

LAPORAN PRATIKUM
ALGORITMA PEMOGRAMAN

PEKAN KE-4

disusun Oleh:

Aisyah Najmiatul Fauziah

NIM 2511533030

Dosen Pengampu: DR. WAHYUDI, s.t, m.t

Asisten Pratikum: Rahmat DRO



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

2025

KATA PENGANTAR

Laporan pratikum ini disusun sebagai bentuk pertanggung jawaban atas pelaksanaan kegiatan pratikum mata kuliah Algoritma Pemograman yang dilaksanakan pada pekan ke-4. Melalui laporan ini penulis dapat lebih memahami materi pratikum dan dengan penulisan laporan i ni dapat melatih ketelitian, keteraturan, serta kemampuan menulis sesuai kaidah akademik pada tingkat dasar. Dengan demikian, laporan pratikum yang dihasilkan dapat berfungsi sebagai media pembelajaran, dokumentasi kegiatan, sekaligus sarana untuk melatih keterampilan menulis ilmiah yang akan berfmanfaaat dalam pembelajaran selanjutnya.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, penulis sangat terbuka terhadap segala bentuk kritik dan saran yang bersifat membangun, demi perbaikan kualitas laporan maupun pemahan penulis dilaporan berikutnya.

Padang, 2025

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan Praktikum	1
1.3 Manfaat Praktikum	1
BAB II.....	2
PEMBAHASAN	2
2.1 Praktikum “Ifelse1”	2
2.2 Praktikum “Latifl”	2
2.3 Praktikum “MultIf”	2
2.4 Pratikum “Nama Bulan”	2
2.5 Pratikum “Nilai Java”	2
BAB III.....	3
KESIMPULAN.....	3
3.1 Kesimpulan.....	3
DAFTAR PUSTAKA.....	4

BAB I

PENDAHULUAN

1.1 Latar belakang

Dalam pemograman, salah satu kemampuan dasar yang harus dikuasai oleh setiap mahasiswa adalah memahami cara kerja **struktur percabangan**. Struktur percabangan digunakan untuk membuat program yang dapat mengambil keputusan berdasarkan kondisi tertentu. Dengan percabangan, program dapat menjalankan perintah yang berbeda tergantung pada nilai atau masukan yang berbeda tergantung pada nilai atau masukan yang diberikan pengguna.

Pada pratikum ini, dilakukan beberapa percobaan menggunakan bahasa pemograman **Java** untuk memahami berbagai bentuk percabangan, yaitu **if**, **if-else**, **if-else-if**, **nasted if**, dan **switch case**.

- Pada **program if** dan **if-else**, percabangan digunakan untuk menentukan apakah pengguna lulus atau tidak berdasarkan IPK yang dimasukkan.
- Program **nasted if** digunakan untuk mengecek dua kondisi sekaligus, yaitu umur dan kepemilikan SIM, untuk menentukan apakah seseorang boleh membawa kendaraan bermotor.
- Program **switch case** digunakan dalam menampilkan nama bulan berdasarkan angka yang diinputkan oleh pengguna, sebagai contoh penerapan percabangan yang lebih terstruktur untuk banyak kondisi pilihan

Melalui pratikum ini, mahasiswa dapat memahami cara kerja setiap jenis percabangan serta perbedaan penggunaannya dalam menyelesaikan masalah yang berbeda. Penguasaan struktur percabangan sangat penting sebagai dasar dalam membangun logika program yang lebih kompleks, sehingga program dapat berjalan sesuai dengan kondisi yang diinginkan

1.2 TUJUAN PRATIKUM

1. Memahami dan menerapkan struktur percabangan dalam bahasa pemograman Java.
2. Mengetahui perbedaan penggunaan antara **if**, **if-else**, **if-else-if**, **nasted if** dan **switch case**.
3. Mampu membuat program sederhana yang dapat mengambil keputusan berdasarkan kondisi tertentu.
4. Melatih kemampuan logika mahasiswa dalam menyelesaikan permasalahan pemograman yang membutuhkan pengambilan keputusan.
5. Meningkatkan keterampilan dalam menggunakan Eclipse IDE sebagai alat bantu pengembangan progrfam Java

