

**Template Lembar Kerja Individu dan Kelompok**

<b>Nama &amp; NPM</b>	<b>Topik:</b>	<b>Tanggal:</b>
<b>IQBAL FERDINAND PUTRA G1F024039</b>	<b>FOR, WHILE</b>	<b>30 SEPTEMBER 2024</b>
<b>[Nomor Soal] Identifikasi Masalah:</b>		
<ol style="list-style-type: none"><li>1) Uraikan permasalahan dan variabel</li><li>2) Rincikan sumber informasi yang relevan (buku / webpage)</li><li>3) Uraikan rancangan solusi yang diusulkan (jika ada).</li><li>4) Analisis susunan solusi, parameter solusi (jika ada).</li></ol>		
<b>[Nomor Soal] Analisis dan Argumentasi</b>		
<ol style="list-style-type: none"><li>1) Uraikan rancangan solusi yang diusulkan.</li><li>2) Analisis solusi, kaitkan dengan permasalahan.</li></ol>		
<b>[Nomor Soal] Penyusunan Algoritma dan Kode Program</b>		
<ol style="list-style-type: none"><li>1) Rancang desain solusi atau algoritma</li><li>2) Tuliskan kode program dan luaran<ol style="list-style-type: none"><li>a) Beri komentar pada kode</li><li>b) Uraikan luaran yang dihasilkan</li><li>c) Screenshot/ Capture potongan kode dan hasil luaran</li></ol></li></ol>		
<b>[Nomor Soal] Kesimpulan</b>		
<ol style="list-style-type: none"><li>1) Analisa<ol style="list-style-type: none"><li>a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!</li><li>b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?</li></ol></li><li>2) Evaluasi<ol style="list-style-type: none"><li>a) Apa konsekuensi dari skenario pemrograman ini?</li><li>b) Evaluasi input, proses, dan luaran yang dihasilkan! (jika ada)</li></ol></li><li>3) Kreasi<ol style="list-style-type: none"><li>a) Apakah ada pengetahuan baru yang dikembangkan dan konsep baru sebagai usulan solusi?</li><li>b) Konstruksikan hubungan antara variabel yang berbeda dengan konsep yang anda ketahui! (jika ada)</li></ol></li></ol>		

**[No. 1] Identifikasi Masalah:**

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

```
public class ContohFor{
public static void main(String[] args) {
    for (double y <= 15; y = 0; y++) {
        if (y % 2 == 1) {    //kondisi 1
            // baris kode kosong 1
        } else if (y == 8) {    //kondisi 2
            // baris kode kosong 2
        } else
            System.out.println(y + " ");
    } } }
```

**Luaran:**

Exception in thread "main" java.lang.Error: Unresolved compilation problems:

Syntax error on token "<=", = expected

Type mismatch: cannot convert from double to boolean

at ContohFor.main(ContohFor.java:5)

**Contoh 2: Salin dan tempel kode program berikut ke Eclipse.**

```
public class ForBersarang {
    public static void main(String[] args) {
        pertama:
        for( int i = 1; i < 5; i++) {
            kedua:
            for(int j = 1; j < 3; j ++ ) {
                System.out.println("i = " + i + "; j = " +j);
            }
        }
        if ( i == 2) {
            // kode yang hilang
        } } } }
```

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

```
import java.util.Scanner;
public class ForBersarang {
    public static void main(String[] args){
        //Instance Input Scanner
        Scanner input = new Scanner(System.in);
        System.out.print("Masukan Input: ");
        int tinggi = input.nextInt(); //Mendapatkan Input Dari User
        for(int t=tinggi; t>=1; t--){
            //Menghitung Jumlah Tinggi Piramida
            for(int s=tinggi; s>=t; s--){
                //Menghitung Jumlah Spasi per Baris
                System.out.print(" ");
            }
            System.out.println(); //Membuat Baris Baru
        }
    }
}
```

```
}  
}
```

#### **Luaran:**

Masukan Input: 7

```
*  
**  
***  
****  
*****  
*****  
*****
```

#### **Latihan 1**

1.1. Evaluasi penyebab kesalahan dan perbaiki kode pada Contoh 1!

Rekomendasikan kata kunci yang tepat diletakkan pada baris kode yang kosong 1 dan 2 untuk dapat menghasilkan luaran berikut:

Luaran contoh 1:

```
0  
2  
4  
6
```

1.2. Cermati contoh kode 2 pada kode //baris kode kosong.

Rekomendasikan kode yang tepat menggunakan break atau continue terhadap pertama atau kedua agar menghasilkan luaran berikut:

Luaran Contoh 2:

```
i = 1; j = 1  
i = 1; j = 2  
i = 2; j = 1  
i = 2; j = 2
```

1.3. Cermati kode contoh 3. Apabila ingin menghasilkan luaran berikut:

Luaran berbentuk piramida

Masukan Input: 7

```
*  
***  
*****  
*****  
*****  
*****  
*****
```

Rekomendasikan kode untuk menghasilkan luaran tersebut!

