

Template Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
Vivilia Regita Pramesti G1F024060		
[Nomor Soal] Identifikasi Masalah:		
<ol style="list-style-type: none">1) Uraikan permasalahan dan variabel2) Rincikan sumber informasi yang relevan (buku / webpage)3) Uraikan rancangan solusi yang diusulkan (jika ada).4) Analisis susunan solusi, parameter solusi (jika ada).		
[Nomor Soal] Analisis dan Argumentasi		
<ol style="list-style-type: none">1) Uraikan rancangan solusi yang diusulkan.2) Analisis solusi, kaitkan dengan permasalahan.		
[Nomor Soal] Penyusunan Algoritma dan Kode Program		
<ol style="list-style-type: none">1) Rancang desain solusi atau algoritma2) Tuliskan kode program dan luaran<ol style="list-style-type: none">a) Beri komentar pada kodeb) Uraikan luaran yang dihasilkanc) Screenshot/ Capture potongan kode dan hasil luaran		
[Nomor Soal] Kesimpulan		
<ol style="list-style-type: none">1) Analisa<ol style="list-style-type: none">a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?2) Evaluasi<ol style="list-style-type: none">a) Apa konsekuensi dari skenario pemrograman ini?b) Evaluasi input, proses, dan luaran yang dihasilkan! (jika ada)3) Kreasi<ol style="list-style-type: none">a) Apakah ada pengetahuan baru yang dikembangkan dan konsep baru sebagai usulan solusi?b) Konstruksikan hubungan antara variabel yang berbeda dengan konsep yang anda ketahui! (jika ada)		

Contoh Jawaban:

Nama & NPM	Topik:	Tanggal:
Vivilia Regita Pramesti G1F024060	FOR WHILE	06 Oktober 2024

[No. 1] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variabel

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!

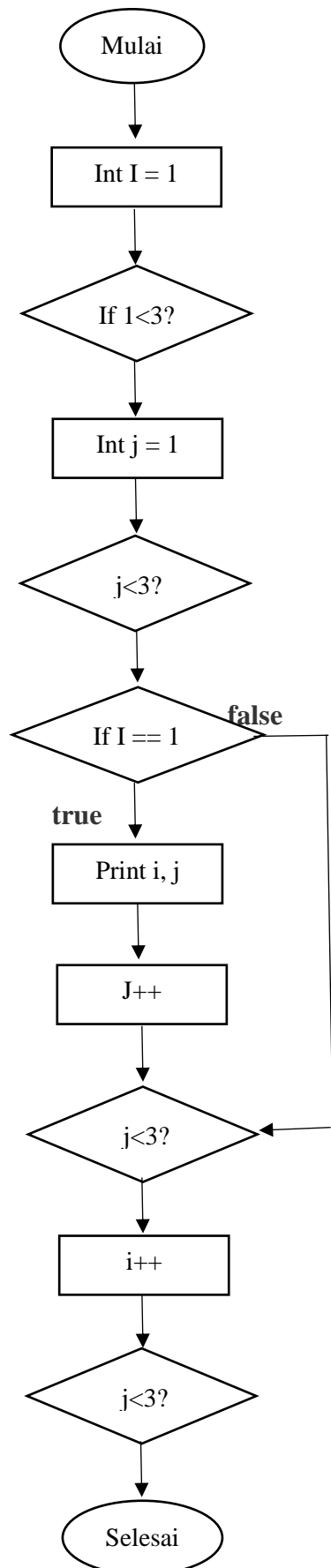
Jawab :

1.1 ada kesalahan for (double y <= 15; y = 0; y++) memiliki beberapa kesalahan yaitu tipe datanya double tipe data tersebut untuk bilangan pecahan sedangkan output yang diinginkan adalah 0 2 4 6 bilangan bulat, sebaiknya diganti dengan tipe data int. selanjutnya y <= 15; diganti dengan y = 0; untuk inisialisasi dan output dimulai dari 0, selanjutnya y = 0; seharusnya y <= 15; tapi karena outputnya ingin nilai 0 2 4 6 jadi saya ganti y < 10;. baris kode yang kosong 1 dan 2 saya biarkan kosong agar output bisa sesuai yang diinginkan.

