

Nama & NPM	Topik:	Tanggal:
Mellinna Husadya G1F024006	UNIT IF DAN SWITCH	23 September 2024

[No. 1] Identifikasi Masalah:

1) Uraikan permasalahan dan variabel

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");
        }
    }
}
```

Luaran Contoh 1:

```
Masukkan Angka Anda : 8
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");
    }
}

```

Luaran Contoh 2:

Masukkan Angka Tugas Anda : 70

Masukkan Angka Quiz Anda : 70

Masukkan Angka UTS Anda : 70

Anda TIDAK mendapatkan nilai A

Latihan 1:

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: $\text{hitung} = \text{nilai} \% 2$ jika $\text{hitung} = 0$ maka bilangan genap, jika $\text{hitung} = 1$ 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!

- IF (nilaiU >= 80 || nilaiT >= 80 || nilaiQ >= 80)
- IF (nilaiU >= 80 || nilaiT >= 80 && nilaiQ >= 80)
- IF (nilaiU >= 80 && nilaiT >= 80 || nilaiQ >= 80)
- IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)

1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!

[No.1] Analisis dan Argumentasi

1.1 Program Java ini memanfaatkan kelas Scanner untuk mengambil input dari pengguna dan menyimpan nilai tersebut dalam variabel bertipe int. Setelah itu, program memeriksa apakah angka yang dimasukkan sama dengan sepuluh; jika iya, akan mencetak "Sepuluh", sedangkan jika tidak, akan

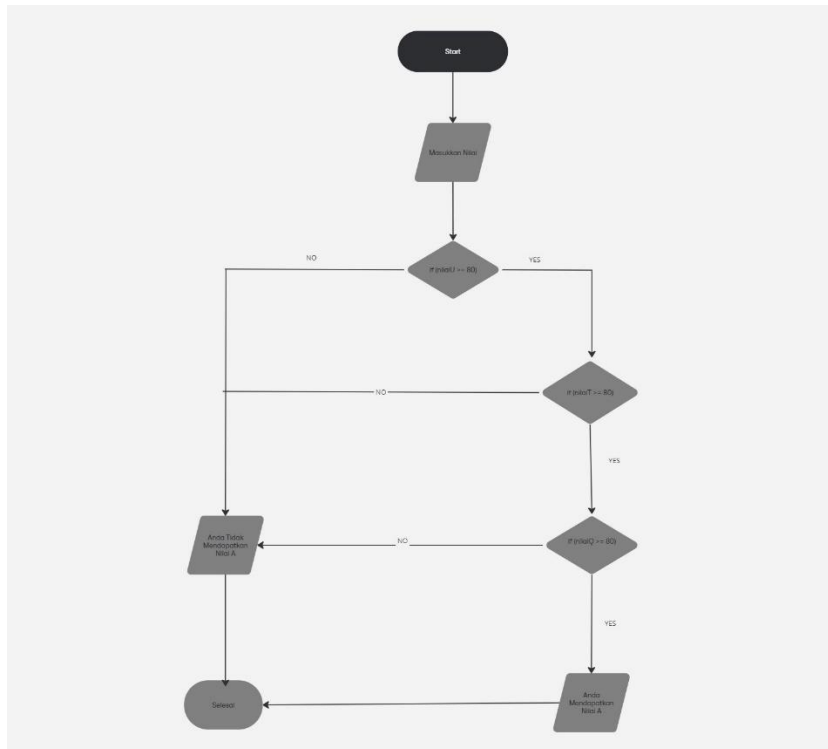
menampilkan "Nilai Bukan Sepuluh". Kemudian, program juga menentukan apakah angka tersebut genap atau ganjil dengan menghitung sisa pembagian angka dengan 2: jika sisa bagi adalah 0, angka tersebut genap, dan jika 1, angka tersebut ganjil. Dengan demikian, program ini berhasil memberikan informasi yang jelas berdasarkan input yang diberikan oleh pengguna.

1.2 Program Java ini menggunakan struktur if bersarang untuk menentukan apakah seorang siswa berhak mendapatkan nilai A berdasarkan tiga komponen: nilai UTS, nilai Tugas, dan nilai Quiz. Setelah meminta pengguna memasukkan masing-masing nilai, program akan memeriksa terlebih dahulu apakah nilai UTS (nilaiU) setidaknya 80. Jika ya, program kemudian akan memeriksa nilai Tugas (nilaiT); jika nilaiT juga 80 atau lebih, program akan melanjutkan untuk mengecek nilai Quiz (nilaiQ). Hanya jika semua syarat tersebut terpenuhi, program akan menampilkan "Anda mendapatkan nilai A". Jika salah satu syarat tidak terpenuhi, program akan mencetak "Anda TIDAK mendapatkan nilai A". Dengan cara ini, struktur if bersarang memungkinkan program untuk mengevaluasi beberapa kondisi secara berurutan sebelum memberikan hasil akhir

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

Pilihan **d. IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)** adalah yang paling tepat karena kondisi ini memastikan bahwa semua nilai yang diperoleh siswa—UTS, Tugas, dan Quiz—harus minimal 80 untuk mendapatkan nilai A. Dengan menggunakan operator && (dan), program akan mengembalikan true hanya jika ketiga syarat tersebut terpenuhi secara bersamaan. Ini mencerminkan kriteria yang jelas dan ketat untuk mencapai nilai A, sesuai dengan logika yang diinginkan, yaitu tidak ada satu pun nilai yang boleh di bawah 80.

1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!



[No.1] Penyusunan Algoritma dan Kode Program

1) Algoritma contoh 1

- Masuk ke aplikasi eceplis
- Import java
- Membuat kelas
- Masukkan kode program
- Perbaiki kode program dengan menambahkan bilangan genap / bilangan ganjil
- Run
- Selesai

2) Algoritma contoh 2

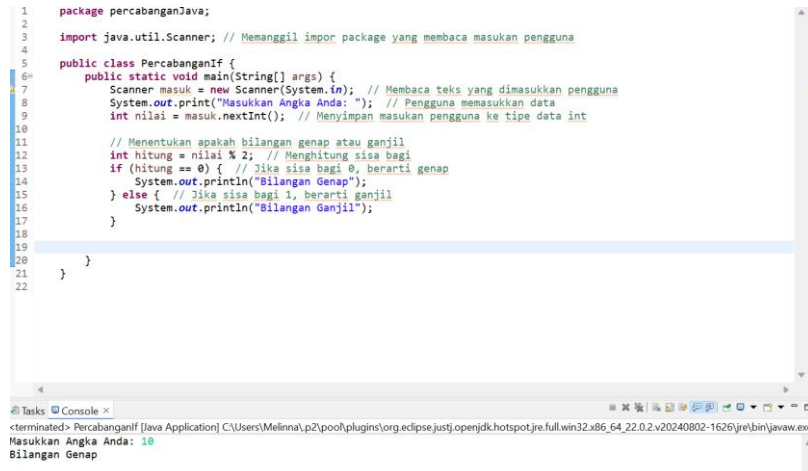
- Masuk ke aplikasi eclipse
- Import java
- Membuat kelas
- Masukkan kode program
- Perbaiki kode program dengan menambah kan if kondisi dan scanner varU
- Run kode program
- Selesai

3) Kode program dan luaran

a) Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

GAMBAR INPUT DAN OUTPUT PADA CONTOH 1

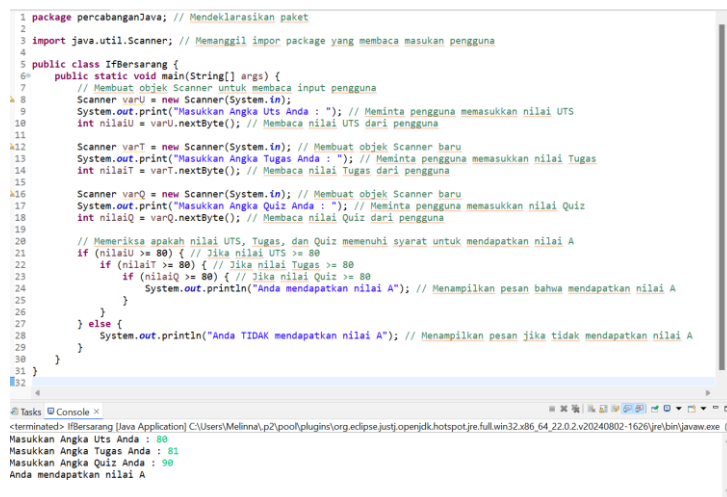


