

#### **A. Percobaan 1**

Di dalam Sistem Informasi Akademik (SIKAD), dosen mengisi nilai mata kuliah Praktikum Dasar Pemrograman yang ditempuh oleh mahasiswa. Dosen tersebut ingin mencari nilai tertinggi dan terendah Kuis dari 10 mahasiswa di dalam satu kelas. Dosen tersebut harus memasukkan nilai dari setiap siswa, kemudian menentukan dan menampilkan nilai tertinggi dan terendah. Perhatikan flowchart berikut ini: Berdasarkan flowchart tersebut, buat program menggunakan bahasa pemrograman Java. Langkah-langkah Percobaan.

1. Buat repository baru pada akun Github Anda, beri nama daspro-jobsheet7
2. Lakukan cloning repository tersebut menggunakan perintah git clone dari terminal
3. Buka folder repository tersebut menggunakan Visual Studio Code
4. Buat file baru, beri nama SiakadForNoAbsen.java
5. Buatlah struktur dasar program Java yang terdiri dari fungsi main().
6. Tambahkan library Scanner di bagian atas (luar) class
7. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()
8. Deklarasikan variabel nilai, tertinggi, dan terendah bertipe double. Inisialisasi tertinggi dengan 0 dan terendah dengan 100
9. Buat struktur perulangan FOR dengan batas kondisi sesuai jumlah mahasiswa yaitu 10
10. Di dalam perulangan FOR tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Setelah itu, buat dua kondisi pemilihan secara terpisah untuk mengecek nilai tertinggi dan terendah dengan membandingkan nilai masukan dengan variabel tertinggi dan variabel terendah
11. Di luar perulangan FOR, tampilkan nilai tertinggi dan terendah
12. Compile dan run program

```

1  import java.util.Scanner;
2
3  public class SiakadFor13 {
4      Run | Debug
5      public static void main(String[] args){
6          Scanner sc13 = new Scanner(System.in);
7          double nilai, tertinggi = 0, terendah = 100;
8
9          for (int i = 1; i <= 10; i++){
10             System.out.print("Masukan nilai Mahasiswa ke- " + i + ": ");
11             nilai = sc13.nextDouble();
12             if (nilai > tertinggi ) {
13                 tertinggi = nilai;
14             }if(nilai < terendah){
15                 terendah = nilai;
16             }
17
18             System.out.println("Nilai Tertinggi: "+ tertinggi);
19             System.out.println("Nilai Tertinggi: "+ terendah);
20         }
21     }

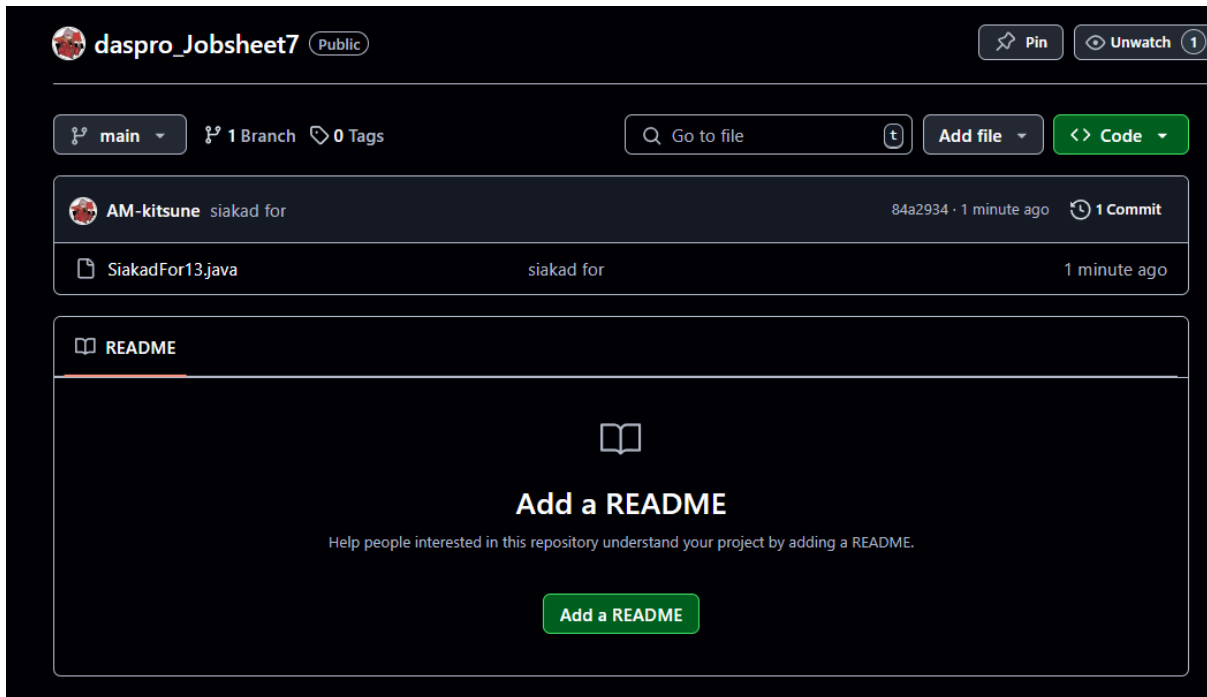
```

```

Masukan nilai Mahasiswa ke- 1: 80
Masukan nilai Mahasiswa ke- 2: 90,2
Masukan nilai Mahasiswa ke- 3: 13
Masukan nilai Mahasiswa ke- 4: 54
Masukan nilai Mahasiswa ke- 5: 67
Masukan nilai Mahasiswa ke- 6: 86
Masukan nilai Mahasiswa ke- 7: 77
Masukan nilai Mahasiswa ke- 8: 90,1
Masukan nilai Mahasiswa ke- 9: 88
Masukan nilai Mahasiswa ke- 10: 78
Nilai Tertinggi: 90.2
Nilai Tertinggi: 13.0
PS D:\java4>

