggi akan selalu 100 dan nilai terendah akan selalu 0.

3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

```
if (nilai > tertinggi) {  
    tertinggi = nilai;  
}  
if (nilai < terendah) {  
    terendah = nilai;  
}
```

Program tersebut berfungsi Untuk melakukan inisialisasi ulang pada variable tertinggi dan variable terendah. Alur dari program tersebut Adalah melakukan perbandingan variable nilai dengan variable tertinggi, apakah variable nilai lebih besar daripada variable tertinggi? Apabila benar maka akan dilakukan inisialisasi pada nilai variable tertinggi menjadi nilai variable nilai. Lalu melakukan perbandingan variable nilai dengan variable terendah, apakah variable nilai lebih kecil daripada variable terendah? Apabila benar maka akan dilakukan inisialisasi pada nilai variable terendah menjadi nilai variable nilai.

4. Modifikasi kode program sehingga terdapat perhitungan untuk menentukan berapa mahasiswa yang lulus dan yang tidak lulus berdasarkan batas kelulusan (nilai minimal 60).

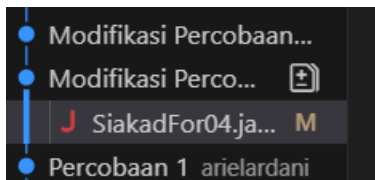
Tampilkan jumlah mahasiswa lulus dan tidak lulus setelah menampilkan nilai tertinggi

dan terendah!

```
int lulus = 0, tidakLulus = 0;
```

```
}  
if (nilai >= 60) {  
    lulus++;  
}  
else {  
    tidakLulus++;  
}
```

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 1”



2.2.3 Pertanyaan

1. Pada potongan kode berikut, tentukan maksud dan kegunaan dari sintaks berikut:

```
if (nilai < 0 || nilai > 100) {  
    System.out.println(x: "Nilai tidak valid. Masukkan lagi nilai yang valid!");  
    continue;  
}
```

a. `nilai < 0 || nilai > 100`

membuat kondisi dimana variable nilai lebih kecil dari 0 atau lebih besar dari 100 adalah True.

b. `continue`

untuk melewati semua program di bawahnya dan langsung Kembali keawal loop.

2. Mengapa sintaks `i++` dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?

Agar algoritmanya sesuai dengan urutan, apabila `i++` di taruh di awal perulangan maka nilai `i` akan menjadi 1 dan pada saat `System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");` output yang keluar pada loop pertama Adalah Masukkan nilai mahasiswa ke-2.

3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?

19 kali.

4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"

```
if (nilai > 80 && nilai <= 100) {  
    System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");  
    System.out.println(x:"Bagus, pertahankan nilainya");  
}
```

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”

2.3.3 Pertanyaan

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?

Satu kali.

2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!

Kondisi dimana input variable namaPelanggan Adalah “batal” dengan mengabaikan besar kecilnya huruf.

3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?

Untuk menjaga kondisi pada struktur perulangan.

4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?

Karena dalam struktur perulangan kondisinya selalu true yang membuat perulangan tersebut terulang tak terbatas atau biasa disebut infinite loop.