

Nama & NPM	Topik:	Tanggal:
Nabila azizah Mutiara s G1F024031	IF dan SWITCH Java	26 September 2024

[No. 1] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variable

Contoh 1: Salin dan tempel kode program berikut ke Eclipse.

```
import java.util.Scanner; //memanggil impor package yang membaca masukan pengguna

public class PercabanganIf {
    public static void main(String[] args) {
        Scanner masuk = new Scanner(System.in); // membaca teks yang dimasukkan pengguna
        System.out.print("Masukkan Angka Anda : "); //pengguna memasukkan data
        int nilai = masuk.nextByte(); //menyimpan masukan pengguna ke tipe data

        if (nilai == 10) { //percabangan yang memeriksa kondisi
            System.out.println("Sepuluh"); //baris kode yang dieksekusi bila benar
        }
        else { //baris kode yang dieksekusi bila kondisi tidak terpenuhi dan salah
            System.out.println("Nilai Bukan Sepuluh");
        }
    }
}
```

contoh 2: Salin dan tempel kode program berikut ke Eclipse.

```
import java.util.Scanner;
public class IfBersarang {
    public static void main(String[] args) {
        Scanner varT = new Scanner(System.in);
        System.out.print("Masukkan Angka Tugas Anda : ");
        int nilaiT = varT.nextByte();

        Scanner varQ = new Scanner(System.in);
        System.out.print("Masukkan Angka Quiz Anda : ");
        int nilaiQ = varQ.nextByte();

        if (nilaiU >= 80) {
            if (nilaiT >= 80) {
                System.out.println("Anda mendapatkan nilai A");
            }
        }
        else{
            System.out.println("Anda TIDAK mendapatkan nilai A");
        }
    }
}
```

Diketahui dari soal

1.1. Bilangan genap merupakan bilangan yang habis dibagi 2. Bilangan ganjil adalah bilangan yang tidak habis dibagi 2. Analisa kode program yang tepat untuk menghitung masukan pengguna termasuk bilangan genap atau bilangan ganjil (lihat Contoh 1)? (Petunjuk: hitung = nilai % 2 jika hitung = 0 maka bilangan genap, jika hitung = 1 maka bilangan ganjil)

- 1.2. Cermati contoh 2, analisa kondisi pada IF bersarang!
Tambahkan satu kondisi IF dengan satu nilai input Quiz (nilaiQ). Jika nilai UTS, Tugas, dan Quiz lebih besar sama dengan 80 maka siswa mendapat nilai A.
- 1.3. Apakah ketiga kondisi IF pada Contoh 1.2. dapat diringkas menjadi satu kondisi?
Periksa satu kondisi mana yang paling tepat menggantikan ketiga kondisi itu!
 - a. IF (nilaiU >= 80 || nilaiT >= 80 || nilaiQ >= 80)
 - b. IF (nilaiU >= 80 || nilaiT >= 80 && nilaiQ >= 80)
 - c. IF (nilaiU >= 80 && nilaiT >= 80 || nilaiQ >= 80)
 - d. IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)
- 1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!

[No.1] Analisis dan Argumentasi

- 1) Saya mengusulkan permasalahan ini dapat diatasi dengan cara menyederhanakan kondisi IF untuk memeriksa semua nilai sekaligus dalam satu pernyataan.
- 2) Alasan solusi ini karena kode akan menjadi lebih ringkas dan mudah dibaca,serta akan mengurangi kemungkinan kesalahan logika dalam percabangan.
- 3) Perbaiki kode program dengan cara menggunakan satu kondisi IF yang menggabungkan semua syarat yang diperlukan ,yaitu dengan memeriksa apakah nilai UTS, Tugas dan Quiz semuanya lebih besar sama dengan 80