```

13. Commit dan push kode program ke Github



```
ea300779\bin SiakadFor13
Masukan nilai Mahasiswa ke- 1: 76,5
Masukan nilai Mahasiswa ke- 2: 82,3
Masukan nilai Mahasiswa ke- 3: 62,1
Masukan nilai Mahasiswa ke- 4: 88,4
Masukan nilai Mahasiswa ke- 5: 65,9
Masukan nilai Mahasiswa ke- 6: 67,9
Masukan nilai Mahasiswa ke- 7: 90,1
Masukan nilai Mahasiswa ke- 8: 55,3
Masukan nilai Mahasiswa ke- 9: 73,7
Masukan nilai Mahasiswa ke- 10: 78,6
Nilai Tertinggi: 90.1
Nilai Tertinggi: 55.3
PS D:\java4>
```

### Pertanyaan:

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

jawaban:

```
for (int i = 1; i <= 10; i++){
    System.out.print("Masukan nilai Mahasiswa ke- " + i + ": ");
    nilai = sc13.nextDouble();
    if (nilai > tertinggi ) {
        tertinggi = nilai;
    }if(nilai < terendah){
        terendah = nilai;
    }
}
```

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

Jawaban:

Pemilihan nilai awal 0 untuk tertinggi dan 100 untuk terendah memastikan bahwa setiap nilai yang dimasukkan akan dibandingkan dengan nilai ekstrim ini, sehingga nilai tertinggi dan terendah yang sebenarnya dapat ditemukan dengan akurat.

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

Kode ini membandingkan setiap nilai baru dengan nilai tertinggi dan terendah yang sudah ditemukan sebelumnya, lalu memperbarui nilai tertinggi atau terendah jika nilai baru lebih besar atau lebih kecil.

