

LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGAMAN STRUKTUR CONTROL PERCABANGAN

Disusun Oleh:

IHSANUL ZAKY EL-MUHAMMADY

NIM:2511533001

DOSEN PENGAMPU: Dr.WAHYUDI S.T , M.T

ASISTEN LABORATORIUM: AUFAN TAUFIQUR RAHMAN



**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
PADANG, 2025**

KATA PENGANTAR

Dengan penuh rasa syukur, saya mengucapkan terima kasih kepada Tuhan Yang Maha Esa, yang telah memberikan kemudahan dan kesempatan sehingga laporan praktikum ini dapat selesai dengan baik.

Saya juga ingin menyampaikan penghargaan yang setinggi-tingginya kepada Dr. Wahyudi, S.T., M.T. selaku dosen pengampu Mata Kuliah Algoritma dan Pemrograman, yang telah memberikan bimbingan dan motivasi selama perkuliahan. Ucapan terima kasih juga saya sampaikan kepada Uda Aufan Tafiqurrahman, asisten laboratorium, yang dengan sabar dan penuh perhatian telah membimbing kami dalam praktikum ini.

Laporan ini disusun sebagai salah satu tugas dalam Mata Kuliah Pemrograman Dasar dengan tema “Struktur Control Percabangan”.

Saya menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, segala masukan dan saran yang membangun sangat saya harapkan untuk perbaikan di masa mendatang.

Padang , 30 september 2025

Penulis

DAFTAR ISI

KATA PENGANTAR.....	
DAFTAR ISI	
BAB I PENDAHULUAN	
1.1 Latar Belakang	
1.2 Tujuan.....	
1.3 Manfaat.....	
BAB II PEMBAHASAN.....	
2.1 Struktur kontrol percabangan.....	
2.2 Langkah pengerjaan	
BAB III KESIMPULAN.....	
3.1 Kesimpulan	
3.2 Saran.....	
DAFTAR PUSTAKA	

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada praktikum kali ini, kita belajar tentang bagaimana mengatur alur jalannya program menggunakan percabangan di Java. Percabangan penting karena memungkinkan program mengambil keputusan berdasarkan kondisi tertentu. Misalnya, kita bisa menyuruh program melakukan sesuatu kalau suatu syarat terpenuhi, atau melakukan hal lain kalau tidak. Struktur yang digunakan untuk ini antara lain *if*, *if-else*, dan *switch*. Dengan memahami cara kerja percabangan, kita bisa membuat program yang lebih fleksibel dan bisa menyesuaikan diri dengan berbagai situasi.

1.2 Tujuan

1. Memahami cara kerja serta penggunaan struktur percabangan seperti *if*, *if-else*, dan *switch* dalam program.
2. Mengevaluasi bagaimana ekspresi Boolean memengaruhi jalannya program.
3. Mengasah kemampuan menyusun logika program melalui penerapan struktur kontrol untuk proses pengambilan keputusan.
4. Melatih keterampilan dalam menemukan dan memperbaiki kesalahan (debugging), serta memastikan hasil dari percabangan program sudah sesuai harapan.

1.3 Manfaat

1. Memahami dan menerapkan penggunaan percabangan seperti *if* dan *switch* dalam mengatur alur program.
2. Meningkatkan kemampuan dalam membuat keputusan di dalam kode program.
3. Mengasah logika pemrograman saat menangani kondisi yang melibatkan nilai benar atau salah.
4. Melatih kemampuan menganalisis jalannya program dan menemukan serta memperbaiki kesalahan (debugging).

BAB II

PEMBAHASAN

2.1 Struktur Control Percabangan

Struktur kontrol percabangan dalam Java adalah fitur dalam pemrograman yang memungkinkan program untuk mengambil keputusan dan menjalankan bagian kode yang berbeda berdasarkan kondisi atau syarat tertentu.

1. Struktur if-else

Ini adalah bentuk percabangan paling dasar yang dipakai untuk menjalankan kode tertentu berdasarkan kondisi yang dicek.