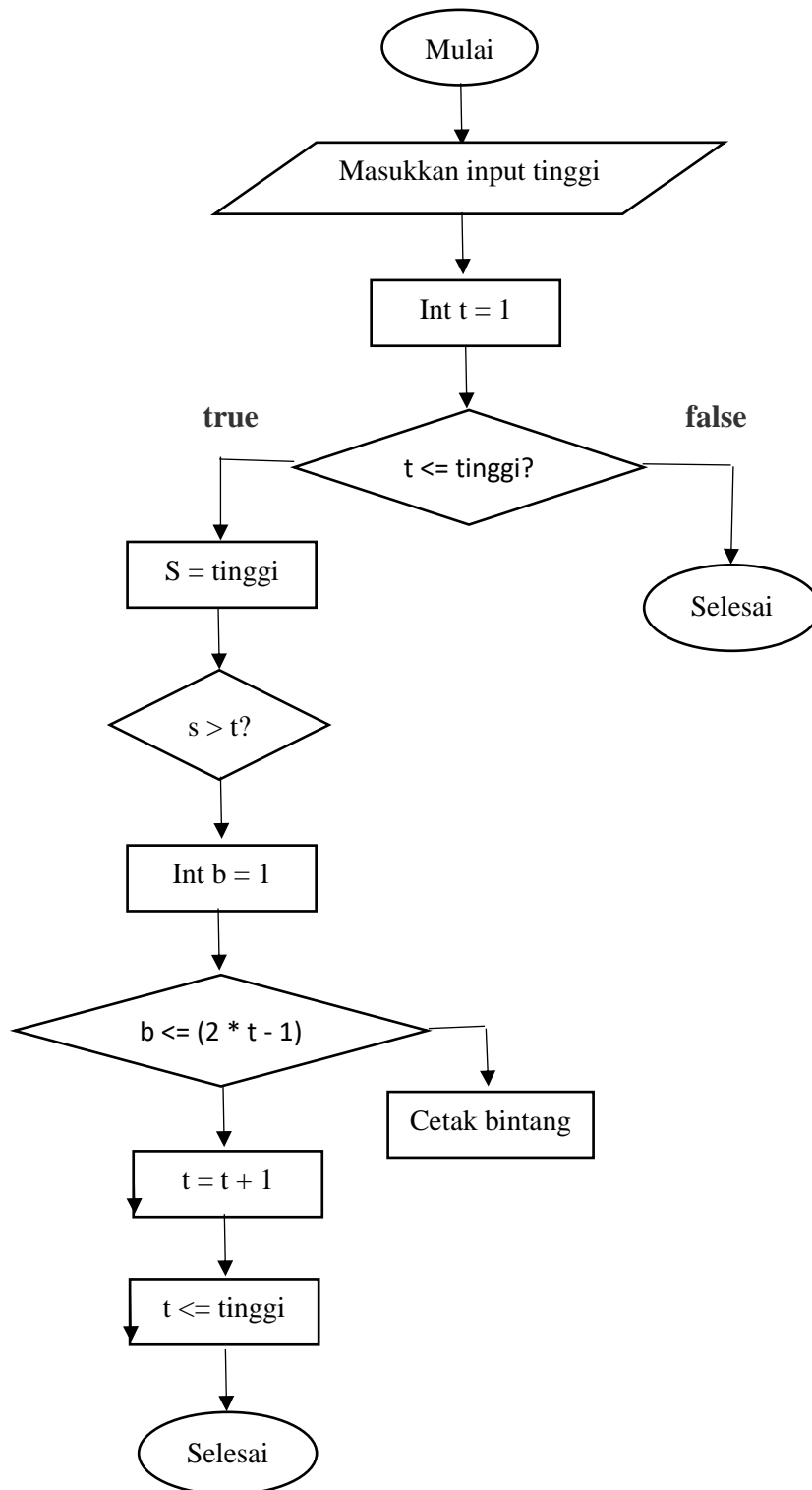
1.2 Saya menggunakan continue dan saya letakkan pada loop pertama. Ketika i == 1, continue digunakan untuk melewati sisa kode di bawahnya dalam loop pertama dan langsung melanjutkan ke iterasi berikutnya dari loop i. Ini berarti bahwa blok kode di bawah if tidak akan dieksekusi saat i == 1.

1.3 Untuk membentuk piramida saya mengubah beberapa operator dan menambahkan loop spasi yang akan menghitung jumlah spasi yang perlu dicetak di setiap baris dan menambahkan loop Bintang yaitu jumlah bintang yang harus dicetak berdasarkan rumus $2 * t - 1$

1.4 Flowchart 1.2 Analisis: Flowchart ini jelas menggambarkan struktur bersarang dan logika kondisi. Penggunaan continue atau break dapat memberikan efek yang berbeda pada alur program, yang dapat membantu dalam kontrol alur eksekusi.



flowchart 1.3 Analisis: Flowchart ini dengan jelas menunjukkan proses logika untuk mencetak piramida. Penggunaan loop bersarang di sini efektif dalam menciptakan pola yang diinginkan, dan setiap langkah terstruktur dengan baik. Diagram ini membantu dalam memahami bagaimana spasi dan bintang diatur untuk membentuk piramida.