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!

```
1  import java.util.Scanner;
2
3  public class SiakadFor13 {
4      Run main | Debug main | Run | Debug
5      public static void main(String[] args) {
6          Scanner sc13 = new Scanner(System.in);
7          double nilai, tertinggi = 0, terendah = 100;
8          int lulus = 0, tidakLulus = 0;
9          int batasKelulusan = 60;
10
11         for (int i = 1; i <= 10; i++) {
12             System.out.print("Masukan nilai mahasiswa ke-" + i + ": ");
13             nilai = sc13.nextDouble();
14
15             if (nilai > tertinggi) {
16                 tertinggi = nilai;
17             }
18
19             if (nilai < terendah) {
20                 terendah = nilai;
21             }
22
23             if (nilai >= batasKelulusan) {
24                 lulus++;
25             } else {
26                 tidakLulus++;
27             }
28         }
29
30         System.out.println("Nilai tertinggi: " + tertinggi);
31         System.out.println("Nilai terendah: " + terendah);
32         System.out.println("Jumlah mahasiswa yang lulus: " + lulus);
33         System.out.println("Jumlah mahasiswa yang tidak lulus: " + tidakLulus);
34     }
35 }
```

5. Commit dan push kode program ke Github

A. Percobaan 2

Seorang dosen ingin memasukkan nilai beberapa mahasiswa ke dalam SIAKAD untuk ditentukan kategori nilai hurufnya. Program harus meminta dosen untuk memasukkan nilai setiap mahasiswa. Jika dosen memasukkan nilai yang tidak valid (negatif atau lebih dari 100), program harus mengabaikan input tersebut dan meminta dosen untuk melakukan input ulang. Selanjutnya, nilai yang valid dikelompokkan ke dalam kategori huruf A ( $80 < \text{nilai} \leq 100$ ), B+ ( $73 < \text{nilai} \leq 80$ ), B ( $65 < \text{nilai} \leq 73$ ), C+ ( $60 < \text{nilai} \leq 65$ ), C ( $50 < \text{nilai} \leq 60$ ), D ( $39 < \text{nilai} \leq 50$ ), dan E ( $\text{nilai} \leq 39$ ). Berdasarkan studi kasus tersebut, buat program menggunakan bahasa pemrograman Java.

#### Langkah-langkah Percobaan

1. Buat file baru, beri nama SiakadWhileNoAbsen.java
2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().
3. Tambahkan library Scanner di bagian atas (luar) class
4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()
5. Deklarasikan variabel nilai, jml, dan i (untuk perulangan) bertipe integer. Inisialisasi i dengan 0 sebagai nilai awal perulangan
6. Tuliskan kode program untuk menerima input banyaknya mahasiswa yang disimpan ke variabel jml. Dengan demikian, batas perulangan akan dinamis sesuai masukan dari pengguna melalui keyboard.
7. Buat struktur perulangan WHILE dengan batas kondisi sesuai jumlah mahasiswa yaitu 5. Perhatikan simbol yang digunakan adalah `<` karena perulangan variabel i dimulai dari 0, bukan 1
8. Di dalam perulangan WHILE tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Setelah itu, buat kondisi pemilihan IF untuk mengecek valid atau tidaknya nilai yang dimasukkan, dengan syarat nilai harus berada pada rentang 0 hingga 100. Kemudian tambahkan kondisi pemilihan IF-ELSE IF-ELSE untuk menampilkan kategori nilai huruf berdasarkan ketentuan.
9. Compile dan run program

```

1  import java.util.Scanner;
2
3  public class SiakadWhile13 {
4      Run main | Debug main | Run | Debug
5      public static void main(String[] args) {
6          Scanner sc13 = new Scanner(System.in);
7          int nilai, jml, i =0;
8
9          System.out.print(s:"Masukan jumlah mahasiswa: ");
10         jml = sc13.nextInt();
11
12         while (i < jml) {
13             System.out.print("Masukan nilai mahasiswa ke-" + (i + 1) + ": ");
14             nilai = sc13.nextInt();
15
16             if (nilai < 0 || nilai > 100) {
17                 System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid:");
18                 continue;
19             }
20
21             if (nilai > 80 && nilai <= 100) {
22                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");
23             }else if (nilai > 73 && nilai <= 80) {
24                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
25             }else if (nilai > 65 && nilai <= 73) {
26                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
27             }else if (nilai > 60 && nilai <= 65) {
28                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
29             }else if (nilai > 50 && nilai <= 60) {
30                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
31             }else if (nilai > 39 && nilai <= 50) {
32                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
33             }else {
34                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
35             }
36             i++;
37         }
38     }
39 }
40