[No.1] Penyusunan Algoritma dan Kode Program

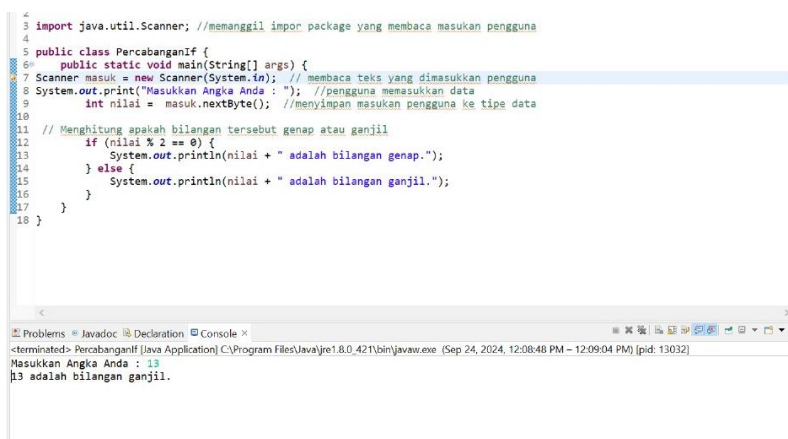
1) Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

Algoritma Bilangan Genap atau Ganjil

- a. Mulai
 - b. Deklarasikan variabel untuk menyimpan input (nilai).
 - c. Tampilkan pesan: "Masukkan Angka Anda : "
 - d. Baca input dari pengguna dan simpan dalam variabel nilai.
 - e. Selesai
- 2) Kode program dan luaran
- a). Screenshot/ Capture potongan kode dan hasil luaran
Beri komentar pada kode yang di Screenshot

Gambar coding 1.1



```
4 import java.util.Scanner; //memanggil impor package yang membaca masukan pengguna
5
6 public class PercabanganIf {
7     public static void main(String[] args) {
8         Scanner masuk = new Scanner(System.in); // membaca teks yang dimasukkan pengguna
9         System.out.print("Masukkan Angka Anda : "); //pengguna memasukkan data
10        int nilai = masuk.nextByte(); //menyimpan masukan pengguna ke tipe data
11
12        // Menghitung apakah bilangan tersebut genap atau ganjil
13        if (nilai % 2 == 0) {
14            System.out.println(nilai + " adalah bilangan genap.");
15        } else {
16            System.out.println(nilai + " adalah bilangan ganjil.");
17        }
18    }
19 }
```

Problems • Javadoc • Declaration • Console ×

<terminated> PercabanganIf [Java Application] C:\Program Files\Java\jre1.8.0_421\bin\javaw.exe (Sep 24, 2024, 12:08:48 PM - 12:09:04 PM) [pid: 13032]

Masukkan Angka Anda : 13

13 adalah bilangan ganjil.

Gambar coding 1.2

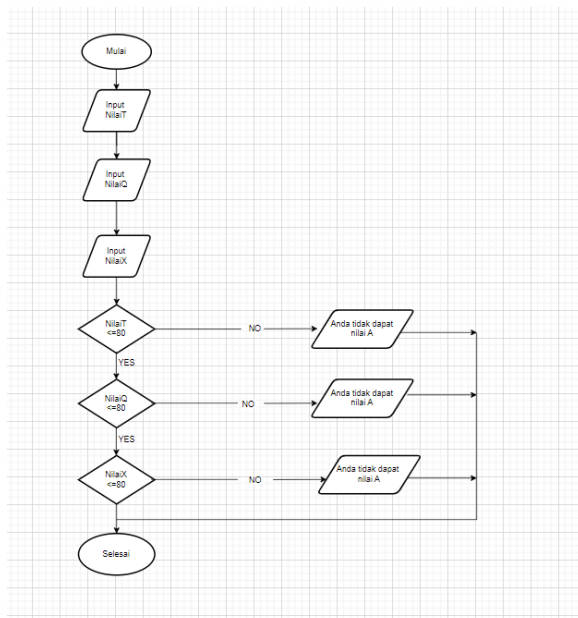
```

3 public class IfBersarang {
4     public static void main(String[] args) {
5         Scanner varU = new Scanner(System.in);
6         System.out.print("Masukkan nilai UTS Anda : ");
7         int nilaiU = varU.nextByte();
8
9         Scanner varT = new Scanner(System.in);
10        System.out.print("Masukkan nilai Tugas Anda : ");
11        int nilaiT = varT.nextByte();
12
13        Scanner varQ = new Scanner(System.in);
14        System.out.print("Masukkan nilai Quiz Anda : ");
15        int nilaiQ = varQ.nextByte();
16
17        if (nilaiU >= 80) {
18            if (nilaiT >= 80) {
19                if (nilaiQ >= 80) {
20                    System.out.println("Anda mendapatkan nilai A");
21                }
22            }
23        }
24        else{
25            System.out.println("Anda TIDAK mendapatkan nilai A");
26        }
27    }
28 }