1.4. Analisa diagram flowchart dari Latihan 1.2 dan 1.3!

### [No.1] Analisis dan Argumentasi

- 1) Saya mengusulkan permasalahan ini dapat diatasi dengan cara:
  - 1.1 Menambahkan pada kode baris 1 yang kosong dengan continue dan juga pada kode baris kedua kosong menambahkan break. Merubah letak yang sebelumnya `y <= 15; y = 0; y++` menjadi `y = 0; y <= 15; y++` dan juga merubah dari double ke int.
  - 1.2 Menambahkan kode baris 1 yang kosong dengan break
  - 1.3 Mengubah arah looping dari yang sebelumnya dari tinggi ke 1 menjadi dari 1 sampai tinggi. Menambahkan loop untuk mencetak spasi di bagian depan Bintang. Menambahkan rumus  $(2 * t - 1)$
  - 1.4 Algoritma menggunakan dua loop for bersarang dengan kondisi  $i < 5$  dan  $j < 5$ . Setiap kali loop dalam (inner loop) dijalankan, variabel j akan diinisialisasi menjadi 1, dan akan melakukan perintah tertentu selama kondisi j kurang dari 5. Loop luar (outer loop) terus berjalan hingga nilai i mencapai 5, yang menyebabkan program selesai.
- 2) Alasan solusi ini karena:
  - 1.2 Continue; akan melanjutkan ke iterasi berikutnya. Break; menghentikan loop dan keluar dari loop. `y = 0; y <= 15; y++` memulai `y = 0` dan terus menambahkan y satu persatu hingga y mencapai 15. Pengubahan double ke int agar nilai tidak berbentuk decimal.
  - 1.2 Menghentikan eksekusi loop pertama, jadi tidak akan melanjutkan ke nilai i yang lebih besar
  - 1.3 Mengubah arah loop dari turun ke naik membuat logika lebih intuitif, membangun piramida dari atas ke bawah. Tambahkan spasi sebelum bintang memastikan bentuk piramida sesuai dari yang sesuai. Menggunakan rumus  $(2 * t - 1)$  untuk cetak jumlah bintang (1,3,5,...)
  - 1.4 Ini merupakan representasi umum dari algoritma bersarang (nested loop) yang berulang untuk nilai i dan j.
- 3) Perbaiki kode program dengan cara:
  - a. 1.1

```
public class ContohFor{
    public static void main(String[] args) {
        for (int y = 0; y <= 15; y++) {
            if (y % 2 == 1) {    //kondisi 1
                continue;
            } else if (y == 8) {    //kondisi 2
                break;
            } else
                System.out.println(y + " ");
        } } }
```
  - b. 1.2

```
public class ForBersarang {
    public static void main(String[] args) {
        pertama:
        for( int i = 1; i < 5; i++) {
            kedua:
            for(int j = 1; j < 3; j ++ ) {
                System.out.println("i = " + i + "; j = " + j);
            }
            if ( i == 2) {
                break;
            } } } }
```
  - c. 1.3

```
import java.util.Scanner;

public class ForBersarang {
```

```

public static void main(String[] args){
    //Instance Input Scanner
    Scanner input = new Scanner(System.in);
    System.out.print("Masukan Input: ");
    int tinggi = input.nextInt(); //Mendapatkan Input Dari User
    for(int t= 1; t <= tinggi; t++){
        //Menghitung Jumlah Tinggi Piramida
        for(int s=tinggi - t; s > 0; s--){
            //Menghitung Jumlah Spasi per Baris
            System.out.print(" ");
        }
        for(int b = 1; b <= (2 * t - 1); b++){
            //Menghitung Jumlah Spasi per Baris
            System.out.print("*");
        }
        System.out.println(); //Membuat Baris Baru
    }
}

```

#### [No.1 ] Penyusunan Algoritma dan Kode Program

##### a) Algoritma

###### a. 1.1

Algoritma adalah langkah-langkah penyelesaian masalah.

- a) Mulai
- b) Buka Programiz
- c) Deklarasi variabel
- d) Looping
- e) Kondisi pertama
- f) Kondisi kedua
- g) Run
- h) Selesai

###### b. 1.2

Algoritma adalah langkah-langkah penyelesaian masalah.

- b) Mulai
- c) Buka Programiz
- d) Looping pertama (i)
- e) Looping kedua (j)
- f) Run
- g) Selesai

###### c.1.3

- d. Mulai
- e. Buka Programiz
- f. Instansiasi scanner
- g. Minta input
- h. Loop untuk tinggi piramida
- i. Loop untuk spasi
- j. Loop untuk bintang
- k. Run

I. Selesai

h) Kode program dan luaran

a. 1.1

Main.java	Output
<pre>1- public class ContohFor{ 2-     public static void main(String[] args) { 3-         for (int y = 0; y &lt;= 15; y++) { 4-             if (y % 2 == 1) {           //kondisi 1 5-                 continue; 6-             } else if (y == 8) {        //kondisi 2 7-                 break; 8-             } else 9-                 System.out.println(y + " "); 10-        } } }</pre>	<pre>java -cp /tmp/38pMXwanbs/ContohFor 0 2 4 6 === Code Execution Successful ===</pre>