```
1 package percabanganJava;
2
3 import java.util.Scanner; // Memanggil impor package yang membaca masukan pengguna
4
5 public class PercabanganIf {
6     public static void main(String[] args) {
7         Scanner masuk = new Scanner(System.in); // Membaca teks yang dimasukkan pengguna
8         System.out.print("Masukkan Angka Anda: "); // Pengguna memasukkan data
9         int nilai = masuk.nextInt(); // Menyimpan masukan pengguna ke tipe data int
10
11         // Menentukan apakah bilangan genap atau ganjil
12         int hitung = nilai % 2; // Menghitung sisa bagi
13         if (hitung == 0) { // Jika sisa bagi 0, berarti genap
14             System.out.println("Bilangan Genap");
15         } else { // Jika sisa bagi 1, berarti ganjil
16             System.out.println("Bilangan Ganjil");
17         }
18     }
19 }
20
21 }
22
```

Tasks Console x

<terminated> PercabanganIf [Java Application] C:\Users\Melinda\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.22.0.2.v20240802-1626\jre\bin\javaw.exe
Masukkan Angka Anda: 10
Bilangan Genap

GAMBAR INPUT DAN OUTPUT PADA CONTOH 2



```
1 package percabanganJava; // Mendeklarasikan paket
2
3 import java.util.Scanner; // Memanggil impor package yang membaca masukan pengguna
4
5 public class IfBersarang {
6     public static void main(String[] args) {
7         // Membuat objek Scanner untuk membaca input pengguna
8         Scanner varU = new Scanner(System.in);
9         System.out.print("Masukkan Angka Uts Anda : "); // Meminta pengguna memasukkan nilai UTS
10        int nilaiU = varU.nextInt(); // Membaca nilai UTS dari pengguna
11
12        Scanner varT = new Scanner(System.in); // Membuat objek Scanner baru
13        System.out.print("Masukkan Angka Tugas Anda : "); // Meminta pengguna memasukkan nilai Tugas
14        int nilaiT = varT.nextInt(); // Membaca nilai Tugas dari pengguna
15
16        Scanner varQ = new Scanner(System.in); // Membuat objek Scanner baru
17        System.out.print("Masukkan Angka Quiz Anda : "); // Meminta pengguna memasukkan nilai Quiz
18        int nilaiQ = varQ.nextInt(); // Membaca nilai Quiz dari pengguna
19
20        // Memeriksa apakah nilai UTS, Tugas, dan Quiz memenuhi syarat untuk mendapatkan nilai A
21        if (nilaiU >= 80) { // Jika nilai UTS >= 80
22            if (nilaiT >= 80) { // Jika nilai Tugas >= 80
23                if (nilaiQ >= 80) { // Jika nilai Quiz >= 80
24                    System.out.println("Anda mendapatkan nilai A"); // Menampilkan pesan bahwa mendapatkan nilai A
25                }
26            }
27        } else {
28            System.out.println("Anda TIDAK mendapatkan nilai A"); // Menampilkan pesan jika tidak mendapatkan nilai A
29        }
30    }
31 }
32
```

Tasks Console x

<terminated> IfBersarang [Java Application] C:\Users\Melinda\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.22.0.2.v20240802-1626\jre\bin\javaw.exe (2)
Masukkan Angka Uts Anda : 80
Masukkan Angka Tugas Anda : 81
Masukkan Angka Quiz Anda : 90
Anda mendapatkan nilai A

b) Analisa luaran yang dihasilkan contoh 1

Jawab:

Hasil yang dihasilkan sudah sesuai dengan yang diharapkan dalam program yang telah dibuat. Tipe data yang ditampilkan juga telah sesuai dengan kebutuhan dan permintaan data yang diinginkan.

- c) Analisa luaran yang dihasilkan contoh 2

Jawab:

Luaran sudah sesuai dengan program yang disusun.

Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data.

1) Analisa contoh 1

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!

Jawab:

Program ini dirancang untuk menentukan apakah angka yang dimasukkan pengguna merupakan bilangan genap atau ganjil. Algoritma yang digunakan memanfaatkan operasi modulus untuk menghitung sisa pembagian angka dengan 2. Jika hasilnya 0, angka tersebut dianggap genap; jika hasilnya 1, maka angka tersebut ganjil. Kode program menyusun logika ini dalam struktur if yang sederhana, sehingga hasilnya dapat ditampilkan dengan jelas. Secara keseluruhan, program ini efektif dalam memberikan informasi yang akurat tentang kategori bilangan berdasarkan input dari pengguna.

- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawab:

Dasar pengambilan keputusan dalam kasus ini adalah untuk memberikan cara yang mudah dan jelas dalam menentukan apakah suatu bilangan genap atau ganjil berdasarkan input pengguna. Menggunakan operasi modulus adalah metode yang praktis dan efisien, karena memungkinkan evaluasi langsung tanpa harus memeriksa setiap bilangan satu per satu. Dengan demikian, program ini membantu pengguna memahami kategori bilangan dengan lebih baik dan memudahkan proses identifikasi.

2) Analisa contoh 2

- c) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!

Jawab:

Program ini dirancang untuk menentukan apakah siswa mendapatkan nilai A berdasarkan tiga komponen: nilai UTS, nilai Tugas, dan nilai Quiz. Algoritma yang digunakan memeriksa setiap nilai secara berurutan, dimulai dari UTS, kemudian Tugas, dan terakhir Quiz, dengan syarat semua nilai harus minimal 80. Kode program menerapkan struktur if bersarang untuk mengecek kondisi ini. Jika semua syarat terpenuhi, program mencetak "Anda mendapatkan nilai A"; jika tidak, akan muncul pesan "Anda TIDAK mendapatkan nilai A." Secara keseluruhan, program ini efektif dalam menilai kinerja siswa sesuai kriteria yang ditetapkan.

d) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawab:

Dasar pengambilan keputusan dalam kasus ini adalah pentingnya memiliki kriteria yang jelas dan ketat untuk penilaian akademik. Dengan menggunakan struktur if bersarang, program memastikan semua komponen nilai dihitung secara adil dan sistematis.

Menetapkan minimal 80 sebagai syarat mencerminkan standar tinggi untuk pencapaian akademik, sehingga hanya siswa yang menunjukkan kinerja baik di semua aspek yang akan mendapatkan nilai A. Ini juga mendorong siswa untuk berusaha lebih keras dalam setiap mata pelajaran.

[No. 2] Identifikasi Masalah:

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 ");
        }
    }
}
```