```

<terminated> IfBersarang [Java Application] C:\Program Files\Java\jre1.8.0_421\bin\javaw.exe (Sep 24, 2024, 12:09:22 PM - 12:09:44 PM) [pid: 7024]
 Masukkan nilai UTS Anda : 97
 Masukkan nilai Tugas Anda : 95
 Masukkan nilai Quiz Anda : 80
 Anda mendapatkan nilai A

Gambar flowchart 1.3



- a) Analisa luaran yang dihasilkan
SOAL 1.1

II. Analisis Kode Program

(import java.util.Scanner; //memanggil impor package yang membaca masukan pengguna)

- Pada baris ini, program mengimpor kelas Scanner dari paket java.util. Kelas ini digunakan untuk mengambil input dari pengguna.

(public class PercabanganIf {})

- Kode ini mendeklarasikan kelas PercabanganIf. Kelas ini menjadi wadah bagi seluruh kode yang akan dieksekusi.

(public static void main(String[] args) {)

- Metode main adalah titik awal eksekusi program. Tanpa metode ini, program tidak dapat dijalankan.

(Scanner masuk = new Scanner(System.in); // membaca teks yang dimasukkan pengguna

System.out.print("Masukkan Angka Anda : "); //pengguna memasukkan data

int nilai = masuk.nextByte(); //menyimpan masukan pengguna ke tipe data)

- Di sini, objek Scanner bernama masuk dibuat untuk membaca input dari pengguna.

Pengguna diminta untuk memasukkan sebuah angka yang disimpan dalam variabel nilai dengan tipe data byte.

```
// Menghitung apakah bilangan tersebut genap atau ganjil
if (nilai % 2 == 0) {
    System.out.println(nilai + " adalah bilangan genap.");
} else {
    System.out.println(nilai + " adalah bilangan ganjil.");
}
```

- Di bagian ini, program memeriksa apakah nilai habis dibagi 2 (menggunakan operator modulus %). Jika hasilnya 0, maka bilangan tersebut adalah genap, dan pesan yang sesuai ditampilkan. Jika tidak, program menampilkan bahwa bilangan tersebut adalah ganjil.

Program ini berhasil menentukan apakah bilangan yang dimasukkan oleh pengguna adalah genap atau ganjil. Struktur if-else yang digunakan sederhana dan efektif untuk tujuan ini.

SOAL NO 1.2

Program ini dirancang apakah seorang siswa tersebut dapat meraih nilai A atau tidak berdasarkan nilai Ujian Tengah Semester (UTS), Tugas, dan Quiz.