1.3 MANFAAT PRATIKUM

1. Mahasiswa dapat memahami konsep dasar pengambilan keputusan dalam pemograman menggunakan struktur percabangan.
2. Mahasiswa mampu menerapkan berbagai bentuk percabangan (if, if-else, if-else-if, nasted if dan swtich case) dalam menyelesaikan masalah nyata.
3. Mahasiswa dapat mengembangkan logika berpikir yang sistematis dan terstruktur dalam proses penulisan kode.
4. Pratikum ini melatih mahasiswa untuk mengembangkan program yang interaktif dan dinamis berdasarakan input pengguna

BAB II

PEMBAHASAN

2.1 PRATIKUM “IFELSE1”

```
1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class Ifelse1 {
6     public static void main(String[] args) {
7         double IPK;
8         Scanner input=new Scanner(System.in);
9         System.out.print("Input IPK anda = ");
10        IPK=input.nextDouble();
11        input.close();
12        if (IPK>2.75) {
13            System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
14        } else {
15            System.out.println("Anda Tidak Lulus");
16        }
17    }
18 }
19
20
21 }
```

Gambar 2.1 kode pemrograman if-else

Struktur if digunakan untuk mengevaluasi satu kondisi. Jika kondisi bernilai benar (true), maka blok perintah di dalam if akan dijalankan.

Adapun penjelasan tentang langkah-langkah program tersebut adalah sebagai berikut:

1. Buka IDE Eclipse dan buat package **pekan4**.
2. Buat class baru dengan nama **Ifelse1**.
3. Import library **java.util.scanner** untuk menerima input dari pengguna
4. Deklarasi variabel **IPK** bertipe **double**.
5. Buat objek **scanner** untuk membaca input dari keyboard.
6. Tampilkan teks “Input IPK anda =” dan simpan hasil input ke variabel **IPK**.
7. Gunakan pernyataan **if-else** untuk memeriksa apakah nilai IPK lebih dari 2.75.
 - Jika ya, tampilkan teks “Anda Lulus sangat Memuaskan dengan IPK [nilai IPK]”.

- Jika tidak, tampilkan teks “Anda Tidak Lulus”

8. Simpan dan jalankan program untuk melihat hasilnya

Dari langkah-langkah diatas kita akan mendapatkan output seperti gambar dibawah ini

```
<terminated> Ifelse1 [Java Application] C:\Users\ASUS\p2\p
Input IPK anda = 3
Anda Lulus Sangat Memuaskan dengan IPK 3.0
```

Gambar 2.1 ouput dari program ifelse1

2.2 PRATIKUM “LATIF1”

```
1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class latif1 {
6
7     public static void main(String[] args) {
8         double IPK;
9         Scanner input=new Scanner(System.in);
10        System.out.print("Input IPK anda = ");
11        IPK=input.nextDouble();
12        input.close();
13        if (IPK>2.75) {
14            System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
15        }
16    }
17 }
18 }
19 }
```

Gambar 2.2 kode program latif1

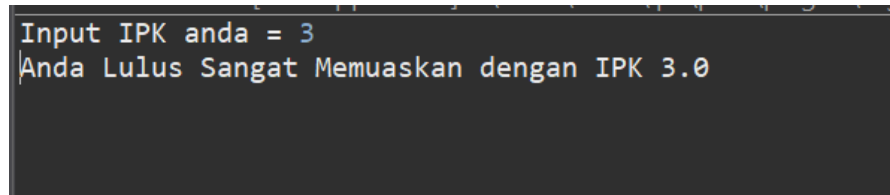
Struktur ini digunakan ketika terdapat dua kemungkinan kondisi, yaitu benar dan salah. Jika kondisi salah, maka program akan mengeksekusi blok **else**.

Adapun penjelasan tentang langkah-langkah program tersebut adalah sebagai berikut:

1. Buat class **latif1** dalam package **pekan4**.
2. Import **java.util scanner**.
3. Deklarasikan variabel **IPK**.
4. Gunakan **Scanner** untukm membaca input IPK dari user.
5. Gunakan pernyataan if tunggal untuk mengecek apakah $IPK > 2.75$.

