

Nama & NPM	Topik:	Tanggal:
Yeni kusherawati G1F024013	IF DAN SWITCH DATA	24 September 2024
[Nomor 1] Identifikasi Masalah:		
<p>1)Uraikan permasalahan dan variabel</p> <p>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) jawab: Berikut adalah penjelasan langkah demi langkah tentang cara kerja kode untuk menentukan apakah sebuah bilangan adalah genap atau ganjil:</p> <p>1. Input Pengguna: - Program mulai dengan meminta pengguna untuk memasukkan sebuah bilangan. Input ini kemudian diubah menjadi tipe data integer (bilangan bulat).</p> <p>2. Operasi Modulus: - Setelah mendapatkan input dari pengguna, program menghitung sisa hasil bagi dari bilangan tersebut dengan 2. Ini dilakukan menggunakan operasi modulus (%).</p> <p>3. Pemeriksaan Kondisi: - Program memeriksa nilai hasil sisa bagi: - Jika sisa bagi sama dengan 0, program mengidentifikasi bilangan tersebut sebagai genap - Jika sisa bagi sama dengan 1, program mengidentifikasi bilangan tersebut sebagai ganjil.</p> <p>4. Output: - Program kemudian mencetak hasil yang menunjukkan apakah bilangan yang dimasukkan adalah genap atau ganjil. Dengan cara ini, pengguna bisa mengetahui dengan cepat kategori bilangan yang mereka masukkan.</p> <p>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.</p> <p>jawab: Berikut adalah penjelasannya:</p> <p>1.Input: Program meminta pengguna memasukkan tiga nilai: UTS (nilaiU), Tugas (nilaiT), dan Quiz (nilaiQ).</p> <p>2.Pemeriksaan Kondisi:</p> <p>Jika nilaiU \geq 80, program memeriksa apakah nilaiT dan nilaiQ juga \geq 80.</p> <p>Jika semua nilai memenuhi syarat, siswa mendapatkan nilai A. Jika tidak, siswa tidak mendapatkan nilai A.</p> <p>3.Menutup Sumber Daya: Program menutup objek scanner setelah selesai.</p> <p>Intinya, program mengevaluasi tiga nilai dan memberikan hasil berdasarkan kondisi yang</p>		

ditentukan.

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)`

jawab

Ketiga kondisi IF tersebut dapat diringkas menjadi satu kondisi yang lebih sederhana. Dari pilihan yang diberikan, berikut adalah analisisnya:

a. `IF (nilaiU >= 80 || nilaiT >= 80 || nilaiQ >= 80)`: Ini memeriksa apakah salah satu nilai memenuhi syarat. Ini tidak sesuai karena kita ingin semua nilai harus ≥ 80 untuk mendapatkan nilai A.

b. `IF (nilaiU >= 80 || nilaiT >= 80 && nilaiQ >= 80)`: Ini memiliki ambiguitas karena operator logika dapat menghasilkan hasil yang tidak diinginkan. Ini juga tidak memenuhi syarat untuk mendapatkan nilai A jika hanya satu nilai yang memenuhi.

c. `IF (nilaiU >= 80 && nilaiT >= 80 || nilaiQ >= 80)`: Ini juga tidak tepat, karena jika `nilaiU` dan `nilaiT` ≥ 80 , tetapi `nilaiQ` tidak, kondisi ini akan terpenuhi.