1. Impor Scanner: Kode ini mengimpor Scanner untuk membaca input dari pengguna.
2. Deklarasi Kelas dan Metode Utama: Kelas If bersarang dan metode main adalah struktur dasar untuk menjalankan program.
3. Input Nilai: Pengguna diminta untuk memasukkan nilai UTS, Tugas, dan Quiz secara berurutan. Setiap nilai disimpan dalam variabel yang sesuai.
4. Pemeriksaan Nilai: Program menggunakan struktur if bersarang untuk memeriksa apakah semua nilai (UTS, Tugas, dan Quiz) minimal 80. Jika semua memenuhi syarat, program menampilkan "Anda mendapatkan nilai A".
Jika tidak, program hanya menunjukkan bahwa siswa tidak mendapatkan nilai A, tanpa memberikan detail lebih lanjut.
5. Catatan: Disarankan untuk menggunakan satu objek Scanner untuk semua input agar lebih efisien, dan menambahkan validasi untuk menangani input yang tidak valid.

Secara keseluruhan, program ini sederhana dan efektif, tetapi ada beberapa perbaikan yang bisa dilakukan untuk meningkatkan kejelasan dan efisiensi.

SOAL NO 1.3

Merangkum Kondisi IF Ketiga kondisi IF dapat diringkas menjadi satu kondisi. Pilihan yang paling tepat adalah:

d. IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)

dikarenakan, pilihan ini memeriksa apakah semua nilai (UTS, Tugas, dan Quiz) harus >= 80 untuk mendapatkan nilai A kemudian menggunakan operator && memastikan bahwa ketiga kondisi harus benar sekaligus. Jika salah satu dari ketiga nilai tersebut kurang dari 80, maka siswa tidak akan mendapatkan nilai A dan juga pilihan lainnya (a, b, dan c) menggunakan operator ||, yang tidak sesuai dengan syarat yang diinginkan karena hanya perlu salah satu yang benar untuk memenuhi kondisi, yang tidak sesuai dengan kriteria nilai A.

SOAL NO 1.4

Flowchart ini menggambarkan alur logika untuk menentukan apakah seorang siswa mendapatkan nilai A berdasarkan tiga komponen nilai UTS, Tugas, dan Quiz. Proses dimulai dengan pengambilan input dari pengguna dan diakhiri dengan output berdasarkan hasil evaluasi nilai.

2. Langkah-langkah Flowchart

1. Mulai: proses dimulai
2. Kemudian mengimput nilai T Q DAN X di setiap kolom yang berbeda

3. Memeriksa nilai

Contoh Apakah nilai $T \geq 80$?:

- Ini adalah keputusan pertama dalam flowchart. Jika nilai UTS memenuhi syarat, program melanjutkan ke langkah berikutnya, jika iya melanjutkan untuk memeriksa nilai Tugas yang ada dibawah nilai T hingga sampai ke kolom selesai maka output yang dihasilkan adalah “ anda mendapatkan nilai A” Jika Tidak ($nilaiT < 80$) Output "Anda TIDAK mendapatkan nilai A". Jika Tidak ($nilaiT < 80$) Output "Anda TIDAK mendapatkan nilai A". Jika Tidak ($nilaiU < 80$) Output "Anda TIDAK mendapatkan nilai A".

4. Selesai:Proses berakhir.

Output yang dikeluarkan telah sesuai dengan codingan yang telah di analisis semua kesalahan yang ada dalam pemrograman.

:

1) Analisa

- a) Permasalahan yang dapat disimpulkan pada laporan pertama ini adalah program bertujuan untuk menentukan apakah angka yang dimasukkan pengguna genap atau ganjil dan mengevaluasi nilai siswa untuk memberikan nilai A. berdasarkan algoritmanya permasalahan yang ada dapat dipecahkan dengan menggunakan modulus programan ganjil genap, memeriksa apakah nilai UTS, Tugas, dan Quiz semuanya ≥ 80 untuk memberikan nilai A. kemudian kode yang ditulis mengikuti algoritma tersebut dengan jelas menggunakan struktur if-else.
- b) Saya memilih menggunakan Scanner untuk input pengguna karena metode ini memungkinkan fleksibilitas, perbaikan program dengan menambahkan pengecekan input untuk memastikan bahwa pengguna hanya memasukkan nilai numerik, karena struktur Java mengharuskan penanganan kesalahan, saya menggunakan if-else untuk membuat keputusan yang jelas dan mudah dibaca.

