

| Nama & NPM | Topik: | Tanggal: |
|--|-----------|-----------------|
| Farrel Alvaro Alinskie .M G1F024024 | Unit 1 IF | 10 oktober 2024 |

[No. 1] Identifikasi Masalah:

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!

- a. $\text{IF} (\text{nilaiU} \geq 80 \mid \mid \text{nilaiT} \geq 80 \mid \mid \text{nilaiQ} \geq 80)$
- b. $\text{IF} (\text{nilaiU} \geq 80 \mid \mid \text{nilaiT} \geq 80 \ \&\& \text{nilaiQ} \geq 80)$
- c. $\text{IF} (\text{nilaiU} \geq 80 \ \&\& \text{nilaiT} \geq 80 \mid \mid \text{nilaiQ} \geq 80)$
- d. $\text{IF} (\text{nilaiU} \geq 80 \ \&\& \text{nilaiT} \geq 80 \ \&\& \text{nilaiQ} \geq 80)$

1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!

[No.1]jawaban

1.1 program

```
1- import java.util.Scanner;
2
3- public class BilanganGenapGanjil {
4-     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6         System.out.print("Masukkan Angka: ");
7         int angka = input.nextInt();
8
9-         if (angka % 2 == 0) {
10             System.out.println("Angka " + angka + " adalah bilangan
                genap.");
11-         } else {
12             System.out.println("Angka " + angka + " adalah bilangan
                ganjil.");
13         }
14     }
15 }
16
```

Output:

```
Masukkan Angka: 2
Angka 2 adalah bilangan genap.
```

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

1.2 program yang menambahkan if

```
4 public static void main(String[] args) {
5     Scanner input = new Scanner(System.in);
6
7     System.out.print("Masukkan Angka UTS Anda: ");
8     int nilaiU = input.nextByte();
9
10    System.out.print("Masukkan Angka Tugas Anda: ");
11    int nilaiT = input.nextByte();
12
13    System.out.print("Masukkan Angka Quiz Anda: ");
14    int nilaiQ = input.nextByte();
15
16    if (nilaiU >= 80) {
17        if (nilaiT >= 80) {
18            if (nilaiQ >= 80) {
19                System.out.println("Anda mendapatkan nilai A");
20            } else {
21                System.out.println("Nilai Quiz Anda kurang dari
22                80, Anda TIDAK mendapatkan nilai A.");
23            }
24        } else {
25            System.out.println("Nilai Tugas Anda kurang dari 80
26            Anda TIDAK mendapatkan nilai A.");
27        }
28    } else {
29        System.out.println("Nilai UTS Anda kurang dari 80, Anda
30        TIDAK mendapatkan nilai A.");
31    }
32 }
```

Output:

```
Masukkan Angka UTS Anda: 80
Masukkan Angka Tugas Anda: 80
Masukkan Angka Quiz Anda: 80
Anda mendapatkan nilai A
```

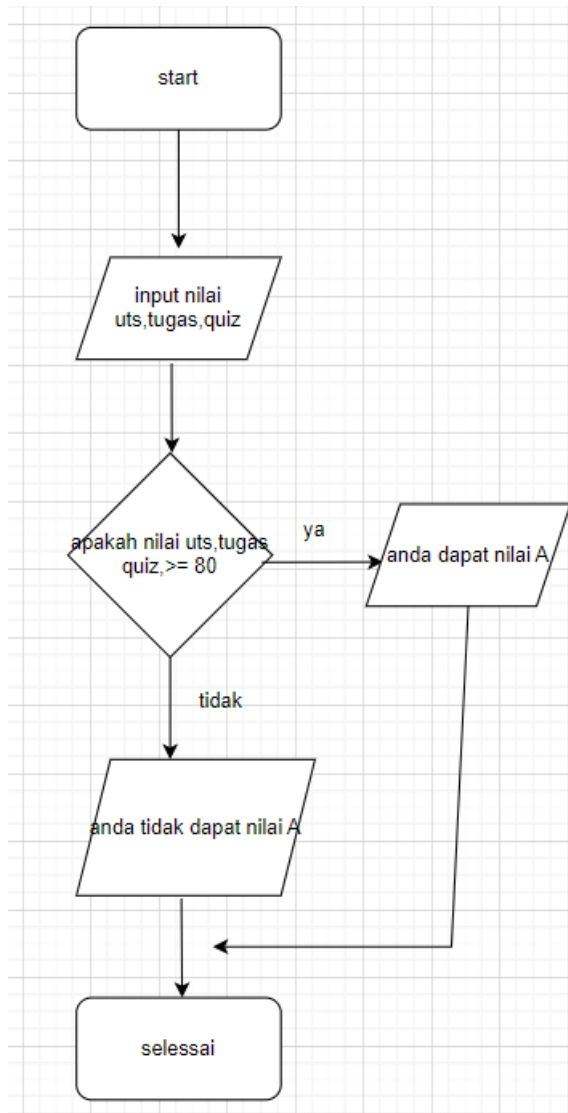
Kondisi bersarang ini memastikan bahwa **semua kondisi** harus terpenuhi untuk mendapatkan nilai A. Artinya, hanya jika nilai UTS, Tugas, dan Quiz semuanya ≥ 80 , siswa akan mendapatkan nilai A.

1.3 Ya, ketiga kondisi IF dari contoh di atas bisa diringkas menjadi satu kondisi dengan menggunakan operator logika && (dan) yang menyatakan bahwa semua kondisi harus benar.

Untuk pertanyaan Anda tentang mana kondisi yang tepat untuk menggantikan tiga kondisi tersebut, jawabannya adalah **d. IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)**.