```

```

PS D:\java3> & 'C:\Program Files\Java\jdk-22\bin\java.exe'
a\Roaming\Code\User\workspaceStorage\ecc580737f3f932c50195
Masukan jumlah mahasiswa: 6
Masukan nilai mahasiswa ke-1: 92
Nilai mahasiswa ke-1 adalah A
Masukan nilai mahasiswa ke-2: 99
Nilai mahasiswa ke-2 adalah A
Masukan nilai mahasiswa ke-3: 80
Nilai mahasiswa ke-3 adalah B+
Masukan nilai mahasiswa ke-4: 43
Nilai mahasiswa ke-4 adalah D
Masukan nilai mahasiswa ke-5: 12
Nilai mahasiswa ke-5 adalah E
Masukan nilai mahasiswa ke-6: 67
Nilai mahasiswa ke-6 adalah B
PS D:\java3>

```

10. Commit dan push kode program ke Github

1. Pada potongan kode berikut, tentukan maksud dan kegunaan dari sintaks berikut:
  - a. `nilai < 0 || nilai > 100`
  - b. `continue`

jawaban:

A. Bagian `nilai < 0 || nilai > 100` berfungsi untuk memvalidasi apakah nilai yang dimasukkan oleh pengguna berada dalam rentang yang diizinkan.

B. `continue`: Jika nilai tidak valid, perintah `continue` akan langsung melompat ke iterasi berikutnya, sehingga program tidak akan memproses nilai yang salah.

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

jawaban:

Menempatkan `i++` di akhir perulangan `while` memastikan bahwa kondisi perulangan diperiksa terlebih dahulu sebelum nilai `i` ditingkatkan, sehingga perulangan berjalan dengan benar dan efisien.

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

jawaban:

perulangan akan di ulang sebanyak 19 kali

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

```

1 import java.util.Scanner;
2
3 public class SiakadWhile13 {
4     Run main | Debug main | Run | Debug
5     public static void main(String[] args) {
6         Scanner sc13 = new Scanner(System.in);
7         int nilai, jml, i = 0;
8
9         System.out.print(s:"Masukan jumlah mahasiswa: ");
10        jml = sc13.nextInt();
11
12        while (i < jml) {
13            System.out.print("Masukan nilai mahasiswa ke-" + (i + 1) + ": ");
14            nilai = sc13.nextInt();
15
16            if (nilai < 0 || nilai > 100) {
17                System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid:");
18                continue;
19            }
20
21            if (nilai > 80 && nilai <= 100) {
22                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");
23                System.out.println(x:"Bagus, pertahankan nilainya!");
24            } else if (nilai > 73 && nilai <= 80) {
25                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
26            } else if (nilai > 65 && nilai <= 73) {
27                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
28            } else if (nilai > 60 && nilai <= 65) {
29                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
30            } else if (nilai > 50 && nilai <= 60) {
31                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
32            } else if (nilai > 39 && nilai <= 50) {
33                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
34            } else {
35                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
36            }
37            i++;
38        }
39    }
40 }

```

```

Masukan jumlah mahasiswa: 5
Masukan nilai mahasiswa ke-1: 99
Nilai mahasiswa ke-1 adalah A