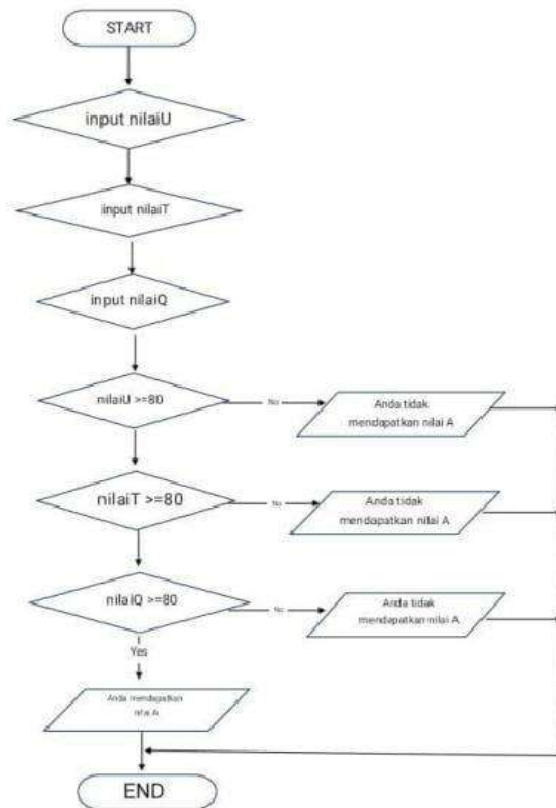
d. `IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)`: Ini adalah kondisi yang tepat. Ketiga nilai harus memenuhi syarat untuk mendapatkan nilai A.

kesimpulan

Jadi, pilihan d (`IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)`) adalah yang paling tepat untuk menggantikan ketiga kondisi itu, karena memastikan bahwa semua nilai harus ≥ 80 untuk mendapatkan nilai A.

1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!

jawab: Uraian flowchart



1. Mulai: Titik awal dari flowchart.
2. Input Nilai:
 - Kotak Input: Minta pengguna untuk memasukkan nilai UTS (*nilaiU*).
 - Kotak Input: Minta pengguna untuk memasukkan nilai Tugas (*nilaiT*).
 - Kotak Input: Minta pengguna untuk memasukkan nilai Quiz (*nilaiQ*).
3. Kondisi Pertama: Periksa apakah *nilaiU* ≥ 80 .
 - Ya: Lanjut ke langkah berikutnya.
 - Tidak: Arahkan ke kotak output yang menyatakan "Anda TIDAK mendapatkan nilai A".
4. Kondisi Kedua: Periksa apakah *nilaiT* ≥ 80 .
 - Ya: Lanjut ke langkah berikutnya.
 - Tidak: Arahkan ke kotak output yang menyatakan "Anda TIDAK mendapatkan nilai A".
5. Kondisi Ketiga: Periksa apakah *nilaiQ* ≥ 80 .
 - Ya: Arahkan ke kotak output yang menyatakan "Anda mendapatkan nilai A".
 - Tidak: Arahkan ke kotak output yang menyatakan "Anda TIDAK mendapatkan nilai A".
6. Akhir: Menunjukkan bahwa proses telah selesai.

Flowchart ini menggambarkan alur logika program, mulai dari pengambilan input nilai hingga keputusan akhir berdasarkan syarat yang ditetapkan. Setiap kondisi mengarah ke hasil yang berbeda, memungkinkan pengguna memahami dengan jelas proses pengambilan keputusan dalam program.

2).Rincikan sumber informasi yang relevan (buku / webpage)

video Materi 1 tentang IF – <https://www.youtube.com/watch?v=G0dfdAFa9iM>

[Nomor 1] Analisis dan Argumentasi

1).Uraikan rancangan solusi yang diusulkan.

jawab:Tujuan: Menentukan apakah siswa mendapatkan nilai A berdasarkan nilai UTS, Tugas, dan Quiz.

Langkah:

1. Minta input nilai UTS, Tugas, dan Quiz.
2. Periksa apakah ketiga nilai tersebut ≥ 80 .
3. Jika semua terpenuhi, tampilkan "Anda mendapatkan nilai A"; jika tidak, tampilkan "Anda TIDAK mendapatkan nilai A".
4. Tutup sumber daya input.

2).Analisis solusi, kaitkan dengan permasalahan.

Analisis: Solusi efektif dan mudah diperluas. Struktur kontrol `if` memastikan akurasi dalam evaluasi nilai.