Luaran Contoh 3:

```
Pilih A atau B : A
Anda sudah rajin belajar
```

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);
}
}

```

Luaran Contoh 4:

```

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

```

Latihan 2:

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 1 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.2] Analisis dan Argumentasi

2.1 Ketika menggunakan kode break dan saat keluar output pilih A atau B disini saya memilih pilih A yang akan terjadi adalah kode akan mencetak “Anda sudah rajin belajar” ketika saya memasukkan B saat output tercetak maka akan keluar “Anda perlu kurangi main game”

Ketika kode break pada baris ke satu di hapus akan mencetak pilih A atau B disini saya mencetak nilai A yang akan menghasilkan “Anda sudah rajin belajar Anda perlu kurangi main game” dan ketika saya menghapus break di baris ke 2 dan memulihkan break pada baris pertama maka yang akan keluar adalah “Anda sudah rajin belajar” dan saat saya menghapus break 1 dan break 2 maka akan mencetak “Anda sudah rajin belajar Anda perlu kurangi main game Pilihan anda diluar A atau B”

break berfungsi untuk menghentikan eksekusi setelah suatu case, memastikan logika program sesuai dengan pilihan pengguna, sementara default menangani situasi ketika input tidak sesuai dengan case yang ada, memberikan umpan balik yang jelas jika pengguna memasukkan pilihan yang tidak valid.

2.2

```
1 package percabanganJava;
2 import java.util.Scanner;
3 public class SwitchJavaBersarang {
4
5     public static void main(String[] args) {
6         byte bulan;
7         int tahun = 2022;
8         int jumlahHari = 0;
9
10        System.out.print("Masukkan data bulan (dalam angka): ");
11        Scanner masukData = new Scanner(System.in);
12        bulan = masukData.nextByte();
13
14        switch (bulan) {
15            case 1: case 3: case 5: case 7: case 8: case 10: case 12: // baris 1
16                jumlahHari = 31;
17                break;
18            case 4: case 6: case 9: case 11: // baris 2
19                jumlahHari = 30;
20                break;
21            case 2:
22                if (tahun % 4 == 0 && (tahun % 100 != 0 || tahun % 400 == 0))
23                    jumlahHari = 29; // Memeriksa tahun kabisat
24                else
25                    jumlahHari = 28;
26                break;
27            default:
28                System.out.println("Haaf bulan hanya sampai 12.");
29                break;
30        }
31        System.out.println("Jumlah hari = " + jumlahHari);
32    }
33 }
34
35
```

2.3

```
Main.java  [Icons]  Run  Output  Clear
1 import java.util.Scanner;
2
3 public class SwitchBersarang {
4     public static void main(String[] args) {
5         Scanner masukData = new Scanner(System.in);
6         // mengambil input
7         System.out.print("Pilih A atau B : ");
8         char data = masukData.next().charAt(0);
9
10        if (data == 'A') {
11            System.out.print("Anda sudah rajin belajar");
12        } else if (data == 'B') {
13            System.out.print("Anda perlu kurangi main game");
14        } else {
15            System.out.print("Pilihan anda diluar A atau B");
16        }
17    }
18 }
19
```