- **if:** Blok kode di dalam if akan dijalankan kalau kondisinya benar (true).
- **else:** Kalau kondisi if salah (false), maka blok kode di dalam else yang akan dieksekusi.
- **if-else if:** Struktur ini memperluas if-else dengan memungkinkan kita memeriksa beberapa kondisi secara berurutan, sehingga program bisa menjalankan blok kode yang sesuai dengan kondisi yang terpenuhi.

2. Struktur switch-case

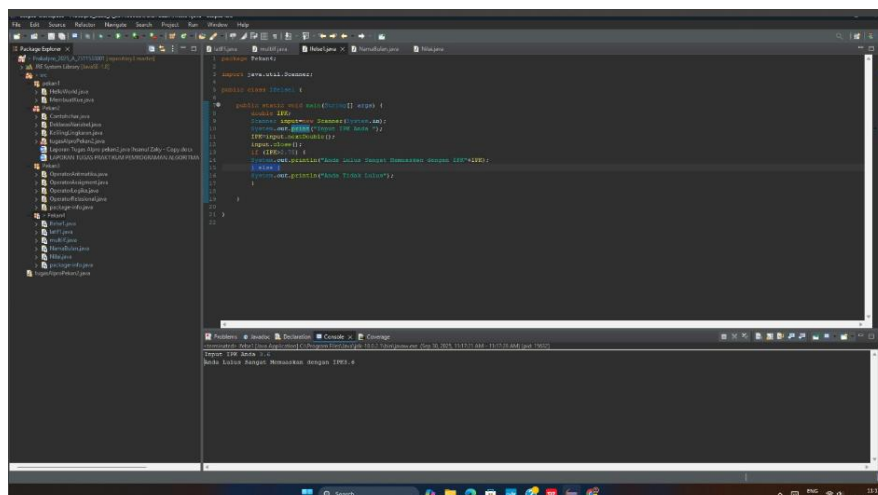
Struktur ini sangat berguna untuk menangani banyak pilihan atau kondisi berbeda secara efisien, terutama ketika ada beberapa nilai yang mungkin dimiliki oleh sebuah variabel.

- **switch:** Memilih satu dari beberapa kemungkinan kasus berdasarkan nilai variabel yang diperiksa.
- **case:** Menentukan kondisi spesifik yang akan dijalankan jika nilai variabel cocok dengan nilai yang ditetapkan dalam kasus tersebut.
- **break:** Berfungsi untuk menghentikan eksekusi setelah kasus yang sesuai dijalankan, agar program tidak terus mengeksekusi kasus berikutnya.
- **default:** Merupakan blok kode yang dijalankan jika tidak ada nilai variabel yang cocok dengan kasus-kasus yang ada. Ini seperti opsi cadangan.