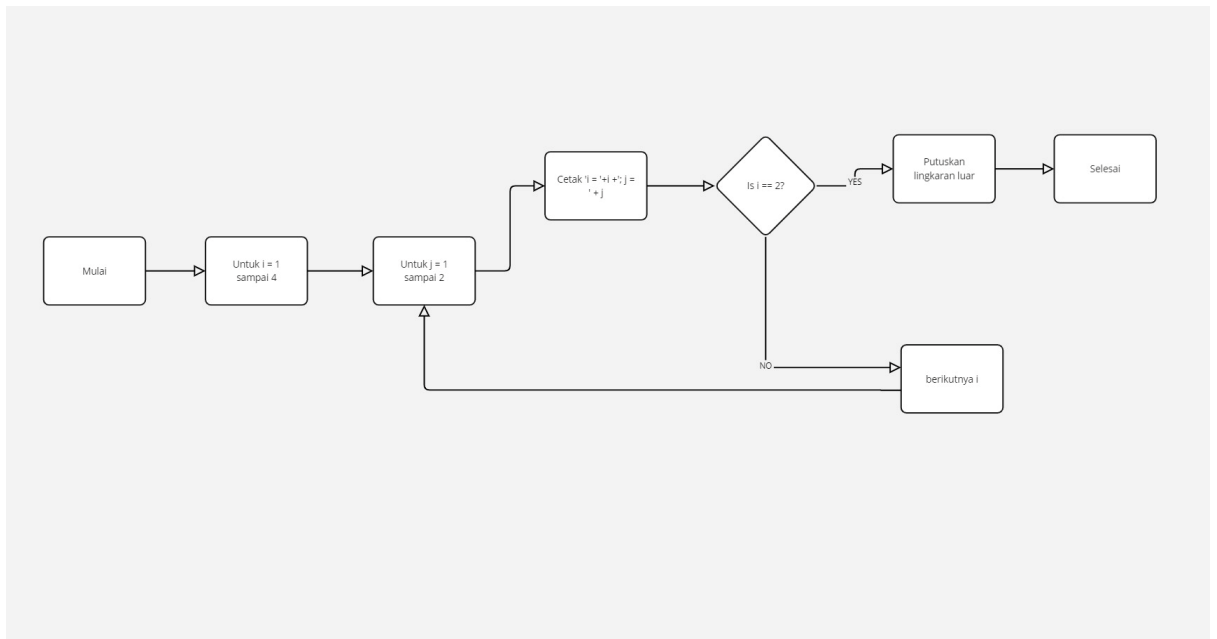
b. 1.2

Main.java	Output
<pre>1- public class ForBersarang { 2-     public static void main(String[] args) { 3-         pertama: 4-         for( int i = 1; i &lt; 5; i++) { 5-             kedua: 6-             for(int j = 1; j &lt; 3; j ++ ) { 7-                 System.out.println("i = " + i + "; j = " + j); 8-             } 9-             if ( i == 2) { 10-                 break; 11-             } } } }</pre>	<pre>java -cp /tmp/41kAjXr50r/ForBersarang i = 1; j = 1 i = 1; j = 2 i = 2; j = 1 i = 2; j = 2 === Code Execution Successful ===</pre>