Kaitan dengan Permasalahan: Solusi ini menyelesaikan tantangan menentukan kelulusan dengan jelas, berdasarkan kriteria yang telah ditetapkan.

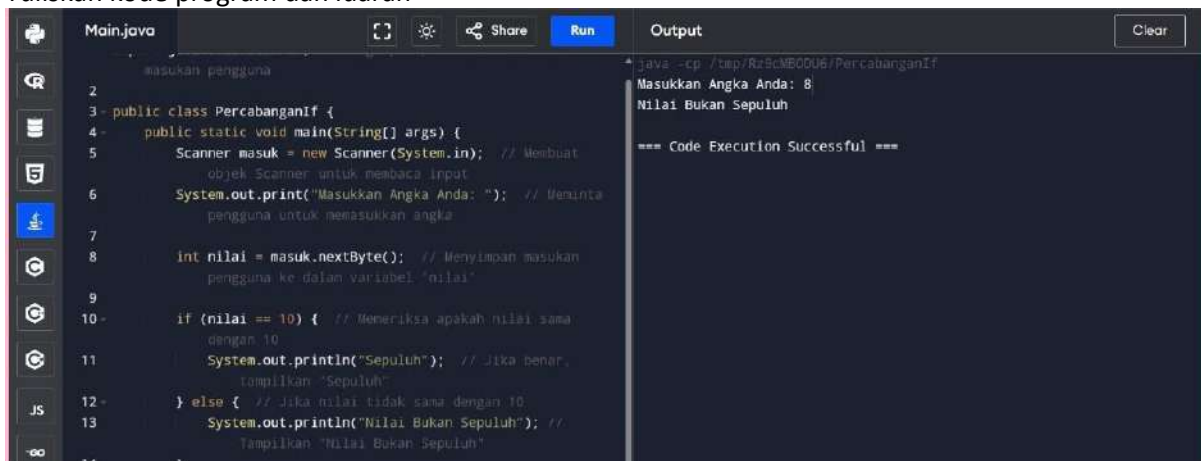
[Nomor 1] Penyusunan Algoritma dan Kode Program

1.1)Rancang desain solusi atau algoritma

Algoritma

- a. Mulai
- b. Buat nama kelas 'PercabanganIf'
- c. Buat method dengan parameter 'public static void main(String[] args)'
- d. Buat kode 'Scanner masuk = new Scanner(System.in):
- e. Buat kode print string 'Masukkan Angka Anda : '
- f. Buat kode input dengan tipe data integer 'int nilai dan int hitung'
- g. Buat kode print untuk menampilkan output
- h. Akhiri program

Tuliskan kode program dan luaran



```
1 masukan pengguna
2
3 public class PercabanganIf {
4     public static void main(String[] args) {
5         Scanner masuk = new Scanner(System.in); // Membuat
6         // objek Scanner untuk membaca input
7         System.out.print("Masukkan Angka Anda : "); // Meminta
8         // pengguna untuk memasukkan angka
9
10        int nilai = masuk.nextByte(); // Menyimpan masukan
11        // pengguna ke dalam variabel 'nilai'
12
13        if (nilai == 10) { // Memeriksa apakah nilai sama
14            // dengan 10
15            System.out.println("Sepuluh"); // Jika benar,
16            // tampilkan "Sepuluh"
17        } else { // Jika nilai tidak sama dengan 10
18            System.out.println("Nilai Bukan Sepuluh"); //
19            // Tampilkan "Nilai Bukan Sepuluh"
20        }
21    }
22 }
```