2.3 Langkah Langkah pengerjaan

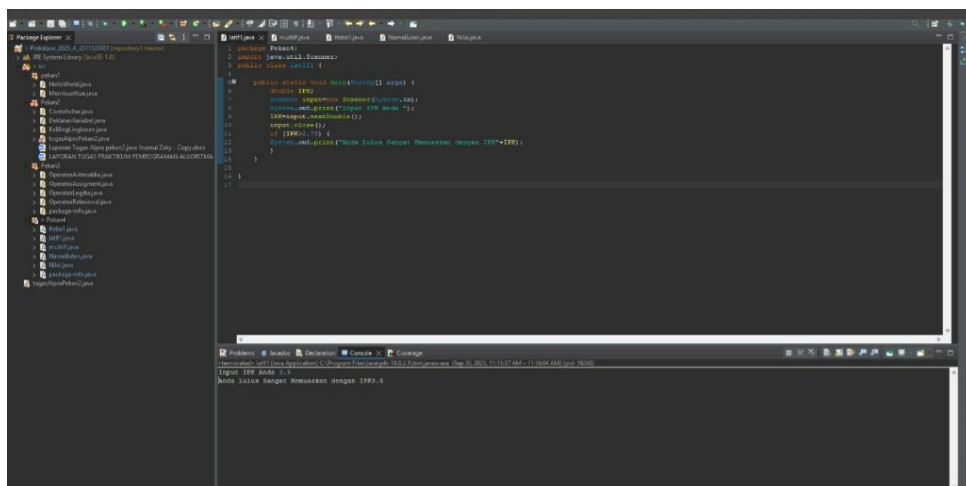
1. Struktur If-Else dengan Input IPK

- Buat sebuah file Java bernama Ifelse1 di dalam package pekan4.
- Import library java.util.Scanner supaya program bisa menerima input dari pengguna.
- Deklarasikan kelas public Ifelse1 yang berisi metode utama public static void main(String[] args).
- Di dalam metode main, buat variabel IPK dengan tipe data double untuk menyimpan nilai Indeks Prestasi Kumulatif yang dimasukkan oleh pengguna.
- Buat objek Scanner bernama input untuk membaca input dari keyboard.
- Tampilkan pesan "Input IPK Anda=" agar pengguna mengetahui bahwa program menunggu nilai IPK.
- Baca nilai yang dimasukkan menggunakan input.nextDouble() dan simpan ke variabel IPK.
- Setelah input selesai, tutup objek Scanner dengan input.close() untuk membebaskan sumber daya sistem.
- Gunakan struktur percabangan if-else untuk menentukan pesan yang akan ditampilkan:
 - Jika $IPK > 2.75$, tampilkan pesan:
"Anda Lulus Sangat Memuaskan dengan IPK [nilai IPK]"
 - Jika $IPK \leq 2.75$, tampilkan pesan:
"Anda Tidak Lulus"
- Jalankan program dan masukkan nilai IPK saat diminta, kemudian amati output yang muncul sesuai dengan kondisi nilai IPK tersebut.



2. Struktur If

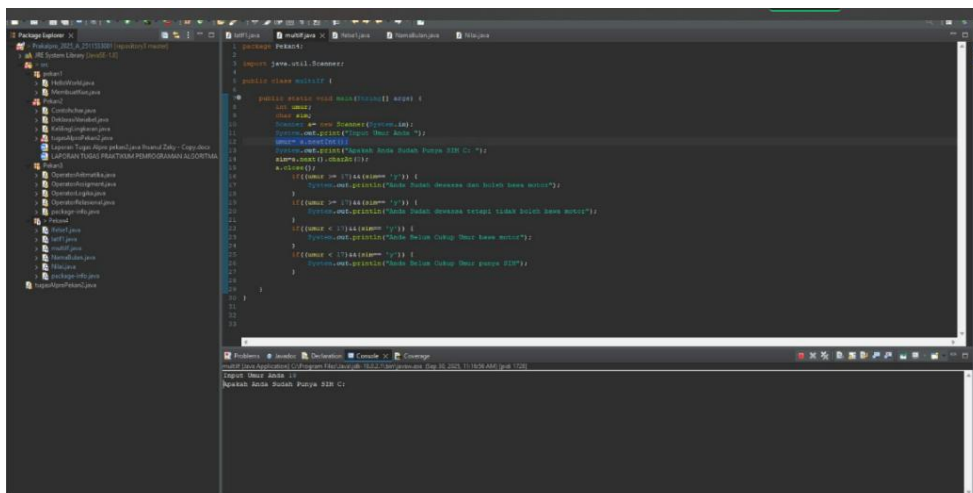
- Buat file Java bernama Intiil di dalam package pekat.
- Import library java.util.Scanner untuk menerima input dari pengguna.
- Buat kelas latifl yang memiliki metode utama public static void main(String[] args).
- Deklarasikan variabel bertipe double dengan nama IPK.
- Buat objek Scanner untuk membaca input dari pengguna.
- Tampilkan pesan "Input IPK Anda=" sebagai petunjuk untuk memasukkan nilai IPK, lalu baca nilai tersebut dan simpan ke variabel IPK.
- Setelah selesai, tutup objek Scanner dengan input.close().
- Gunakan percabangan if untuk mengecek apakah nilai $IPK \geq 2.75$. Jika ya, tampilkan pesan kelulusan.
- Jalankan program dan coba masukkan nilai IPK untuk menguji hasilnya.