```
java -cp /tmp/bFnksAJRWZ/SwitchBersarang
Pilih A atau B : A
Anda sudah rajin belajar
=== Code Execution Successful ===
```

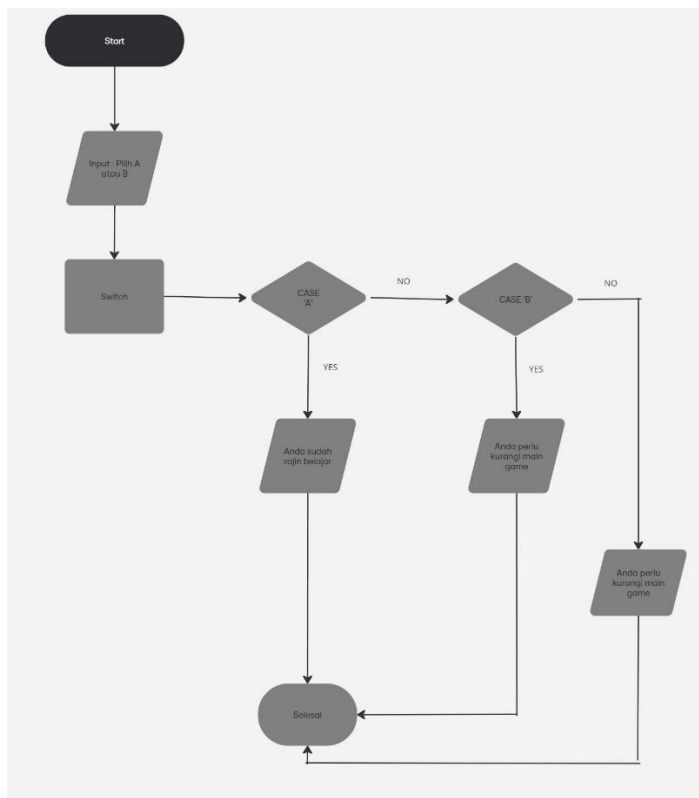
Ya, masalah di atas dapat diubah agar menggunakan perintah `if`. Pertama, kita dapat menggunakan struktur `if` untuk memeriksa nilai variabel `data` yang diberikan, yaitu, memeriksa nilai apakah sama

dengan 'A' atau 'B', kemudian memberikan respons sesuai nilai tersebut. Pertama-tama, jika 'data' sama dengan 'A', maka kita akan mencetak pesan bahwa pengguna sudah rajin belajar, jika sama dengan 'B', kita akan mencetak pesan bahwa pengguna harus belajar lebih sedikit bermain game, jika tidak keduanya, maka kita akan mencetak pesan bahwa "input tidak valid". Terlepas dari kenyataan bahwa baik 'if' maupun perintah 'switch' sama baiknya dalam hal menyelesaikan masalah yang diberikan, perintah 'switch' masih akan lebih jelas dan sering lebih praktis ketika kita hanya memiliki beberapa pilihan sederhana. Di sisi lain, perintah 'if' adalah lebih fleksibel untuk pemeriksaan kondisi yang lebih kompleks. Sederhananya, pilihan tergantung pada jumlah kondisi yang perlu diperiksa.

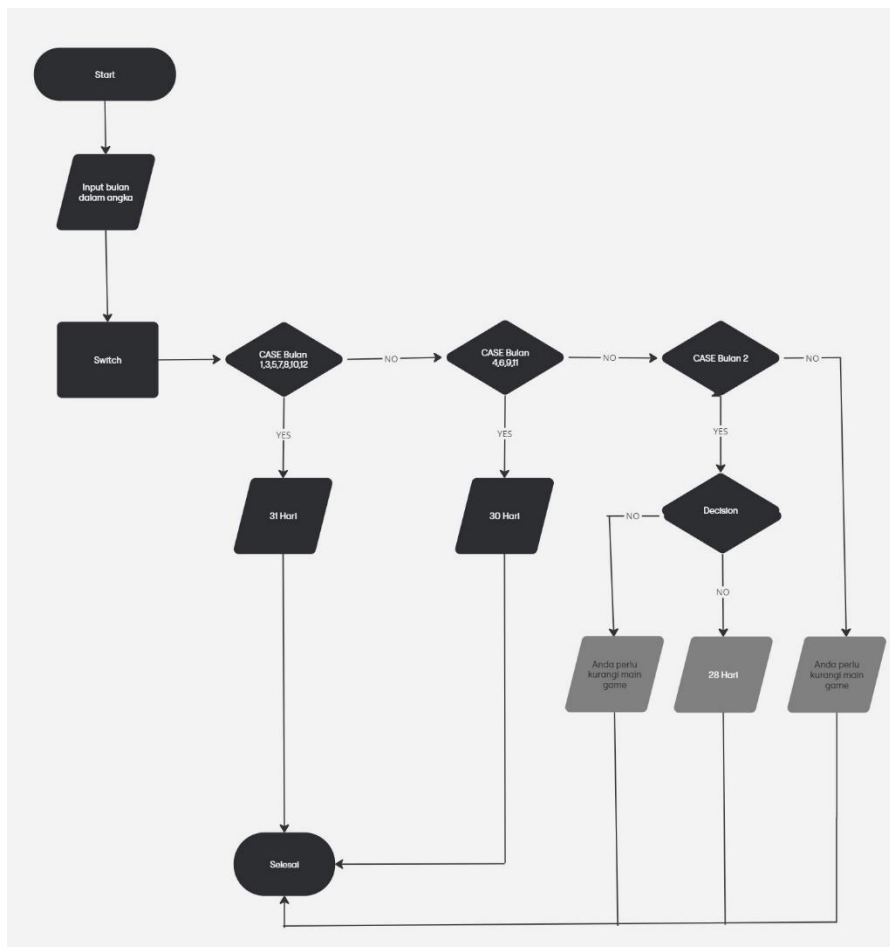