Output

```
java -cp /tmp/RzScNB0DU6/PercabanganIf
Masukkan Angka Anda: 8
Nilai Bukan Sepuluh

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

Komentar pada kode ini

Scanner: Digunakan untuk menerima input dari pengguna.

nextByte(): Membaca input sebagai byte (batasan -128 hingga 127), bisa diganti nextInt() agar lebih fleksibel.

if-else: Mengecek apakah input sama dengan 10, lalu menampilkan "Sepuluh" jika benar, atau "Nilai Bukan Sepuluh" jika salah.

1.2)Rancang desain solusi atau algoritma

Algoritma

- a. Mulai
- b. Buat nama kelas 'IfBersarang'
- c. Buat method dengan parameter 'public static void main(String[] args)'
- d. Buat kode 'Scanner var() = new Scanner(System.in);'
- e. Buat kode print string 'Masukkan Angka UTS anda : '
- f. Buat kode input dengan tipe data integer 'int nilaiU = varU.nextByte();'
- g. Buat kode 'Scanner var() = new Scanner(System.in);'
- h. Buat kode print string 'Masukkan Angka Tugas anda : '
- i. Buat kode input dengan tipe data integer 'int nilaiT = varT.nextByte();'
- j. Buat kode 'Scanner var() = new Scanner(System.in);'
- k. Buat kode print string 'Masukkan Angka Quiz anda : '
- l. Buat kode input dengan tipe data integer 'int nilaiQ = varQ.nextByte();'
- m. Buat kondisi if else, dengan if 'nilaiU, nilaiT, dan nilaiQ>=80' dengan print string 'Mendapatkan nilai A' dan else 'Tidak mendapatkan nilai A'
- n. Buat kode print untuk menampilkan output
- o. Akhiri program

Kode program dan luaran

```

1- import java.util.Scanner;
2-
3- public class IfBersarang {
4-     public static void main(String[] args) {
5-         Scanner varU = new Scanner(System.in);
6-         System.out.print("Masukkan Angka UTS Anda : ");
7-         int nilaiU = varU.nextByte();
8-
9-         Scanner varT = new Scanner(System.in);
10-        System.out.print("Masukkan Angka Tugas Anda : ");
11-        int nilaiT = varT.nextByte();
12-
13-        Scanner varQ = new Scanner(System.in);
14-        System.out.print("Masukkan Angka 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- }
  
```

Output:

```

java -cp .\bin\main\main.jar IfBersarang
Masukkan Angka UTS Anda : 90
Masukkan Angka Tugas Anda : 90
Masukkan Angka Quiz Anda : 90
Anda mendapatkan nilai A
--- Code Execution Successful ---
  
```

Berikan komentar pada kode ini

s canner: Digunakan untuk menerima tiga input (UTS, tugas, dan quiz) dari pengguna.

nextByte(): Digunakan untuk membaca input angka dalam bentuk byte, yang mungkin lebih baik diganti dengan nextInt() untuk menghindari batasan rentang nilai.

if-else: Mengecek apakah total nilai lebih dari atau sama dengan 80. Jika iya, tampilkan "Anda mendapatkan nilai A". Jika tidak, tampilkan "Anda TIDAK mendapatkan nilai A".

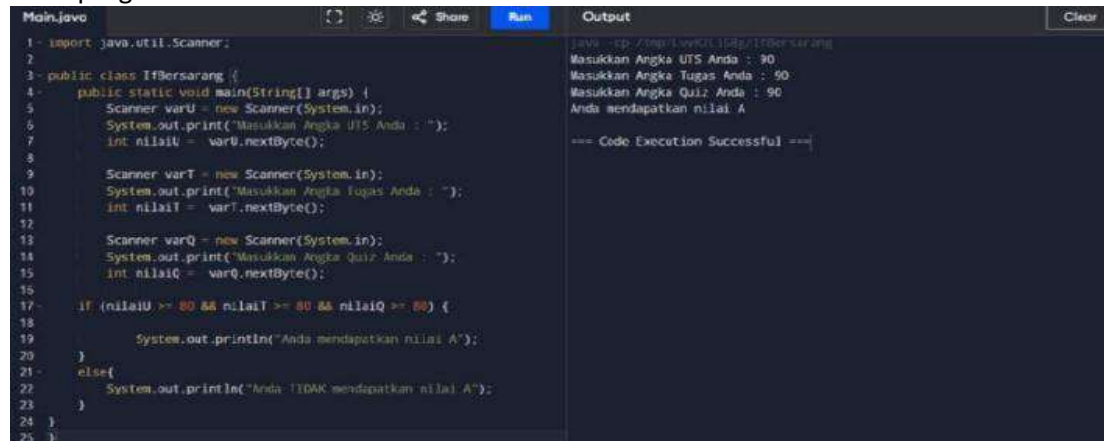
1.3)Rancang desain solusi atau algoritma

Algoritma

- a. Mulai
- b. Buat nama kelas 'IfBersarang'
- c. Buat method dengan parameter 'public static void main(String[] args)'
- d. Buat kode 'Scanner var() = new Scanner(System.in);'
- e. Buat kode print string 'Masukkan Angka UTS anda : '
- f. Buat kode input dengan tipe data integer 'int nilaiU = varU.nextByte();'
- g. Buat kode 'Scanner var() = new Scanner(System.in);'
- h. Buat kode print string 'Masukkan Angka Tugas anda : '
- i. Buat kode input dengan tipe data integer 'int nilaiT =

- varT.nextByte());'
 j. Buat kode 'Scanner var() = new Scanner(System.in):'
 k. Buat kode print string 'Masukkan Angka Quiz anda : '
 l. Buat kode input dengan tipe data integer 'int nilaiQ = varQ.nextByte());'
 m. Buat kondisi if else, dengan if 'nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80' dengan print string 'Mendapatkan nilai A' dan else 'Tidak mendapatkan nilai A'
 n. Buat kode print untuk menampilkan output
 o. Akhiri program

Kode program dan luaran