6. Jika kondisi benar, tampilkan teks “Anda Lulus Sangat Memuaskan dengan IPK [nilai ipk]”.
7. Jalankan program dan masukkan beberapa nilai IPK untuk menguji

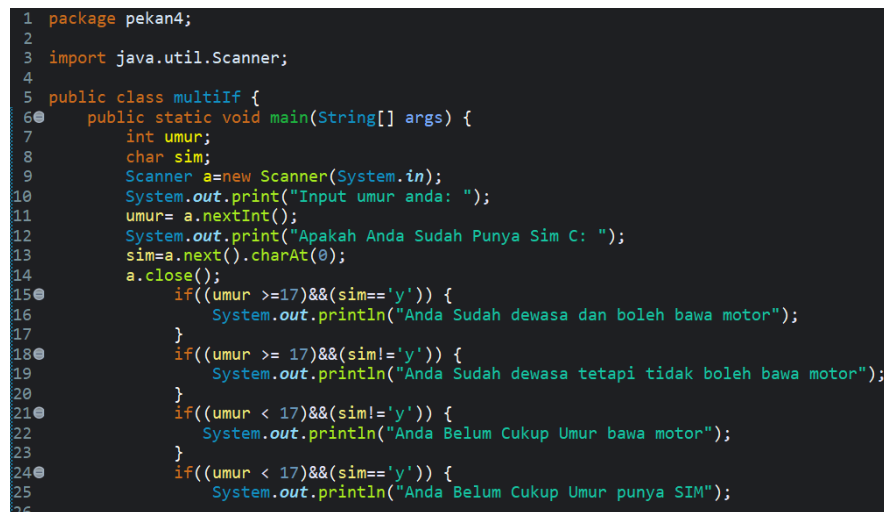
Dari langkah-langkah diatas kita akan mendapatkan output seperti pada gambar dibawah ini:



```
Input IPK anda = 3
Anda Lulus Sangat Memuaskan dengan IPK 3.0
```

Gambar 2.2 output dari program Latif1

2.3 PRATIKUM MULTIF



```
1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class multiIf {
6     public static void main(String[] args) {
7         int umur;
8         char sim;
9         Scanner a=new Scanner(System.in);
10        System.out.print("Input umur anda: ");
11        umur= a.nextInt();
12        System.out.print("Apakah Anda Sudah Punya Sim C: ");
13        sim=a.next().charAt(0);
14        a.close();
15        if((umur >=17)&&(sim=='y')) {
16            System.out.println("Anda Sudah dewasa dan boleh bawa motor");
17        }
18        if((umur >= 17)&&(sim!='y')) {
19            System.out.println("Anda Sudah dewasa tetapi tidak boleh bawa motor");
20        }
21        if((umur < 17)&&(sim=='y')) {
22            System.out.println("Anda Belum Cukup Umur bawa motor");
23        }
24        if((umur < 17)&&(sim!='y')) {
25            System.out.println("Anda Belum Cukup Umur punya SIM");
26        }
27    }
28 }
```

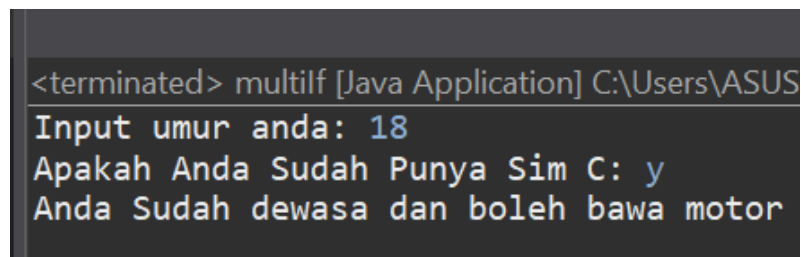
Gambar 2.3 kode program MultiIf

Digunakan untuk mengevaluasi lebih dari dua kondisi. Struktur ini cocok digunakan jika terdapat beberapa kemungkinan yang eksklusif

Adapun penjelasan tentang langkah-langkah program tersebut adalah sebagai berikut:

1. Buat class multif pada package pekan4.
2. Import java.util.Scanner.
3. Deklarasikan variabel umur (int) dan sim (char).
4. Gunakan Scanner untuk membaca umur pengguna.
5. Tanyakan apakah pengguna sudah memiliki SIM C, lalu baca “(y/t).
6. Gunakan beberapa pernyataan if untuk mengecek kombinasi umur dan kepemilikan SIM:
 - Jika umur ≥ 17 dan sim = y, tampilkan “Anda Sudah Dewasa dan Boleh Bawa Motor”.
 - Jika umur ≥ 17 dan sim \neq y, tampilkan “Anda sudah Dewasa Tetapi tidak Boleh Bawa Motor”.
 - Jika umur < 17 dan sim = y, tampilkan “Anda belum cukup umur bawa motor”.
 - Jika umur < 17 dan sim \neq y, tampilkan “Anda belum cukup umur punya Sim”.
7. Jalankan program dan coba berbagai kombinasi untuk menguji semua kondisi.

Dari langkah-langkah diatas kita akan mendapatkan output seperti gambar dibawah ini:



```
<terminated> multif [Java Application] C:\Users\ASUS
Input umur anda: 18
Apakah Anda Sudah Punya Sim C: y
Anda Sudah dewasa dan boleh bawa motor
```

Gambar 2.3 output dari program multiIf

2.4 PRATIKUM “NAMA BULAN”

```
2 import java.util.Scanner;
3 public class NamaBulan {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6         System.out.print("Masukkan angka bulan (1 - 12): ");
7         int bulan = scanner.nextInt();
8         switch (bulan) {
9             case 1:
10                System.out.println("Januari");
11                break;
12             case 2:
13                System.out.println("Februari");
14             case 3:
15                System.out.println("Maret");
16             case 4:
17                System.out.println("April");
18             case 5:
19                System.out.println("Mei");
20             case 6:
21                System.out.println("Juni");
22             case 7:
23                System.out.println("Juli");
24             case 8:
25                System.out.println("Agustus");
26             case 9:
27                System.out.println("September");
28             case 10:
29                System.out.println("Oktober");
30             case 11:
31                System.out.println("November");
32             case 12:
33                System.out.println("Desember");
34                break;
35             default:
36                System.out.println("Angka tidak valid");
37         }
38     }
39 }
```

Gambar 2.4 kode program nama bulan

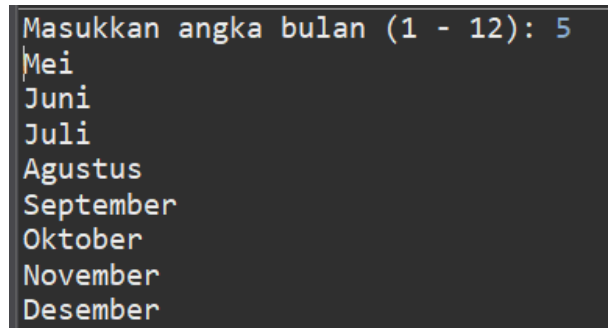
Struktur switch digunakan untuk memilih satu blok perintah dari beberapa pilihan berdasarkan nilai variabel tertentu. Struktur ini lebih rapi untuk banyak kondisi dibandingh if-self-if.

Adapun penjelasan dari langkah langkah program tersebut adalah sebagai berikut:

1. Buat class NamaBulan dalam package pekan4.
2. Import java.utill.scanner,
3. Buat variabel bulan (int) untuk menyimpan angka bulan.
4. Gunakan scanner untuk meminta input angka bulan (1-12).

5. Gunakan struktur switch-case untuk mencetak nama bulan sesuai angka yang dimasukkan :
 - Case 1 “Januari”, case 2 “Februari”, dan seterusnya hingga case 12 “Desember”.
6. Gunakan default untuk menangani input yang tidak valid.
7. Jalankan program dan masukkan berbagai angka untuk memastikan semua case berjalan dengan benar.