Perbandingan antara if dan switch menunjukkan beberapa perbedaan penting. Dari segi keterbacaan, switch lebih mudah dipahami ketika ada banyak pilihan, karena strukturnya yang teratur dengan case. Sebaliknya, penggunaan if dengan banyak kondisi dapat menjadi rumit dan panjang karena memerlukan banyak else if. Dalam hal kinerja, switch dapat lebih efisien karena compiler dapat mengoptimalkan pemetaan tabel, sedangkan jika mungkin lebih lambat dalam situasi dengan banyak kondisi, karena harus memeriksa setiap kondisi satu **per satu**. Namun, jika menawarkan lebih dari itu, dapat menangani kondisi yang lebih kompleks seperti jarak nilai dan kombinasi logika (AND, OR), sementara switch terbatas pada pemeriksaan kesetaraan nilai untuk variabel yang sama.

2.4

flowchart latihan 2.1



Flowchart latihan 2.2



[No.2] Penyusunan Algoritma dan Kode Program

4) Algoritma contoh 3

- Masuk ke eclipse
- Mulai kelas
- Masukkan kode program yang telah dibuat
- Run kan kode
- Masukkan pilihan A di output selanjutnya enterkan
- Selesai

5) Algoritma contoh 4

- Masuk ke eclipse
- Mulai
- Masukkan kode program
- Tambah bulan 5- 12 sesuai perintah
- Run kan hasil

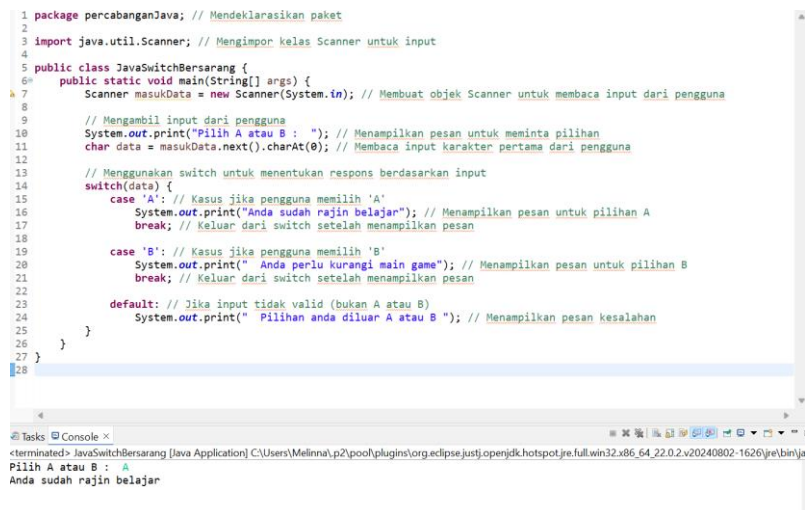
(f) Selesai

6) Kode program dan luaran

a) Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

GAMBAR INPUT DAN OUTPUT PADA CONTOH 3