```

Main.java
1 import java.util.Scanner;
2
3 public class IfBersarang {
4     public static void main(String[] args) {
5         Scanner varU = new Scanner(System.in);
6         System.out.print("Masukkan Angka UTS Anda : ");
7         int nilaiU = varU.nextByte();
8
9         Scanner varT = new Scanner(System.in);
10        System.out.print("Masukkan Angka Tugas Anda : ");
11        int nilaiT = varT.nextByte();
12
13        Scanner varQ = new Scanner(System.in);
14        System.out.print("Masukkan Angka Quiz Anda : ");
15        int nilaiQ = varQ.nextByte();
16
17        if (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80) {
18            System.out.println("Anda mendapatkan nilai A");
19        }
20        else{
21            System.out.println("Anda TIDAK mendapatkan nilai A");
22        }
23    }
24 }
25 }

Output
Java - cp /tmp/look111Gqz/IfBersarang
Masukkan Angka UTS Anda : 80
Masukkan Angka Tugas Anda : 90
Masukkan Angka Quiz Anda : 90
Anda mendapatkan nilai A
--- Code Execution Successful ---
  
```

Beri komentar pada kode

Scanner: Program menerima tiga input dari pengguna (UTS, tugas, dan quiz) menggunakan objek Scanner.
 nextByte(): Digunakan untuk membaca input, namun disarankan menggunakan nextInt() agar lebih fleksibel menangani angka yang lebih besar.
 if-else: Kondisi if memeriksa apakah nilai UTS, tugas, dan quiz semuanya ≥ 80 . Jika ya, program menampilkan "Anda mendapatkan nilai A". Jika tidak, tampilkan "Anda TIDAK mendapatkan nilai A".

[Nomor 1] Kesimpulan

1) Analisa

Program ini berhasil memberikan solusi yang sederhana dan efektif untuk menentukan apakah seorang siswa mendapatkan nilai A berdasarkan nilai UTS, Tugas, dan Quiz. Dengan langkah-langkah yang jelas, pengguna dapat dengan mudah mengikuti proses input dan mendapatkan hasilnya.

1. Mudah Dipahami: Struktur programnya sangat straightforward, sehingga siapa saja bisa mengerti cara kerjanya. Pengguna hanya perlu memasukkan nilai, dan program langsung memberikan hasil yang jelas.
2. Fleksibel: Salah satu hal bagus dari program ini adalah kemampuannya untuk ditambahkan fitur baru. Misalnya, jika nanti ingin menambah kriteria penilaian lain, itu bisa dilakukan tanpa mengubah banyak bagian dari kode.
3. Pengelolaan Sumber Daya: Program juga memperhatikan pengelolaan sumber daya dengan baik. Setelah selesai, objek input ditutup, yang penting untuk menjaga

performa aplikasi.

4. Bermanfaat dalam Pendidikan: Ini sangat relevan untuk dunia pendidikan. Baik guru maupun siswa bisa menggunakan program ini untuk mengevaluasi pencapaian akademik dengan cepat dan mudah.
5. Peluang Pengembangan: Program ini bisa dikembangkan lebih lanjut, misalnya dengan menambahkan fitur untuk menyimpan hasil atau analisis lebih mendala

Nama & NPM	Topik:	Tanggal:
Yeni kusherawati G1F024013	IF DAN SWITCH DATA	24 September 2024
[Nomor 2] Identifikasi Masalah:		
<p>1)Uraikan permasalahan dan variable</p> <p>2.1. Cermati kode pada Contoh 3. Hapuslah kode <code>break</code>; pada //baris 1, eksekusi kembali. Kemudian hapuslah kode <code>break</code>; pada //baris 2, eksekusi kembali. Analisis perbedaan hasil luaran ketika kode menggunakan <code>break</code>, ketika kode <code>break</code> baris 1 dihapus, dan ketika kode <code>break</code> baris 2 dihapus! Analisa kegunaan baris kode dengan <code>break</code> dan kata kunci <code>default</code>!