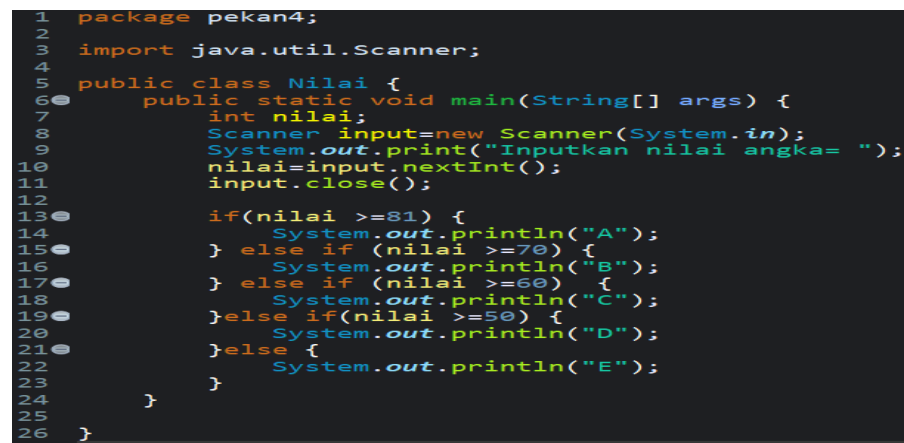
Dari langkah-langkah diatas kita akan mendapatkan output seperti gambar dibawah ini:



```
Masukkan angka bulan (1 - 12): 5
Mei
Juni
Juli
Agustus
September
Oktober
November
Desember
```

Gambar 2.4 output dari program Nama Bulan

2.5 PRATIKUM “NILAI JAVA”



```
1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class Nilai {
6     public static void main(String[] args) {
7         int nilai;
8         Scanner input=new Scanner(System.in);
9         System.out.print("Inputkan nilai angka= ");
10        nilai=input.nextInt();
11        input.close();
12
13        if(nilai >=81) {
14            System.out.println("A");
15        } else if (nilai >=70) {
16            System.out.println("B");
17        } else if (nilai >=60) {
18            System.out.println("C");
19        } else if (nilai >=50) {
20            System.out.println("D");
21        } else {
22            System.out.println("E");
23        }
24    }
25 }
26 }
```

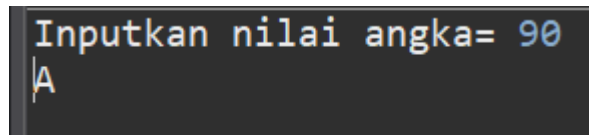
Gambar 2.5 kode program nilai java

Berfungsi untuk menerima input nilai angka dari pengguna serta menentukan nilai huruf (A-E) berdasarkan rentang nilai yang telah ditentukan menggunakan struktur if-else if-else.

Adapun penjelasan dari langkah-langkah program tersebut adalah sebagai berikut:

1. Buat class NilaiJava pada package pekan4.
2. Centang public static void main(String[] args).
3. Import library scanner, scanner digunakan untuk membaca input dari keyboard.
4. Deklarasi variabel dan scanner.
5. Input data dari pengguna, program meminta pengguna memasukkan nilai.
6. Program mengecek kondisi secara berurutan, begitu salah satu kondisi benar hasil dicetak dan blok lainnya dilewati
7. Jalankan program

Dari langkah-langkah diatas diperoleh output sebagai berikut:



```
Inputkan nilai angka= 90
A
```

Gambar 2.5 output dari program Nilai Java

BAB III

KESIMPULAN

3.1 KESIMPULAN

Berdasarkan hasil pratikum, dapat disimpulkan bahwa:

1. Struktur percabangan merupakan komponen penting dalam pengendalian alur program.
2. Pemilihan jenis percabangan harus disesuaikan dengan jumlah kondisi yang akan diperiksa.
3. Penggunaan if, if-else, if-else-if dan switch case dapat mempermudah proses pengambilan keputusan secara otomatis.
4. Mahasiswa telah berhasil memahami dan mengimplementasikan struktur percabangan dalam berbagai contoh kasus menggunakan bahasa pemograman java.
5. Dengan penguasaan percabangan, program dapat dibuat lebih fleksibel, responsif dan mampu menangani berbagai kondisi input.

DAFTAR PUSTAKA

- [1] Oracle Corporation. (2023). The Java™ Tutorials: Learning the Java Language. Diakses dari <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/>
- [2] Horstmann, C. S. (2020). Core Java Volume I: Fundamentals (11th ed.). Prentice Hall.
- [3] Deitel, H. M., & Deitel, P. J. (2021). Java How to Program (11th ed.). Pearson Education.