```

## 5. Commit dan push kode program ke Github

### A. Percobaan 3

Di sebuah kafe, kasir ingin memproses transaksi beberapa pelanggan. Pelanggan dapat membeli lebih dari satu item (kopi dengan harga Rp 12.000, teh dengan harga Rp 7.000, dan roti dengan harga Rp 20.000), dan kasir akan terus memasukkan jumlah pembelian untuk setiap pelanggan. Jika ada pelanggan yang memutuskan untuk membatalkan transaksi (dengan memasukkan "batal"), maka kasir akan menghentikan input transaksi dan program berhenti. Berdasarkan studi kasus tersebut, buat program menggunakan bahasa pemrograman Java. Langkah-langkah Percobaan

1. Buat file baru, beri nama KafeDoWhileNoAbsen.java
2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().
3. Tambahkan library Scanner di bagian atas (luar) class
4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()
5. Deklarasikan variabel kopi, teh, dan roti bertipe integer untuk menampung banyaknya item yang dibeli pelanggan, serta namaPelanggan bertipe String.



Deklarasi dan inisialisasi hargaKopi dengan 12000, hargaTeh dengan 7000, hargaRoti dengan 20000.

6. Buat struktur perulangan DO-WHILE dengan kondisi true

7. Di dalam perulangan DO-WHILE tersebut, tambahkan perintah untuk memasukkan namaPelanggan. Kemudian tambahkan kondisi IF untuk mengecek isi variabel namaPelanggan. Selanjutnya, tambahkan perintah untuk memasukkan banyaknya item yang dibeli pelanggan untuk setiap menu, apabila masukan nama pelanggan bukan "batal". Hitung total harga pembelian dan tampilkan hasilnya  
Keterangan: sc.nextLine(); setelah print totalHarga merupakan sintaks untuk membersihkan newline dari buffer

8. Compile dan run program

```
1  import java.util.Scanner;
2
3  public class KafeDowWhile13 {
    Run main | Debug main | Run | Debug
4      public static void main(String[] args) {
5          Scanner sc13 = new Scanner(System.in);
6          int kopi, teh, roti, totalHarga;
7          String namaPelanggan;
8          int hargaTeh = 7000, hargaKopi = 12000, hargaRoti = 20000;
9
10         do {
11             System.out.print(s:"Masukan nama pelanggan (ketik 'batal' untuk keluar): ");
12             namaPelanggan = sc13.nextLine();
13
14             if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
15                 System.out.println(x:"Transaksi dibatalkan");
16                 break;
17             }
18             System.out.print(s:"Jumlah kopi: ");
19             kopi = sc13.nextInt();
20             System.out.print(s:"Jumlah teh: ");
21             teh = sc13.nextInt();
22             System.out.print(s:"Jumlah roti: ");
23             roti = sc13.nextInt();
24
25             totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
26             System.out.println("Total yang harus dibayar: Rp " + totalHarga);
27             sc13.nextLine();
28
29         } while (true);
30
31         System.out.println(x:"Semua transaksi selesai.");
32     }
33 }
34
```

```

Page\ec2500737f5f952c501955fa9007dc19\rednat.java\jul_ws\java3_ea500778\bin - Karek
Masukan nama pelanggan (ketik 'batal' untuk keluar): Khesa (Kucing)
Jumlah kopi: 3
Jumlah teh: 0
Jumlah roti: 1
Total yang harus dibayar: Rp 56000
Masukan nama pelanggan (ketik 'batal' untuk keluar): Kuda
Jumlah kopi: 1
Jumlah teh: 4
Jumlah roti: 2
Total yang harus dibayar: Rp 80000
Masukan nama pelanggan (ketik 'batal' untuk keluar): BATAL
Transaksi dibatalkan
Semua transaksi selesai.
PS D:\java3>

```

9. Commit dan push kode program ke Github

C. Pertanyaan :

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

Jawaban: 1 kali

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

Jawaban: Ketika adanya kodingan “break”

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

Jawaban: Agar Loop menjadi infinite loop.