</p> <p>jawab: Analisis Kode dan Pengaruh Penghapusan <code>break</code></p> <p>1. Dengan <code>`break`</code>:</p> <ul style="list-style-type: none"> - Input 'A': Output "Anda sudah rajin belajar". - Input 'B': Output "Anda perlu kurangi main game". - Input lain: Output "Pilihan anda diluar A atau B". <p>2.Tanpa <code>`break`</code> pada Baris 1:</p> <ul style="list-style-type: none"> - Input 'A': Output "Anda sudah rajin belajar" diikuti oleh "Anda perlu kurangi main game" karena program melanjutkan ke case berikutnya. <p>3.Tanpa <code>`break`</code> pada Baris 2:</p> <ul style="list-style-type: none"> - Input 'B': Output "Anda perlu kurangi main game" diikuti oleh "Pilihan anda diluar A atau B" karena program melanjutkan ke <code>`default`</code>. <p>Kegunaan <code>`break`</code> dan <code>`default`</code></p> <ul style="list-style-type: none"> - <code>`break`</code>: Menghentikan eksekusi dari blok <code>`switch`</code> setelah case yang cocok, mencegah fall-through. - <code>`default`</code>: Menangani input yang tidak valid dengan memberikan pesan kepada pengguna. <p>Kesimpulan:</p> <p><code>`break`</code> penting untuk mengontrol alur program, sedangkan <code>`default`</code> berguna untuk menangani input yang tidak sesuai. Tanpa <code>`break`</code>, hasilnya bisa membingungkan.</p> <p>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.</p> <p>jawab:</p>		

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` !

jawab:

Masalah: Program ingin memberikan umpan balik berdasarkan input pengguna ('A' atau 'B').
Jika input tidak valid, program memberikan pesan kesalahan.

Pengubahan menjadi pernyataan `If`: Ya, masalah ini bisa diubah menjadi pernyataan `if`.
Masalah ini dapat dengan mudah diubah menjadi pernyataan `if` , menggunakan logika yang sama untuk memeriksa input.

Perbandingan `if` dan `switch`

IF: Fleksibel untuk kondisi kompleks dan ekspresi logika.

Mudah ditambah kondisi baru tanpa mengubah struktur utama.

SWITCH:

Ideal untuk memeriksa satu variabel dengan banyak nilai tetap.

Lebih terstruktur dan lebih mudah dibaca untuk kasus sederhana.

2.4. Buatlah dokumentasi gambar flowchart dari Latihan 2.1. dan Latihan 2.2

2). Rincikan sumber informasi yang relevan (buku / webpage)

Video Materi 2 tentang SWITCH – <https://www.youtube.com/watch?v=RB4nz4xkisM>

[Nomor 2] Analisis dan Argumentasi

1). Uraikan rancangan solusi yang diusulkan.

Tujuan: Program memberikan umpan balik berdasarkan pilihan pengguna ('A' atau 'B') dan menangani input yang tidak valid.

Langkah-langkah:

- Minta pengguna memilih antara 'A' atau 'B'.
- Evaluasi input menggunakan `if` atau `switch`:
- Jika 'A', cetak "Anda sudah rajin belajar".

- Jika 'B', cetak "Anda perlu kurangi main game".
- Jika tidak valid, cetak "Pilihan anda diluar A atau B".
- Tampilkan hasil dan tutup objek input.

2).Analisis solusi, kaitkan dengan permasalahan.

Analisis:

Solusi ini efektif dan jelas dalam memberikan umpan balik berdasarkan input pengguna.

Kaitan dengan Permasalahan:

Program langsung menangani permasalahan interaksi pengguna dengan respons yang sesuai, memastikan komunikasi yang baik dan akurat.

Kesimpulan:

Solusi ini sederhana, efisien, dan mudah dimodifikasi, serta secara efektif menyelesaikan masalah yang dihadapi.

[Nomor 2] Penyusunan Algoritma dan Kode Program

1.1)Rancang desain solusi atau algoritma

ALGORITMA

- Mulai.
- Input: Minta pengguna memilih antara 'A' atau 'B'.
- Evaluasi Input
- Jika input == 'A':
Cetak "Anda sudah rajin belajar".
- Jika input == 'B':
Cetak "Anda perlu kurangi main game"
- Jika input tidak valid:
Cetak "Pilihan anda diluar A atau B".
- Akhir.

2).Tuliskan kode program dan luaran