[No.1] Analisis dan Argumentasi

Contoh 1:

Saya mengusulkan permasalahan ini dapat diatasi dengan merubah program yang salah yaitu `for (double y <= 15; y = 0; y++) {` merubahnya dengan `for (int y = 0; y < 10; y++) {` agar output sesuai. Perbaikan kode program bisa dilakukan dengan menyesuaikan kondisi dalam loop, sehingga hanya angka genap yang dicetak. Kita dapat meninggalkan blok kode kosong untuk kondisi bilangan ganjil dan angka 8, uga memastikan bahwa hanya nilai y yang memenuhi kondisi genap yang dicetak ke konsol. Dengan cara ini, program akan berjalan dengan efisien dan menghasilkan keluaran yang sesuai dengan harapan, yaitu mencetak angka 0, 2, 4, dan 6.

Contoh 2:

Saya mengusulkan permasalahan ini dengan menambahkan `continue` diloop pertama. Alasan solusi ini adalah struktur kontrol tersebut memungkinkan kita untuk menghentikan atau melanjutkan iterasi sesuai dengan kondisi yang diinginkan. Perbaikan kode program dengan cara menambahkan `continue` ketika `i == 1` untuk langsung beralih ke iterasi i berikutnya.

Contoh 3:

Saya merekomendasikan penggunaan loop bersarang untuk mencetak pola piramida bintang karena metode ini memungkinkan untuk mengontrol jumlah spasi dan bintang yang ada pada setiap baris, yang menghasilkan bentuk piramida yang simetris. Loop bersarang memberikan fleksibilitas untuk membuat pola yang diinginkan, yang membuat solusi ini berhasil. Perbaikan kode dilakukan dengan menambahkan loop untuk mencetak spasi dan menggunakan rumus $2 * t - 1$ untuk menghitung jumlah bintang. Langkah ini memungkinkan program membuat piramida dengan tinggi yang diinginkan pengguna.

[No.1] Penyusunan Algoritma dan Kode Program

1) Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

Algoritma contoh 1:

- a. Mulai
- b. Inisialisasi `y = 0`
- c. Selama `y < 10`
- d. Jika `y % 2 == 1`: Abaikan
- e. Jika `y == 8`: Abaikan
- f. Jika tidak: Cetak y
- g. Tambahkan `y++`
- h. Selesai.

Algoritma contoh 2:

- a. Mulai
- b. Buat loop pertama i
- c. Jika i sama dengan 1 buat loop j, cetak nilai i dan j
- d. ika i sama dengan 2 buat loop j, cetak nilai i dan j
- e. selesai.

Algoritma contoh 3:

- (a) Mulai
- (b) Import kelas scanner
- (c) Tampilkan pesan untuk meminta pengguna memasukkan tinggi piramida
- (d) Baca input tinggi dari pengguna dan simpan dalam variabel tinggi.
- (e) Buat loop spasi
- (f) Buat loop Bintang
- (g) Selesai

2) Kode program dan luaran

- a) Screenshot/ Capture potongan kode dan hasil luaran
Beri komentar pada kode yang di Screenshot

Contoh 1:

```
1 package For;
2
3 public class LatihanFor {
4     public static void main(String[] args) {
5         for (int y = 0; y < 10; y++) {
6             if (y % 2 == 1) { //kondisi 1
7                 // baris kode kosong 1, dibiarkan kosong agar output sesuai
8             } else if (y == 8) { //kondisi 2
9                 // baris kode kosong 2, dibiarkan kosong agar output sesuai
10            } else
11                System.out.println(y);
12        } } }
```

Problems Javadoc Declaration Console x

<terminated> LatihanFor [Java Application] C:\Users\ASUS\AppData\Local\Temp\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.12.v20240802-1518\jre\bin\javaw.exe (4 Okt 2024 00.11.48 - 00.11.49) [pid: 1270]

0
2
4
6

Contoh 2:

```
1 package Bersarang;
2
3 public class ForBersarang {
4
5     public static void main(String[] args) {
6         pertama:
7         for (int i = 1; i < 3; i++) {
8             kedua:
9             for (int j = 1; j < 3; j++) {
10                 if (i == 1) {
11                     System.out.println("i = " + i + "; j = " + j);
12                 }
13             }
14             if (i == 1) {
15                 continue; // Melanjutkan ke iterasi berikutnya
16             }
17             // Loop j kedua untuk i = 2
18             for (int j = 1; j < 3; j++) {
19                 System.out.println("i = " + i + "; j = " + j);
20             }
21         }
22     }
23 }
24 }
```