c. 1.3

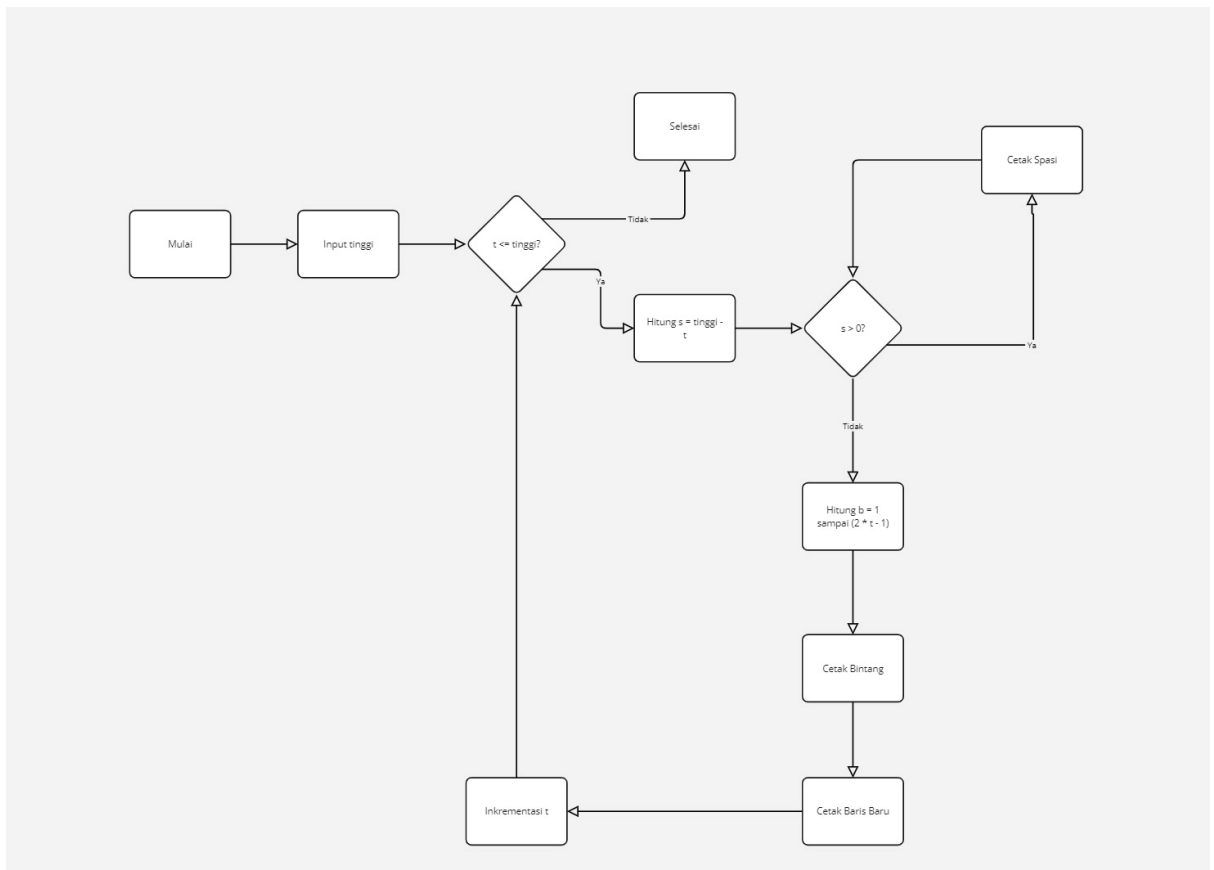
Main.java	Output
<pre>1- import java.util.Scanner; 2- 3- public class ForBersarang { 4-     public static void main(String[] args){ 5-         //Instance Input Scanner 6-         Scanner input = new Scanner(System.in); 7-         System.out.print("Masukan Input: "); 8-         int tinggi = input.nextInt(); //Mendapatkan Input Dari User 9-         for(int t= 1; t &lt;= tinggi; t++){ 10-            //Menghitung Jumlah Tinggi Piramida 11-            for(int s=tinggi - t; s &gt; 0; s--){ 12-                //Menghitung Jumlah Spasi per Baris 13-                System.out.print(" "); 14-            } 15-            for(int b = 1; b &lt;= (2 * t - 1); b++){ 16-                //Menghitung Jumlah Spasi per Baris 17-                System.out.print("*"); 18-            } 19-            System.out.println(); //Membuat Baris Baru 20-        } 21-    } 22- }</pre>	<pre>java -cp /tmp/040ZeETOnF/ForBersarang Masukan Input: 7 * *** ***** ***** ***** ***** ***** === Code Execution Successful ===</pre>

d. Flowchart 1.2



i)

j) Flowchart 1.3



a) Analisa luaran yang dihasilkan

1) 1.1 Luaran sudah sesuai dengan program yang dengan menghasilkan (0,2,4,6)

2) 1.2 Luaran sudah sesuai dengan program yang dengan menghasilkan

(i = 1; j = 1

i = 1; j = 2

i = 2; j = 1

i = 2; j = 2)

3) 1.3 Luaran sudah sesuai dengan program yang dengan menghasilkan

Masukan Input: 7

```

*
***
*****
*****
*****
*****
*****
*****
*****
*****

```



## [No.1] Kesimpulan

(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)

### 1) Analisa (1.1)

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program:  
Program ini menggunakan loop for untuk melakukan iterasi dari nilai 0 hingga 15. Di loop terdapat dua kondisi if dan else if. Kondisi 1 nilai y adalah adalah bilangan ganjil perintah continue akan dilewatkan sehingga lanjut ke nilai y berikutnya tanpa menjalankan baris kode di bawahnya. Kondisi 2 jika nilai y adalah 8 maka pernyataan break (berhenti eksekusi).
- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini;  
Penggunaan kelas public untuk menandai ContohFor bisa diakses semua bagian program. Continue untuk melewati iterasi nilai y adalah bilangan ganjil hanya pemroses bilangan genap. Penggunaan break untuk menghentikan loop nilai y mencapai 8

### 2) Evaluasi (1.2)

- a) Apa konsekuensi/dampak dari kode program yang dibuat:
  - Nilai i hanya akan mencapai 2, dan setelah itu, seluruh loop dihentikan.
  - Iterasi selanjutnya (i = 3, i = 4) tidak akan dieksekusi.
- b) Evaluasi input program, proses perhitungan, dan luaran yang dihasilkan:

Input program

- Loop luar berjalan untuk nilai i dari 1 hingga kurang dari 5.
- Loop dalam berjalan untuk nilai j dari 1 hingga kurang dari 3.