```

Main.java
1: import java.util.Scanner;
2:
3: public class SwitchElersarang {
4:     public static void main(String[] args) {
5:         Scanner masukData = new Scanner(System.in);
6:         // mengambill input
7:         System.out.print("Pilih A atau B : ");
8:         char data = masukData.next().charAt(0);
9:         switch(data) {
10:             case 'A':
11:                 System.out.print("Anda sudah rajin belajar");
12:                 // break 1
13:             case 'B':
14:                 System.out.print(" Anda perlu kurangi main game");
15:                 break; // break 2
16:             default:
17:                 System.out.print(" Pilihan anda diluar A atau B ");
18:         }
19:     }
20: }

```

Output

```

C:\Users\Aldi V\OneDrive\Documents> java -cp . /tmp/NT Aldi V\JN/SwitchElersarang
Pilih A atau B : A
Anda sudah rajin belajar / Anda perlu kurangi main game
=== Code Execution Successful ===

```

```

Main.java
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-         switch(data) {
10-             case 'A':
11-                 System.out.print("Anda sudah rajin belajar!");
12-                 // hari 1
13-             case 'B':
14-                 System.out.print("Anda perlu kurang! main game!");
15-                 // hari 2
16-             default:
17-                 System.out.print("Pilihan anda diluar A atau B");
18-         }
19-     }
20- }

```