```
1 package percabanganJava; // Mendeklarasikan paket
2
3 import java.util.Scanner; // Mengimpor kelas Scanner untuk input
4
5 public class JavaSwitchBersarang {
6     public static void main(String[] args) {
7         Scanner masukData = new Scanner(System.in); // Membuat objek Scanner untuk membaca input dari pengguna
8
9         // Mengambil input dari pengguna
10        System.out.print("Pilih A atau B : "); // Menampilkan pesan untuk meminta pilihan
11        char data = masukData.next().charAt(0); // Membaca input karakter pertama dari pengguna
12
13        // Menggunakan switch untuk menentukan respons berdasarkan input
14        switch(data) {
15            case 'A': // Kasus jika pengguna memilih 'A'
16                System.out.print("Anda sudah rajin belajar"); // Menampilkan pesan untuk pilihan A
17                break; // Keluar dari switch setelah menampilkan pesan
18
19            case 'B': // Kasus jika pengguna memilih 'B'
20                System.out.print("Anda perlu kurangi main game"); // Menampilkan pesan untuk pilihan B
21                break; // Keluar dari switch setelah menampilkan pesan
22
23            default: // Jika input tidak valid (bukan A atau B)
24                System.out.print("Pilihan anda diluar A atau B"); // Menampilkan pesan kesalahan
25        }
26    }
27 }
28
```

Tasks Console x
<terminated> JavaSwitchBersarang [Java Application] C:\Users\Melinnal\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\ja
Pilih A atau B : A
Anda sudah rajin belajar

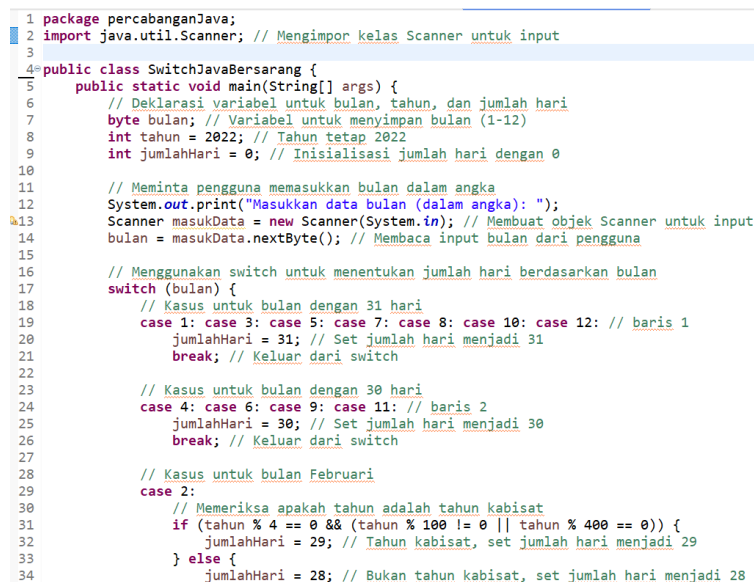
b) Analisa luaran yang dihasilkan

jawab:

Luaran sudah sesuai dengan program yang disusun.

Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data.

Gambar input dan output contoh 4