3. Struktur Multi If

- Buat file Java bernama multiIf di dalam package pekan4.
- Import library java.util.Scanner untuk membaca input dari pengguna.
- Deklarasikan kelas publik mitilf dengan metode utama public static void main(String[] args).
- Buat variabel int umur dan char sim untuk menyimpan data yang diinput oleh pengguna.
- Buat objek Scanner bernama a untuk membaca input dari keyboard.
- Tampilkan pesan "Input umur anda" dan baca nilai integer yang dimasukkan, simpan ke variabel umur.

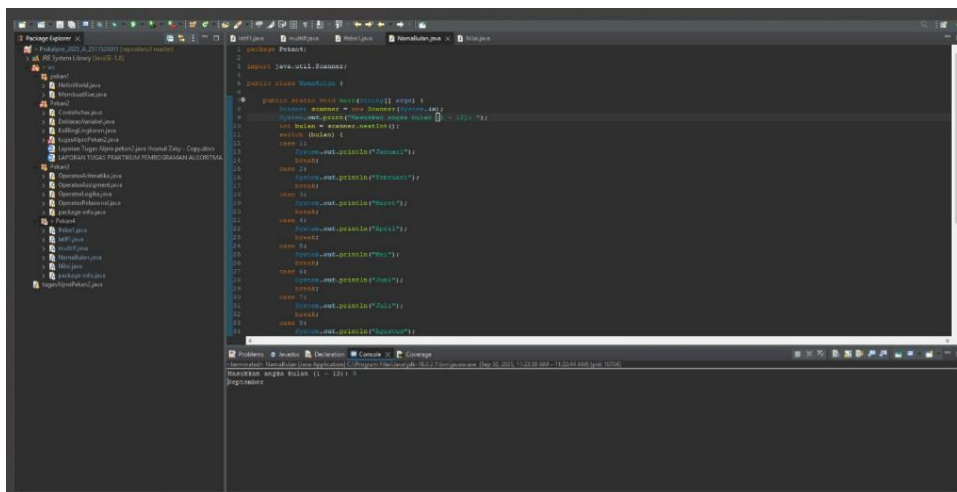
- Tampilkan pesan "Apakah anda sudah punya sim?" lalu baca karakter pertama dari input pengguna dengan `charAt(0)`, dan simpan ke variabel `sim`.
- Setelah selesai, tutup objek `Scanner` dengan `a.close()`.
- Gunakan empat kondisi `if` terpisah untuk memeriksa kombinasi nilai umur dan `sim` sebagai berikut:
 - Jika umur ≥ 17 dan `sim == 'y'`, tampilkan pesan:
"Anda sudah dewasa dan boleh bawa motor"
 - Jika umur ≥ 17 dan `sim != 'y'`, tampilkan pesan:
"Anda sudah dewasa tapi tidak boleh bawa motor"
 - Jika umur < 17 dan `sim != 'y'`, tampilkan pesan:
"Anda belum cukup umur bawa motor"
 - Jika umur < 17 dan `sim == 'y'`, tampilkan pesan:
"Anda belum cukup umur punya sim"



4. Struktur Switch-Case

- Buat file Java bernama `NamaBulan` di dalam package `pekan4`.
- Import library `java.util.Scanner` untuk menerima input dari pengguna.
- Deklarasikan kelas publik `NamaBulan` dengan metode utama `public static void main(String[] args)`.
- Buat variabel bertipe `int` dengan nama `bulan` untuk menyimpan angka yang diinput oleh pengguna.
- Buat objek `Scanner` bernama `scanner` untuk membaca input dari keyboard.
- Tampilkan pesan "Masukan angka bulan (1 - 12):" agar pengguna tahu harus memasukkan angka bulan.
- Baca nilai integer yang dimasukkan dan simpan ke variabel `bulan`.