```

Output
java -cp /tmp/WT-6MAGQV8/SwitchBersarang
Pilih A atau B : A
Anda sudah rajin belajar! Anda perlu kurang! main game! Pilihan anda diluar
A atau B
=== Code Execution Successful ===

```

Berikan komentar pada kode ini

Perbaikan yang Mungkin:

Validasi Input: Saat ini, program hanya mempertimbangkan huruf besar 'A' dan 'B'. Untuk menangani input huruf kecil ('a' atau 'b'), program bisa mengonversi input ke huruf besar menggunakan `Character.toUpperCase(data)` atau menambahkan case tambahan untuk huruf kecil.

Pesan Pengguna: Pesan prompt dapat diperluas agar lebih jelas, seperti: "Silakan masukkan 'A' atau 'B': ".

Kode ini sederhana dan cukup baik sebagai contoh dasar dari penggunaan pernyataan switch di Java.

1.2)Rancang desain solusi atau algoritma

ALGORITMA

- Mulai
- Buat nama kelas 'SwitchBersarang'
- Buat method utama dengan parameter 'public static void main(String[] args)'
- Buat kode byte bulan
- Buat baris kode tahun dengan tipe data integer
- Buat baris kode jumlahhari dengan tipe data integer
- Buat print String 'Masukkan data bulan (dalam angka): '
- Buat kode 'Scanner masukData = new Scanner(System.in);'
- Buat kode 'bulan = masukData.nextByte[]':
- Buat switch bulan
- Buat kode case, jumlahhari, dan kode break
- Buat kode if else pada case 2
- Buat kode dengan print 'masukkan tahun,masukkan data bulan (dalam angka):'
- Akhiri program.

Tuliskan kode program dan luaran

```

Main.java
16- System.out.print("Masukkan data bulan (dalam angka): ");
17- );
18- bulan = input.nextByte();
19-
20- switch (bulan) {
21-     case 1: case 3: case 5: case 7: case 8: case 10:
22-         case 12: // 31 hari
23-             jumlahHari = 31;
24-             break;
25-     case 4: case 6: case 9: case 11: // 30 hari
26-         jumlahHari = 30;
27-         break;
28-     case 2: // Februari
29-         if (tahun % 4 == 0 && (tahun % 100 != 0 ||
30-             tahun % 400 == 0)) {
31-             jumlahHari = 29; // Tahun kabisat
32-         } else {
33-             jumlahHari = 28; // Tahun biasa
34-         }
35-     }
36- }

```

```

Output
java -cp /tmp/w0LV6chYvB/JumlahHariDalamBulan
Masukkan tahun: 2024
Masukkan data bulan (dalam angka): 11
Jumlah hari di bulan 11 tahun 2024 = 30
=== Code Execution Successful ===

```

Berikan komentar pada kode ini

Program ini meminta input tahun dan bulan dari user. Berdasarkan input bulan, program akan menggunakan pernyataan switch untuk menentukan jumlah hari di bulan tersebut.

- Untuk bulan dengan 31 hari: Januari (1), Maret (3), Mei (5), Juli (7), Agustus (8), Oktober (10), dan Desember (12).

- Untuk bulan dengan 30 hari: April (4), Juni (6), September (9), dan November (11).

- Februari (2) diperiksa apakah tahun yang dimasukkan adalah tahun kabisat. Jika ya, Februari punya 29 hari, jika tidak, hanya 28 hari.

Output program menunjukkan jumlah hari di bulan yang dimasukkan berdasarkan tahun yang dimasukkan user.

[Nomor 2] Kesimpulan

1)Analisa

Secara keseluruhan, program ini menyelesaikan masalah interaksi pengguna dengan cara yang efisien, jelas, dan mudah dipahami. Dengan struktur yang baik, program ini memberikan hasil yang akurat dan memperbaiki pengalaman pengguna, sekaligus menunjukkan potensi untuk pengembangan lebih lanjut jika diperlukan. Program ini secara efektif memberikan umpan balik berdasarkan input pengguna ('A' atau 'B') dengan cara yang sederhana dan terstruktur.