```
1 package percabanganJava;
2 import java.util.Scanner; // Mengimpor kelas Scanner untuk input
3
4 public class SwitchJavaBersarang {
5     public static void main(String[] args) {
6         // Deklarasi variabel untuk bulan, tahun, dan jumlah hari
7         byte bulan; // Variabel untuk menyimpan bulan (1-12)
8         int tahun = 2022; // Tahun tetap 2022
9         int jumlahHari = 0; // Inisialisasi jumlah hari dengan 0
10
11        // Meminta pengguna memasukkan bulan dalam angka
12        System.out.print("Masukkan data bulan (dalam angka): ");
13        Scanner masukData = new Scanner(System.in); // Membuat objek Scanner untuk input
14        bulan = masukData.nextByte(); // Membaca input bulan dari pengguna
15
16        // Menggunakan switch untuk menentukan jumlah hari berdasarkan bulan
17        switch (bulan) {
18            // Kasus untuk bulan dengan 31 hari
19            case 1: case 3: case 5: case 7: case 8: case 10: case 12: // baris 1
20                jumlahHari = 31; // Set jumlah hari menjadi 31
21                break; // Keluar dari switch
22
23            // Kasus untuk bulan dengan 30 hari
24            case 4: case 6: case 9: case 11: // baris 2
25                jumlahHari = 30; // Set jumlah hari menjadi 30
26                break; // Keluar dari switch
27
28            // Kasus untuk bulan Februari
29            case 2:
30                // Memeriksa apakah tahun adalah tahun kabisat
31                if (tahun % 4 == 0 && (tahun % 100 != 0 || tahun % 400 == 0)) {
32                    jumlahHari = 29; // Tahun kabisat, set jumlah hari menjadi 29
33                } else {
34                    jumlahHari = 28; // Bukan tahun kabisat, set jumlah hari menjadi 28
35                }
36            }
37    }
38 }
```