Proses perhitungan

- Pada setiap iterasi, program mencetak nilai i dan j.
- Ketika i sama dengan 2, program menemui pernyataan if (i == 2) dan perintah break menghentikan eksekusi loop luar, sehingga tidak ada lagi iterasi untuk i = 3 dan seterusnya.

Luaran dihasilkan

i = 1; j = 1

i = 1; j = 2

i = 2; j = 1

i = 2; j = 2

### 3) Kreasi (1.3)

- a) Apakah ada pengetahuan baru yang dikembangkan dan konsep baru sebagai usulan solusi:

Pada program ini, tidak ada konsep baru yang dikembangkan , namun konsep loop bersarang diterapkan untuk membentuk pola piramida menggunakan simbol \*.

b) Susunlah hubungan antara variabel yang berbeda dengan konsep yang anda ketahui! (jika ada):

- Variabel tinggi mempengaruhi jumlah iterasi dari loop luar, yang menentukan jumlah baris dari piramida.
- Variabel t digunakan dalam loop luar untuk menentukan baris saat ini, serta mempengaruhi nilai s (jumlah spasi) dan b (jumlah simbol \*).
- Loop dalam (s dan b) berfungsi untuk mengatur jumlah spasi dan jumlah bintang yang akan ditampilkan pada setiap baris.

## [No. 2] Identifikasi Masalah:

1) Uraikan permasalahan dan variabel

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

```
public class ContohWhile{
public static void main(String[] args) {
    int i=1;
    while(i<=6){
        System.out.println(i);
        i++;
        if(i==4){
            break;          //ubah1
        }
    }
}
```

**Luaran:**

```
1
2
3
```

**Contoh 5:** Salin dan tempel kode program berikut ke Eclipse.

```
import java.util.Scanner;

public class ForBersarang {
    public static void main(String[] args) {
        Scanner dataKata = new Scanner(System.in);
        System.out.print("Masukkan Kata yang ingin diulang : ");
        String kata = dataKata.nextLine();

        Scanner dataJumlah = new Scanner(System.in);
        System.out.print("Masukkan Jumlah ingin diulang : ");
        int jumlah = dataJumlah.nextInt();

        int i = 0; //Inisialisasi batas dasar
        while(i < jumlah){
            System.out.println(kata);
            i++; //Faktor pengulang Increment
        }
    }
}
```

**Luaran Contoh 5:**

```
Masukkan Kata yang ingin diulang : Fakultas Teknik
Masukkan Jumlah ingin diulang : 5
Fakultas Teknik
Fakultas Teknik
Fakultas Teknik
Fakultas Teknik
Fakultas Teknik
```

## Latihan 2

2.1. Ubahlah baris kode pada Contoh 4

//Ubah1 menjadi `if(i % 3 == 0){ ◇ running, periksa hasilnya`

//Ubah2 menjadi `continue; ◇ running, periksa hasilnya`

Evaluasi perbandingan luaran sebelum dan setelah diubah! Simpulkan maksud dari perubahan tersebut!

2.2. Cermati Contoh 5. Periksa luaran, bila ketika di eksekusi, jumlah yang diulang = 0!

Evaluasi luaran, bila kode diubah menjadi `do ... while` dengan masukan sama jumlah yang diulang = 0.

Simpulkan perbedaan `while` dan `do ... while`!

2.3. Bila diketahui pernyataan pseudocode berikut:

[1] inisiasi `idPelajaran`

[2] inisiasi nilai pelajaran

[3] inisiasi nilai rata-rata

[4] Minta pengguna untuk menuliskan jumlah pelajaran

[5] Ketika `idPelajaran` lebih kecil dari jumlah pelajaran

[6] Minta pengguna untuk menuliskan nilai pelajaran

[7] Hitung nilai rata-rata =  $(\text{nilai pelajaran} + \text{nilai rata-rata}) / 2$

[8] Tambah satu ke `idPelajaran`

[9] Tampilkan nilai rata-rata

Rekomendasikan kode untuk menyelesaikan Pseudocode tersebut!

2.4. Rancang diagram flowchart dari Latihan 2.1, Latihan 2.2, dan Latihan 2.3!

## [No.2] Analisis dan Argumentasi

4) Saya mengusulkan permasalahan ini dapat diatasi dengan cara:

2.1 Perbedaannya adalah `break` ini menghentikan semua loop, sementara `continue` ini melompati iterasi tertentu tidak menghentikan loop keseluruhannya.

2.2 Meskipun jumlah = 0, blok `do` tetap akan dijalankan sekali sebelum kondisi diperiksa, lalu program luaran kata tersebut satu kali.

`while`: Kondisi akan diperiksa sebelum perulangan dimulai Jika kondisi tidak dipenuhi dari awal maka perulangan tidak akan dijalankan sama sekali.