[No. 2] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variabel

Contoh 3: Salin dan tempel kode program berikut ke Eclipse.

```
import java.util.Scanner;

public class SwitchBersarang {
    public static void main(String[] args) {
        Scanner masukData = new Scanner(System.in);
        // mengambil input
        System.out.print("Pilih A atau B : ");
        char data = masukData.next().charAt(0);
        switch(data) {
            case 'A':
                System.out.print("Anda sudah rajin belajar");
                break; // baris 1
            case 'B':
                System.out.print("  Anda perlu kurangi main game");
                break; // baris 2
            default:
                System.out.print("  Pilihan anda diluar A atau B ");
        }
    }
}
```

Contoh 4: Salin dan tempel kode program berikut ke Eclipse.

```
import java.util.Scanner;

public class SwitchBersarang {

    public static void main(String[] args) {
        byte bulan;
        int tahun = 2022;
        int jumlahHari = 0;
        System.out.print("Masukkan data bulan (dalam angka): ");
        Scanner masukData = new Scanner(System.in);
        bulan = masukData.nextByte();

        switch (bulan) {
            case 1: case 3: //baris 1
                jumlahHari = 31;
                break;
            case 4: //baris 2
                jumlahHari = 30;
                break;
            case 2:
                if (tahun % 4 == 0)
                    jumlahHari = 29;
                else
                    jumlahHari = 28;
                break;
            default:
                System.out.println("Maaf bulan hanya sampai 12.");
                break;
        }
        System.out.println("Jumlah hari = " + jumlahHari);
    }
}
```

Diketahui dari soal :

2.1. Cermati kode pada Contoh 3. Hapuslah kode break; pada //baris 1, eksekusi kembali.

Kemudian hapuslah kode break; pada //baris 2, eksekusi Kembali. Analisis perbedaan hasil luaran ketika kode menggunakan break, Ketika kode break baris dihapus, dan ketika kode break baris 2 dihapus! Analisa kegunaan baris kode dengan break dan kata kunci default!

- 2.2. Cermati kode pada Contoh 4 yang menampilkan jumlah hari sesuai dengan bulannya. Namun kode tersebut baru sampai bulan ke-4. Tambahkan sampai bulan ke-12 pada baris ke-1 dan baris ke-2.
- 2.3. Cermati permasalahan yang dipecahkan pada Contoh 3.
Apakah masalah ini bisa diubah menjadi perintah IF? Jika bisa, rincikan analisa Anda! Bandingkan masalah yang dapat diselesaikan percabangan dengan IF atau SWITCH !
- 2.4. Buatlah dokumentasi gambar flowchart dari Latihan 2.1. dan Latihan 2.2!

[No.1] Analisis dan Argumentasi

- 1) Saya mengusulkan permasalahan ini dapat diatasi dengan cara memanfaatkan pernyataan break dalam switch case dan menambah kasus dalam kode
- 2) Alasan solusi ini karena akan membuat program lebih lengkap dan fungsional

Perbaiki kode program dengan cara menambahkan kasus untuk bulan 5-12 dan memastikan setiap case diakhiri dengan break untuk menjaga kejelasan dan keakuratan output.

[No.2] Penyusunan Algoritma dan Kode Program

- 1) Algoritma

Algoritma contoh 3 soal 2.1

1. Mulai
2. Tampilkan pesan: "Pilih A atau B:"
3. Baca input karakter data
4. Lakukan switch pada data:
5. Selesai

Algoritma contoh 4 soal 2.2

1. Mulai
2. Tampilkan pesan: "Masukkan data bulan (dalam angka):"
3. Baca input bulan
4. Set jumlahHari ke 0
5. Lakukan switch pada bulan
6. Tampilkan "Jumlah hari = " dan nilai jumlahHari
7. Selesai

Algoritma Soal no 2.3 (if vs switch)