```
35         }
36         break; // Keluar dari switch
37
38         // Jika bulan tidak valid (lebih dari 12)
39         default:
40             System.out.println("Maaf bulan hanya sampai 12."); // Pesan kesalahan untuk bulan tidak valid
41             break; // Keluar dari switch
42     }
43
44     // Menampilkan jumlah hari
45     System.out.println("Jumlah hari = " + jumlahHari); // Mencetak hasil jumlah hari
46 }
47 }
```

Tasks Console ×

<terminated> SwitchJavaBersarang [Java Application] C:\Users\Melinda\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\bin\

Masukkan data bulan (dalam angka): 9

Jumlah hari = 30

c) Analisa luaran yang dihasilkan

jawab:

Luaran sudah sesuai dengan program yang disusun.

Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data.

[No.2]Kesimpulan

Analisa pada contoh 3

a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!

Jawab:

Program JavaSwitchBersarang dirancang untuk meminta pengguna memilih antara dua opsi, yaitu 'A' atau 'B', dan memberikan respons yang sesuai berdasarkan pilihan tersebut. Dalam implementasinya, program menggunakan struktur percabangan switch untuk mengevaluasi input dan memberikan feedback yang relevan. Jika pengguna memilih 'A', program menginformasikan bahwa mereka sudah rajin belajar; jika memilih 'B', program menyarankan untuk mengurangi waktu bermain game. Jika input tidak valid, program menampilkan pesan kesalahan. Algoritma yang digunakan mencakup langkah-langkah untuk mengambil input pengguna, memeriksa input dengan switch, dan memberikan output berdasarkan kondisi yang terpenuhi. Dengan pengelolaan yang tepat melalui penggunaan break, program menghindari eksekusi kode yang tidak diinginkan setelah menemukan kasus yang sesuai.

b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawab:

Penggunaan struktur switch dalam kasus ini didasarkan pada kebutuhan untuk memeriksa satu variabel (yaitu karakter input) terhadap beberapa kemungkinan nilai yang telah ditentukan (yaitu 'A' dan 'B'). Switch memberikan cara yang lebih terstruktur dan mudah dibaca daripada menggunakan serangkaian if-else if, terutama ketika jumlah pilihan semakin banyak. Ini memudahkan pemeliharaan kode dan meningkatkan keterbacaan.

Selain itu, dengan menambahkan default, program memberikan penanganan untuk situasi di mana input tidak sesuai dengan yang diharapkan, sehingga meningkatkan ketahanan program. Keputusan untuk menggunakan objek Scanner di awal untuk mengambil input pengguna juga menunjukkan praktik yang baik dalam pengelolaan input.

Analisa pada contoh 4

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!

Jawab:

Program SwitchJavaBersarang dirancang untuk menentukan jumlah hari dalam sebuah bulan berdasarkan input pengguna. Dengan menggunakan struktur switch, program mengevaluasi nilai bulan yang dimasukkan dan mengatur jumlah hari yang sesuai. Jika bulan adalah Januari, Maret, Mei, Juli, Agustus, Oktober, atau Desember, jumlah harinya diatur menjadi 31. Untuk bulan April, Juni, September, dan November, jumlah harinya adalah 30. Sementara itu, untuk bulan Februari, program memeriksa apakah tahun tersebut merupakan tahun kabisat untuk menentukan apakah jumlah harinya 28 atau 29. Jika pengguna memasukkan angka yang tidak valid (lebih dari 12), program menampilkan pesan kesalahan. Dengan pendekatan ini, program berhasil menyelesaikan permasalahan terkait penentuan jumlah hari dalam bulan.

- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawab:

Penggunaan struktur switch dalam program ini dipilih karena kemampuannya untuk mengelola banyak kasus dengan cara yang lebih terstruktur dan mudah dibaca dibandingkan dengan serangkaian pernyataan if-else. Dengan mengelompokkan bulan berdasarkan jumlah hari, program menjadi lebih jelas dan efisien. Penambahan logika untuk memeriksa tahun kabisat di dalam case untuk bulan Februari menunjukkan perhatian terhadap detail dan akurasi. Selain itu, penggunaan objek Scanner untuk mengambil input pengguna di awal memastikan bahwa semua input dikelola secara efisien. Semua keputusan ini bertujuan untuk meningkatkan keterbacaan dan efisiensi kode, serta memberikan pengalaman pengguna yang lebih baik.

Refleksi

Pada tugas ke empat ini tentang if dan switch saya banyak belajar apa itu if, if else, if switch, kegunaan break pada kode program, menggunakan default, if bersarang, cara membuat flowchart dari web dengan menggunakan kode program yang telah saya susun sebelumnya dan lain lain.