`Do while`: Perulangan dijalankan satu kali dari sebelumnya kondisi diperiksa dulu, lalu perintah dalam blok `do` akan selalu dieksekusi sekali.

5) Alasan solusi ini karena:

2.1 Perubahan di `if(i % 3 == 0)` dan `continue` digunakan untuk alur program agar dapat dilakukan oleh kondisi tertentu dan cara perintah `continue` lompat iterasi tertentu tidak menghentikan seluruh perulangan. Ini memberi bantu untuk mengontrol jalannya perulangan secara baik.

2.2 `do while`: Menggantinya `while` dengan `do while` untuk digunakan untuk menunjukkan perbedaan dari kedua struktur perulangan. Dengan `do while` kode akan dijalankan setidaknya sekali sebelum kondisi dievaluasi, sedangkan `while` memeriksa kondisi dulu lalu menjalankan blok perintah.

6) Perbaiki kode program dengan cara :

a.2.1

```
public class ContohWhile{
    public static void main(String[] args) {
        int i = 1;
        while(i <= 6){
            System.out.println(i);
```

```

        i++;
        if(i % 3 == 0){ // Ubah1
            continue; // Ubah2
        }
    }
}
}

```

c. 2.3

```
import java.util.Scanner;
```

```

public class HitungRataRata {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int idPelajaran = 0;
        double nilaiPelajaran, nilaiRataRata = 0;

        System.out.print("Masukkan jumlah pelajaran: ");
        int jumlahPelajaran = input.nextInt();

        while(idPelajaran < jumlahPelajaran) {
            System.out.print("Masukkan nilai pelajaran ke-" + (idPelajaran + 1) + ": ");
            nilaiPelajaran = input.nextDouble();
            nilaiRataRata = (nilaiPelajaran + nilaiRataRata) / 2;
            idPelajaran++;
        }

        System.out.println("Nilai rata-rata: " + nilaiRataRata);
    }
}

```

## [No.2 ] Penyusunan Algoritma dan Kode Program

- k) Algoritma  
Algoritma adalah langkah-langkah penyelesaian masalah.
  - (a) Mulai
  - (b) Inisialisasi variabel  $i = 1$
  - (c) Lakukan perulangan **while** selama  $i \leq 6$
  - (d) Kembali ke langkah 3 sampai kondisi  $i \leq 6$  tidak terpenuhi
  - (e) Selesai
- l) Algoritma  
Algoritma adalah langkah-langkah penyelesaian masalah.
  - (f) Mulai
  - (g) Inisialisasi variabel  $idPelajaran = 0$  dan  $nilaiRataRata = 0$
  - (h) Lakukan perulangan while selama  $idPelajaran < jumlahPelajaran$
  - (i) Setelah perulangan selesai, cetak nilai rata-rata akhir
  - (j) Selesai

- m) Kode program dan luaran 2.1

Main.java	Output
<pre> 1- public class ContohWhile{ 2-     public static void main(String[]            args) { 3-         int i = 1; 4-         while(i &lt;= 6){ 5-             System.out.println(i); 6-             i++; 7-             if(i % 3 == 0){ // Ubah1 8-                 continue; // Ubah2 9-             } 10-        } 11-    } 12- } 13- </pre>	<pre> java -cp /tmp/02Bx6ecdeP/ContohWhile 1 2 3 4 5 6  === Code Execution Successful === </pre>

i) Kode program dan luaran 2.3

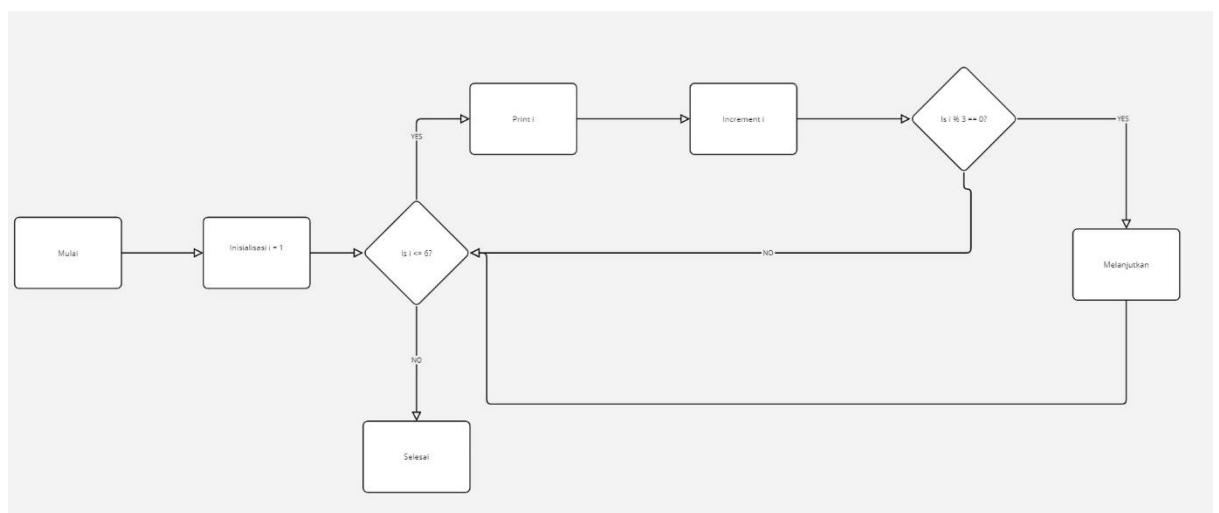
Main.java	Output
<pre> 1- import java.util.Scanner; 2- 3- public class HitungRataRata { 4-     public static void main(String[] args) { 5-         Scanner input = new Scanner(System.in); 6- 7-         int idPelajaran = 0; 8-         double nilaiPelajaran, nilaiRataRata = 0; 9- 10-        System.out.print("Masukkan jumlah pelajaran: "); 11-        int jumlahPelajaran = input.nextInt(); 12- 13-        while(idPelajaran &lt; jumlahPelajaran) { 14-            System.out.print("Masukkan nilai pelajaran ke-" + (idPelajaran + 1) + 15-                             " "); 16-            nilaiPelajaran = input.nextDouble(); 17-            nilaiRataRata = (nilaiPelajaran + nilaiRataRata) / 2; 18-            idPelajaran++; 19-        } 20- 21-        System.out.println("Nilai rata-rata: " + nilaiRataRata); 22-    } 23- } 24- </pre>	<pre> java -cp /tmp/qgcnC8mUVI/HitungRataRata Masukkan jumlah pelajaran:   </pre>