1. Tampilkan pesan: "Pilih A atau B:"
2. Baca input karakter data
3. Jika data adalah 'A'
Tampilkan "Anda sudah rajin belajar"
4. Jika data adalah 'B':
Tampilkan "Anda perlu kurangi main game"
5. Jika tidak ada yang cocok:
Tampilkan "Pilihan anda diluar A atau B"
6. Selesai

2. Screenshot/ Capture potongan kode dan hasil luaran

Gambar 2.1

```
import java.util.Scanner;

public class SwitchBersarang {
    public static void main(String[] args) {
        Scanner masukData = new Scanner(System.in);
        // mengambil input
        System.out.print("Pilih A atau B : ");
        char data = masukData.next().charAt(0);
        switch(data) {
            case 'A':
                System.out.print("Anda sudah rajin belajar");
                // breakbaris 1
            case 'B':
                System.out.print(" Anda perlu kurangi main game");
                // break baris 2
            default:
                System.out.print(" Pilihan anda diluar A atau B ");
        }
    }
}

java -cp /tmp/Nfsux0iIQ0/SwitchBersarang
Pilih A atau B : B
Anda perlu kurangi main game Pilihan anda diluar A atau B
=== Code Execution Successful ===

Pilih A atau B : A
Anda sudah rajin belajar Anda perlu kurangi main game Pilihan anda
diluar A atau B
=== Code Execution Successful ===
```

Gambar 2.2

```
import java.util.Scanner;

public class SwitchBersarang {

    public static void main(String[] args) {
        byte bulan;
        int tahun = 2022;
        int jumlahHari = 0;
        System.out.print("Masukkan data bulan (dalam angka):
        ");
        Scanner masukData = new Scanner(System.in);
        bulan = masukData.nextByte();

        switch (bulan) {
            case 1: case 3: //baris 1
                jumlahHari = 31;
                break;
            case 4: //baris 2
                jumlahHari = 30;
                break;
            case 2:
                if (tahun % 4 == 0)
                    jumlahHari = 29;
                else
                    jumlahHari = 28;
            case 5: case 7: case 8: case 10: case 12:
                jumlahHari = 31;
                break;
            case 6: case 9: case 11:
                jumlahHari = 30;
                break;
            default:
                System.out.println("Maaf bulan hanya sampai
                12.");
                break;
        }
        System.out.println("Jumlah hari = " + jumlahHari);
    }
}
```



```
Masukkan data bulan (dalam angka): 7
Jumlah hari = 31
```

```
=== Code Execution Successful ===
```

Gambaran 2.3

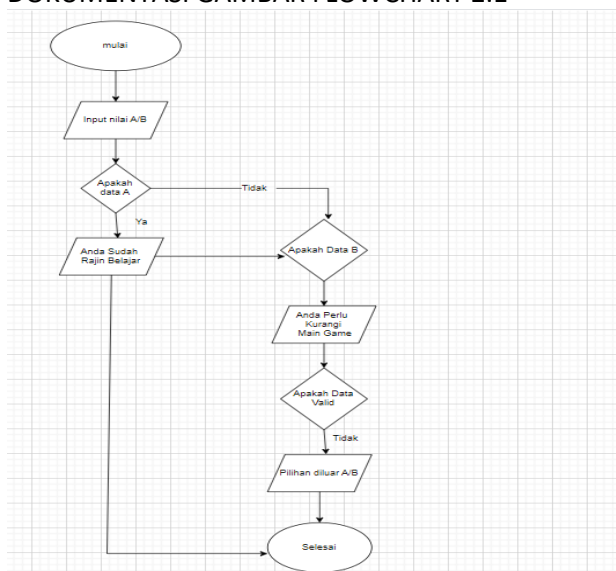
```
public class PilihanBelajar {
    public static void main(String[] args) {
        Scanner masukData = new Scanner(System.in);

        // Mengambil input
        System.out.print("Pilih A atau B: ");
        char data = masukData.next().charAt(0);

        // Menggunakan if-else untuk menangani pilihan
        if (data == 'A') {
            System.out.print("Anda sudah rajin belajar");
        } else if (data == 'B') {
            System.out.print("Anda perlu kurangi main game");
        } else {
            System.out.print("Pilihan anda diluar A atau B");
        }
    }
}
```

```
java -cp /tmp/qywlMuybeD/PilihanBelajar
Pilih A atau B: B
Anda perlu kurangi main game
=== Code Execution Successful ===
```