Problems Javadoc Declaration Console x

<terminated> ForBersarang [Java Application] C:\Users\ASUS\AppData\Local\Temp\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.12.v20240802-1518\jre\bin\javaw.exe (4 Okt 2024 00.12.00 - 00.12.03) [pid: 2024]

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

Contoh 3:

```

1 package Scanner;
2
3 import java.util.Scanner;
4
5 public class DuaBersarang {
6
7     public static void main(String[] args) {
8         Scanner input = new Scanner(System.in);
9         System.out.print("Masukan Input: ");
10        int tinggi = input.nextInt(); // Mendapatkan Input Dari User
11
12        for (int t = 1; t <= tinggi; t++) {
13            // Menghitung Jumlah Spasi per Baris
14            for (int s = tinggi; s > t; s--) {
15                System.out.print(" "); // Mencetak spasi
16            }
17            // Menghitung Jumlah Bintang per Baris
18            for (int b = 1; b <= (2 * t - 1); b++) {
19                System.out.print("*"); // Mencetak bintang
20            }
21            System.out.println(); // Membuat Baris Baru
22        }
23    }
24 }

```

Masukan Input: 7

```

|
|      *
|      ***
|      *****
|      *******
|      *********
|      ***********
|      *****
|      ***
|      *

```

b) Analisa luaran yang dihasilkan

Jawab:

Luaran sudah sesuai dengan program yang disusun.

[No.1] Kesimpulan

Analisa

a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program! Contoh 1 menunjukkan pentingnya penggunaan tipe data yang tepat pada loop agar mendapatkan output sesuai yang diinginkan. Dengan memperbaiki program yang salah program akan berjalan sesuai harapan. Contoh 2 menunjukkan bahwa penggunaan break dan continue dapat memberikan hasil yang berbeda dan bisa membantu mencetak kombinasi nilai yang diinginkan. Contoh 3: menunjukkan penggunaan loop bersarang untuk mencetak pola yang simetris, di mana pengaturan spasi dan bintang sangat penting untuk mencapai tampilan yang diinginkan.

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

Jawab: Keputusan dalam kasus ini didasarkan pada beberapa aspek penting. Pertama, identifikasi kesalahan sintaksis, seperti tipe data yang salah dan pengaturan loop yang tidak tepat, perlu diperbaiki agar program berfungsi dengan benar.

[No. 2] Identifikasi Masalah:

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 = (nilai pelajaran + nilai rata-rata) / 2
- [8] Tambah satu ke idPelajaran
- [9] Tampilkan nilai rata-rata

Rekomendasikan kode untuk menyelesaikan Pseudocode tersebut!

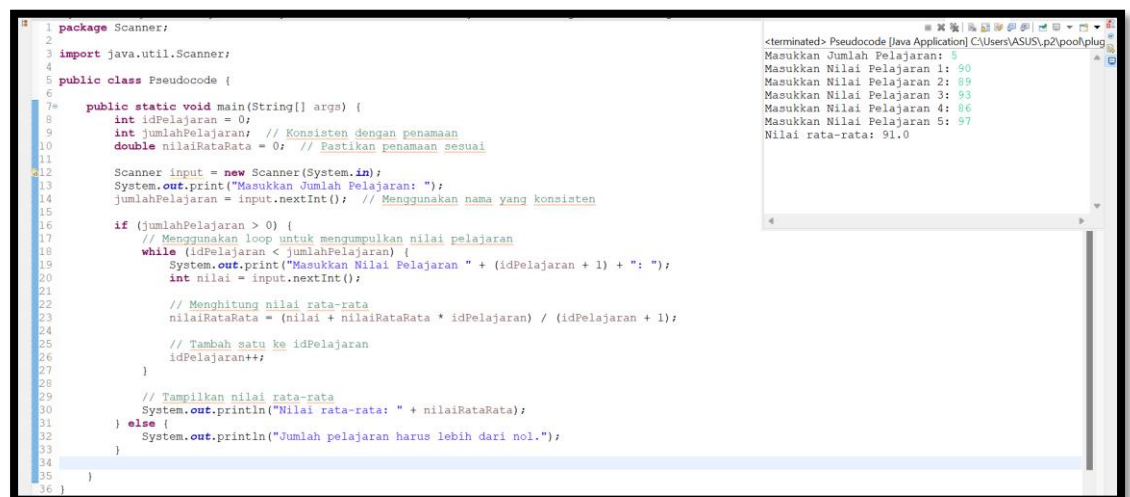
2.4. Rancang diagram flowchart dari Latihan 2.1, Latihan 2.2, dan Latihan 2.3!