- Setelah selesai, tutup objek Scanner dengan scanner.close().
- Gunakan struktur switch-case untuk mengecek nilai variabel bulan:
 - Jika bulan == 1, tampilkan pesan "Januari"
 - Jika bulan == 2, tampilkan pesan "Februari"
 - Jika bulan == 3, tampilkan pesan "Maret"
 - Jika bulan == 4, tampilkan pesan "April"
 - Jika bulan == 5, tampilkan pesan "Mei"
 - Jika bulan == 6, tampilkan pesan "Juni"
 - Jika bulan == 7, tampilkan pesan "Juli"
 - Jika bulan == 8, tampilkan pesan "Agustus"
 - Jika bulan == 9, tampilkan pesan "September"
 - Jika bulan == 10, tampilkan pesan "Oktober"
 - Jika bulan == 11, tampilkan pesan "November"
 - Jika bulan == 12, tampilkan pesan "Desember"
 - Jika nilai bulan tidak berada di antara 1 sampai 12, tampilkan pesan "Angka tidak valid" sebagai opsi default.



5. Struktur Else If

- Buat file Java bernama Nilai di dalam package pekan4.
- Import library java.util.Scanner untuk membaca input dari pengguna.
- Deklarasikan kelas publik Nilai dengan metode utama public static void main(String[] args).
- Buat variabel bertipe int dengan nama nilai untuk menyimpan angka input.

- Buat objek Scanner bernama input untuk membaca data dari keyboard.
- Tampilkan pesan "Inputkan nilai angka=" sebagai petunjuk agar pengguna memasukkan nilai.
- Baca nilai integer yang dimasukkan dan simpan ke variabel nilai.
- Setelah selesai, tutup objek Scanner dengan input.close().
- Gunakan struktur percabangan if-else if-else bertingkat untuk mengecek rentang nilai dan menampilkan grade sebagai berikut:
 - Jika nilai ≥ 81 , tampilkan "A"
 - Jika nilai ≥ 70 dan kurang dari 81, tampilkan "B"
 - Jika nilai ≥ 60 dan kurang dari 70, tampilkan "C"
 - Jika nilai ≥ 50 dan kurang dari 60, tampilkan "D"
 - Jika nilai < 50 , tampilkan "E"

The screenshot shows an IDE with a Java file named `Nilai.java`. The code implements a grade calculator using a Scanner to read an integer value and a series of if-else statements to determine the grade. The output window shows the prompt "Inputkan nilai angka:" and the user input "75", which results in the grade "B" being displayed.

```

package Perkuliahan;

import java.util.Scanner;

public class Nilai {

    public static void main(String[] args) {
        int nilai;
        Scanner input = new Scanner(System.in);
        System.out.print("Inputkan nilai angka: ");
        nilai = input.nextInt();
        input.close();

        if (nilai >= 81) {
            System.out.println("A");
        } else if (nilai >= 70) {
            System.out.println("B");
        } else if (nilai >= 60) {
            System.out.println("C");
        } else if (nilai >= 50) {
            System.out.println("D");
        } else {
            System.out.println("E");
        }
    }
}
  
```

Output:

```

Inputkan nilai angka: 75
B
  
```

BAB III

KESIMPULAN

3.1 KESIMPULAN

Dari praktikum yang telah dilakukan, dapat disimpulkan bahwa struktur kontrol percabangan memiliki peran penting dalam mengarahkan alur program berdasarkan kondisi tertentu. Dengan memanfaatkan if, if-else, maupun switch, program dapat mengambil keputusan secara tepat sehingga menghasilkan output yang sesuai dengan kondisi yang terjadi.

3.3 SARAN

Sebaiknya dosen menyelenggarakan sesi pra-praktikum di kelas sebagai pengantar materi, agar mahasiswa memperoleh pemahaman awal yang lebih baik. Dengan demikian, potensi kepanikan atau kesalahan selama praktikum dapat diminimalkan.

Disarankan agar materi praktikum dibagikan terlebih dahulu melalui platform iLearn, sehingga mahasiswa memiliki waktu yang cukup untuk mempelajari dan mempersiapkan diri sebelum pelaksanaan praktikum.

DAFTAR PUSTAKA

[5] Nugroho, ADI . Algoritma dan Struktur Data dalam Bahasa Java. Yogyakarta: Penerbit Andi, 2008.

[2] Hartono, Budi dan tim, Pemrograman Java untuk Pemula. Semarang: Penerbit Yayasan Prima Agus Teknik; 2022..