DOKUMENTASI GAMBAR FLOWCHART 2.1



DOKUMENTASI GAMBAR FLOWCHART 2.2



a) Analisa luaran yang dihasilkan

Analisa gambar 2.1

- **Dengan break:**

Jika pengguna memilih 'A', program menampilkan "Anda sudah rajin belajar" dan keluar dari switch. Jika memilih 'B', program menampilkan "Anda perlu kurang main game" dan keluar dari switch.

- **Tanpa break pada (case 'A'):**

Jika memilih 'A', program menampilkan "Anda sudah rajin belajar" dan terus mengeksekusi semua case berikutnya, sehingga juga menampilkan "Anda perlu kurang main game".

- **Tanpa break pada baris 2 (case 'B'):**

Jika memilih 'B', program menampilkan "Anda perlu kurang main game" dan kemudian juga mengeksekusi default, yang mengakibatkan "Pilihan anda diluar A atau B" ditampilkan.

Break itu berfungsi untuk menghentikan case yang sedang berjalan sedangkan default, default menangani semua kondisi yang tidak ditangkap oleh case yang ada,

Analisa 2.2

Untuk melengkapi kode saya menambahkan kode dbawah ini pada swicth

case 5: case 7: case 8: case 10: case 12:

```
jumlahHari = 31;
```

```
break;
```

case 6: case 9: case 11:

```
jumlahHari = 30;
```

```
break;
```

Analisa 2.3

Pemrograman tersebut dapat diselesaikan dengan if else

Perbedaan yang terjadi adalah:

- **Switch:** itu dia lebih bersih dan mudah dibaca ketika ada banyak kondisi untuk satu variable, switch juga cocok untuk karakter atau integer
- **IF:** lebih fleksibel untuk kondisi kompleks dan kombinasi logika, dan dapat digunakan untuk berbagai tipe data

Kesimpulan

Analisa

- a) Pada laporan ini memecahkan masalah apabila break dihapus pada contoh codingan 3, yang menghasilkan output yang tidak diinginkan kemudian pada contoh 4 kode ini menghitung jumlah hari berdasarkan bulan yang dimasukkan. Permasalahan muncul ketika kode hanya menangani 4 bulan pertama, sehingga bulan 5 hingga 12 perlu ditambahkan. Lalu untuk algoritma **Contoh 3:** Algoritma menggunakan struktur kontrol switch untuk menentukan respons berdasarkan input karakter. Kegunaan break adalah untuk mencegah eksekusi berlanjut ke case berikutnya. **Contoh 4:** Algoritma menggunakan switch untuk menentukan jumlah hari dalam bulan. Struktur ini juga memerlukan penanganan untuk bulan yang lebih dari 4.
- b) Lalu kegunaan break adalah mengontrol alur eksekusi, switch lebih baik untuk pilihan tetap sedangkan if lebih fleksibel untuk kondisi kompleks., dan menambahkan bulan hingga 12 penting untuk akurasi dan fungsionalitas aplikasi sebagai pemerluasan kode

REFLEKSI

Selama minggu ini, saya belajar banyak tentang penggunaan struktur kontrol dalam pemrograman, terutama pernyataan switch dan if. Saya memahami bagaimana kedua struktur tersebut dapat digunakan untuk menangani kondisi berbeda dalam program. Penggunaan break dalam switch sangat penting untuk mencegah hasil yang tidak diinginkan.