b) Analisa luaran yang dihasilkan

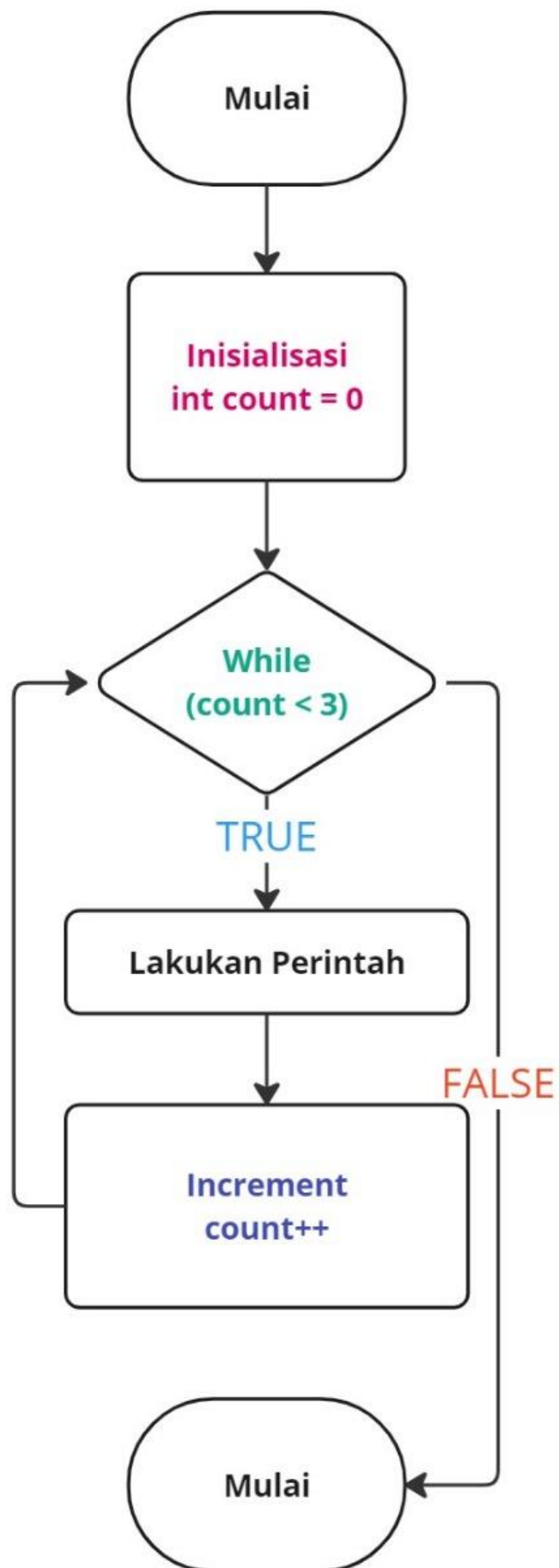
Luaran sudah sesuai dengan program yang disusun.