Kondisi ini menyatakan bahwa **semua** dari nilaiU, nilaiT, dan nilaiQ harus lebih besar atau sama dengan 80 untuk memberikan hasil "nilai A". Operator && memastikan bahwa ketiga syarat harus terpenuhi.

Pilihan lain (a, b, c) tidak tepat karena operator || (atau) hanya memerlukan salah satu kondisi benar agar pernyataan menjadi benar, sehingga hasilnya tidak sesuai dengan maksud asli.



1.4

| Nama & NPM | Topik: | Tanggal: |
|--|---------------|-----------------|
| Farrel Alvaro Alinskie .M G1F024024 | Unit 2 switch | 10 oktober 2024 |

[No. 2] Identifikasi Masalah:

```
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]jawaban

2.1. Jika break; pada baris 1 dihapus:

```
java
```

```
case 'A':
```

```
    System.out.print("Anda sudah rajin belajar");
```

```
// break; // Dihapus
```

Jika Anda menghapus break; pada baris 1, ketika pengguna memilih A, kode akan terus dieksekusi hingga menemukan break; berikutnya. Jadi, hasilnya akan menampilkan dua pesan:

```
less
```

Pilih A atau B : A

Anda sudah rajin belajar Anda perlu kurangi main game

Hal ini terjadi karena tanpa break;, aliran program tidak berhenti setelah case 'A' dan terus melanjutkan ke case 'B', sampai menemukan break;.

2. Jika break; pada baris 2 dihapus:

```
java
```

```
case 'B':
```

```
System.out.print(" Anda perlu kurangi main game");  
// break; // Dihapus
```

Jika Anda menghapus break; pada baris 2, ketika pengguna memilih B, setelah menampilkan pesan untuk case 'B', eksekusi tetap berlanjut, tetapi tidak ada case lain setelahnya. Jadi, output tetap sama:
less

Pilih A atau B : B

Anda perlu kurangi main game

Kegunaan break: Statement break; digunakan untuk menghentikan eksekusi lebih lanjut dari case lain setelah satu case dieksekusi. Tanpa break;, semua case setelah case yang sesuai akan dieksekusi sampai ditemukan break;.

Kegunaan default: Kata kunci default digunakan untuk menangani situasi ketika tidak ada case yang cocok dengan nilai yang diberikan. Ini memastikan program memberikan tanggapan yang sesuai untuk input yang tidak valid.

2.2

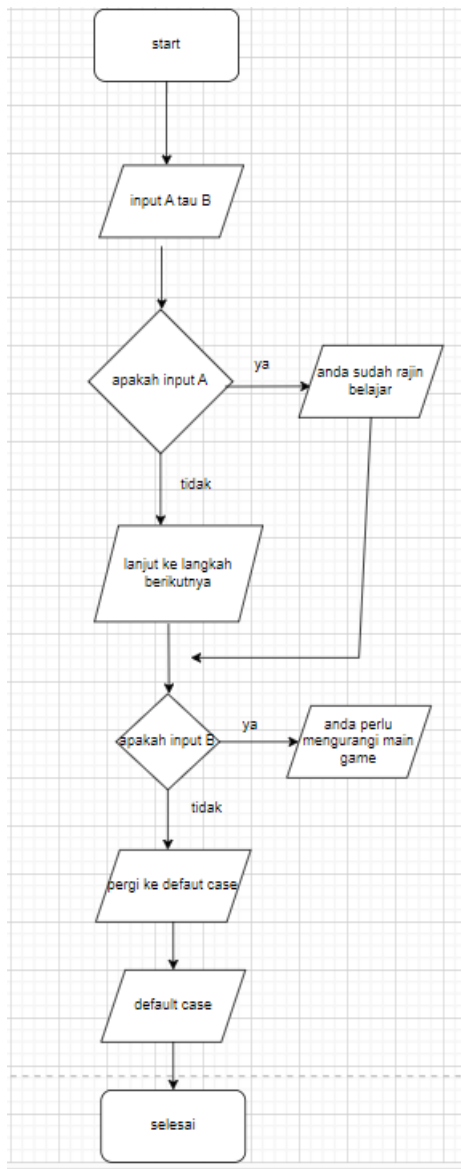
```
1 import java.util.Scanner;
2
3 public class SwitchBersarang {
4     public static void main(String[] args) {
5         byte bulan;
6         int tahun = 2022;
7         int jumlahHari = 0;
8         System.out.print("Masukkan data bulan (dalam angka): ");
9         Scanner masukData = new Scanner(System.in);
10        bulan = masukData.nextByte();
11
12        switch (bulan) {
13            case 1: case 3: case 5: case 7: case 8: case 10: case 12
14                :
15                jumlahHari = 31;
16                break;
17            case 4: case 6: case 9: case 11:
18                jumlahHari = 30;
19                break;
20            case 2:
21                if (tahun % 4 == 0)
22                    jumlahHari = 29;
23                else
24                    jumlahHari = 28;
25                break;
26            default:
27                System.out.println("Maaf bulan hanya sampai 12.");
28                break;
29        }
```

2.3 Analisis IF vs SWITCH:

- **if-else:** Berguna jika kita perlu mengevaluasi ekspresi yang lebih kompleks, misalnya, membandingkan variabel dengan berbagai kondisi yang bukan hanya berbasis nilai tunggal. if-else juga lebih fleksibel dalam menangani kondisi logika yang rumit.
- **switch:** Berguna ketika kita harus mengevaluasi satu variabel terhadap berbagai kemungkinan nilai tetap. Lebih sederhana dan efisien untuk menangani banyak kondisi yang saling eksklusif berbasis nilai tertentu. Namun, tidak sefleksibel if dalam hal ekspresi logika yang kompleks.

2.4

Flowchart dari Latihan 2.1



b.flowchart untuk Latihan 2.2