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

Jawaban: Karena perulangan do-while tidak perlu kita siapkan nilai awal atau mengubah nilainya selama perulangan, maka perulangan ini akan terus berjalan selama kondisi yang kita tetapkan benar, bahkan jika kita tidak sengaja lupa untuk mengaturnya

A. TUGAS

1. Seorang pengelola bioskop ingin membuat program untuk menghitung total penjualan tiket dalam satu hari. Tiket dijual dengan harga Rp 50.000 per tiket. Program harus menghitung total tiket yang terjual dan total harga penjualan tiket selama satu hari dengan ketentuan sebagai berikut: Jika pelanggan membeli lebih dari 4 tiket, pelanggan mendapatkan diskon 10%. Jika pelanggan membeli lebih dari 10 tiket, pelanggan mendapatkan diskon 15%. Jika input jumlah tiket tidak valid (negatif), program akan mengabaikan input tersebut dan meminta input ulang. Catatan: Perulangan dapat menggunakan for, while, atau do-while. Penambahan break atau continue jika diperlukan

```

1  import java.util.Scanner;
2
3  public class Bioskop13 {
4      public static void main(String[] args) {
5          Scanner sc13 = new Scanner(System.in);
6          int hargaTiket = 50000;
7          int totalTiketTerjual = 0;
8          double totalPenjualan = 0.0;
9
10         while (true) {
11             System.out.print(s:"Masukkan jumlah tiket yang dibeli (ketik '0' untuk Membatalkan : ");
12             int totalTiket = sc13.nextInt();
13             double hargaTotal = totalTiket * hargaTiket;
14             double diskon = 0.0;
15             double hargaTotalDisc = hargaTotal * (1 - diskon);
16
17             if (totalTiket == 0) {
18                 break;
19             }
20             if (totalTiket < 0) {
21                 System.out.println(x:"Jumlah tiket tidak valid. Silakan masukkan jumlah yang benar.");
22                 continue;
23             }
24
25             if (totalTiket > 10) {
26                 diskon = 0.15;
27             } else if (totalTiket > 4) {
28                 diskon = 0.10;
29             }
30
31             totalTiketTerjual += totalTiket;
32             totalPenjualan += hargaTotalDisc;
33
34             System.out.println("Harga total untuk " + totalTiket + " tiket adalah: Rp " + hargaTotalDisc);
35
36             System.out.println(x:"=====");
37             System.out.println("Total tiket Yang terjual: " + totalTiketTerjual);
38             System.out.println("Total penjualan tiket: Rp " + totalPenjualan);
39         }
40     }

```

2. Sebuah tempat parkir ingin membuat program untuk menghitung total pembayaran parkir dari beberapa kendaraan. Tarif parkir adalah Rp 3.000 per jam untuk mobil dan Rp 2.000 per jam untuk motor. Namun, jika durasi parkir lebih dari 5 jam, diberikan tarif tetap sebesar Rp 12.500 untuk semua kendaraan. Program akan terus meminta masukan selama input bukan 0. Implementasikan flowchart tersebut ke dalam bentuk kode program Java!

```
import java.util.Scanner;

public class Parkir13 {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        Scanner sc13 = new Scanner(System.in);
        int jenis, durasi;

        int total = 0;
        while (true) {
            System.out.print(s:"Masukan jenis kendaraan (1 Moil, 2 Motor, 0 Keluar): ");
            jenis = sc13.nextInt();
            if (jenis == 0) {
                System.out.println(x:"Nambah gih");
                break;
            } else if (jenis == 1 || jenis == 2) {
                System.out.print(s:"Masukan durasi parkir dalam jam: ");
                durasi = sc13.nextInt();
                if (durasi > 5) {
                    total = 12500;
                } else {
                    if (jenis == 1) {
                        total = durasi * 3000;
                    } else if (jenis == 2) {
                        total = durasi * 2000;
                    }
                }
                System.out.println("Total yang harus dibayar: Rp " + total);
            } else {
                System.out.println(x:"Jenis kendaraan tidak valid. Silakan masukkan 1 untuk Mobil atau 2 untuk Motor.");
            }
        }
    }
}
```