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

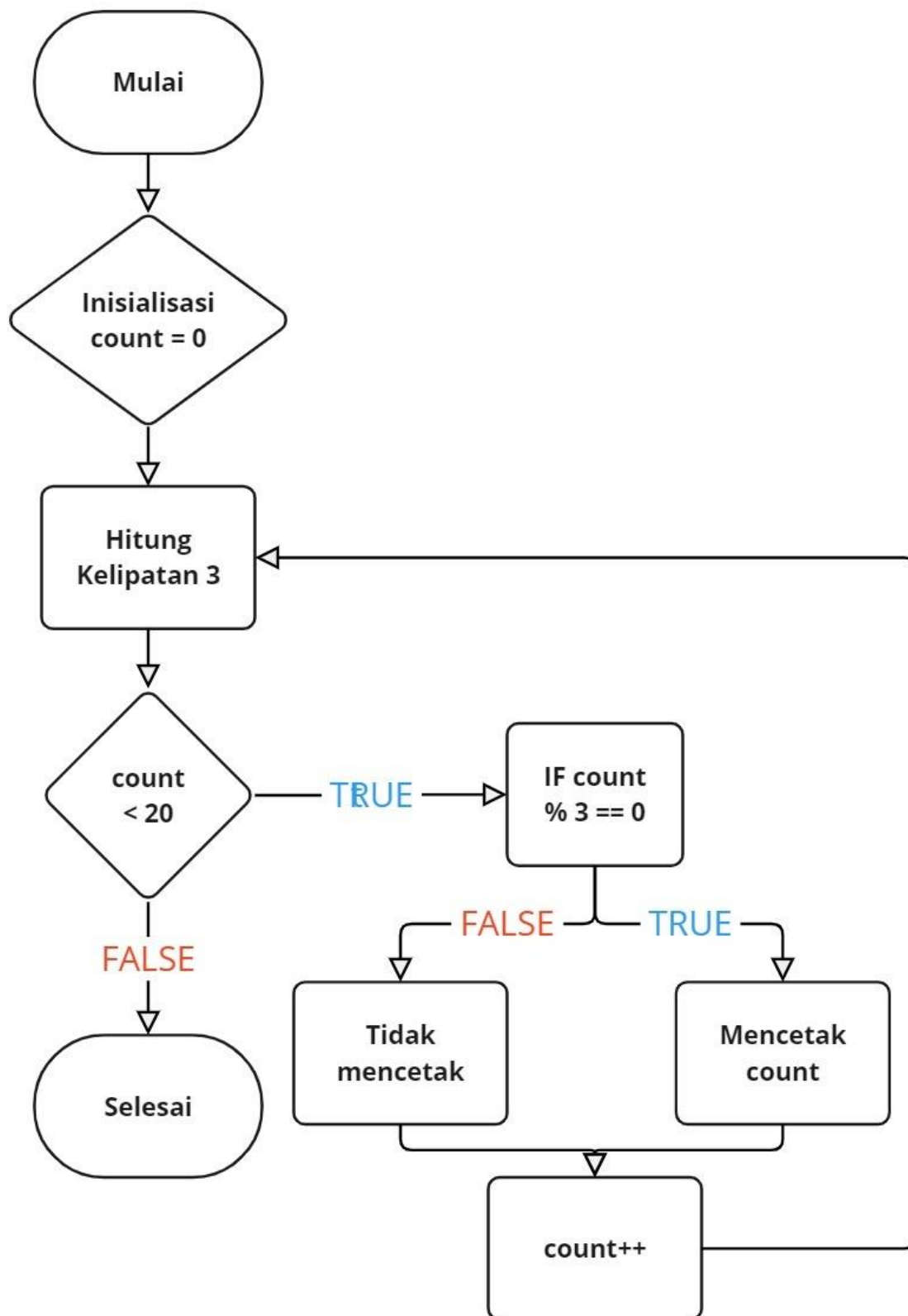
j) Flowchart 2.1



k) Flowchart 2.2



c.2.3



[No.2] Kesimpulan

(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)

c) Analisa



## 2.1

Pada program ini ingin untuk menghasilkan luaran nilai variabel *i* mulai dari angka 1 sampai 6. Ada juga sebuah kondisi yang mana program akan melewati suatu tahapan pada nilai *i* adalah kelipatan dari 3.

Program memulai dengan menginisialisasikan untuk variabel *i* ini bernilai 1. Pada while akan berjalan selama *i* ini kurang dari atau sama dengan 6. Setiap iterasi, nilai *i* akan dicetak, lalu ditambahkan satu. Bila nilai *i* ini adalah kelipatan 3 tahapan continue akan dijalankan yang membuat program melewati sisa kode dalam iterasi

## 2.3

Pada program ini akan digunakan untuk hitung rata-rata dari nilai beberapa pelajaran yang dimasukkan oleh saya.

Saya memasukkan jumlah pelajaran.

Program ini akan mengulangi proses meminta nilai pelajaran lalu menghitung rata-rata ini dengan rumus berikut  $(\text{nilaiPelajaran} + \text{nilaiRataRata}) / 2$  setelah setiap masukan.

Program ini akan terus mengulangi tahap ini sampai semua nilai pelajaran dimasukkan lalu mencetak luaran hasil rata-rata.

## Refleksi

(Tuliskan singkat tentang pengalaman belajar, pemaknaan pengetahuan yang baru, tantangan yang dihadapi pada minggu tersebut. Ringkasan singkat dari semua soal, bukan per soal)