Jawab:

2.1 Mengubah kondisi dari $\text{if}(i == 4)$ menjadi $\text{if}(i \% 3 == 0)$ mengubah program dari menghentikan loop pada 4 menjadi melewati pencetakan angka 3. Dan dengan menggunakan continue, program akan melanjutkan iterasi berikutnya tanpa mencetak angka yang saat itu merupakan kelipatan 3.

2.2 While itu kodenya di eksekusi ketika kondisi awal terpenuhi, jika kondisi tidak terpenuhi maka kode tidak dieksekusi. Sedangkan do while kode setidaknya dieksekusi sekali walaupun kode tidak terpenuhi

2.3 Saya telah membuat program untuk pseudocode

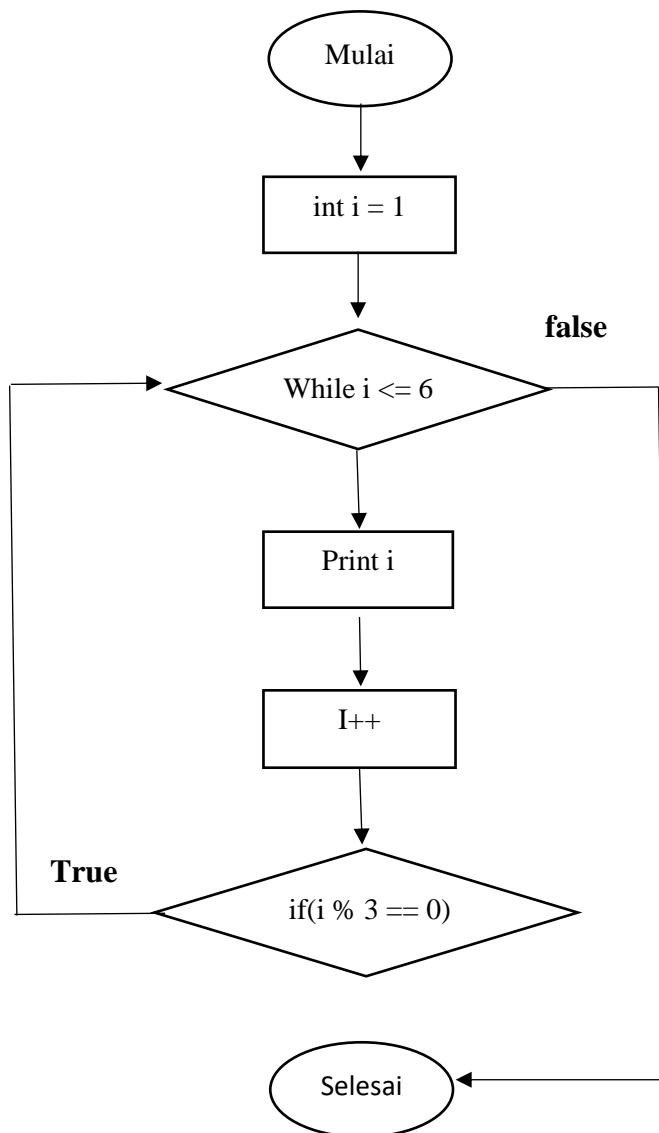


```
1 package Scanner;
2
3 import java.util.Scanner;
4
5 public class Pseudocode {
6
7     public static void main(String[] args) {
8         int idPelajaran = 0;
9         int jumlahPelajaran; // Konsisten dengan penamaan
10        double nilaiRataRata = 0; // Pastikan penamaan sesuai
11
12        Scanner input = new Scanner(System.in);
13        System.out.print("Masukkan Jumlah Pelajaran: ");
14        jumlahPelajaran = input.nextInt(); // Menggunakan nama yang konsisten
15
16        if (jumlahPelajaran > 0) {
17            // Menggunakan loop untuk mengumpulkan nilai pelajaran
18            while (idPelajaran < jumlahPelajaran) {
19                System.out.print("Masukkan Nilai Pelajaran " + (idPelajaran + 1) + ": ");
20                int nilai = input.nextInt();
21
22                // Menghitung nilai rata-rata
23                nilaiRataRata = (nilai + nilaiRataRata * idPelajaran) / (idPelajaran + 1);
24
25                // Tambah satu ke idPelajaran
26                idPelajaran++;
27            }
28
29            // Tampilkan nilai rata-rata
30            System.out.println("Nilai rata-rata: " + nilaiRataRata);
31        } else {
32            System.out.println("Jumlah pelajaran harus lebih dari nol.");
33        }
34    }
35 }
36 }
```

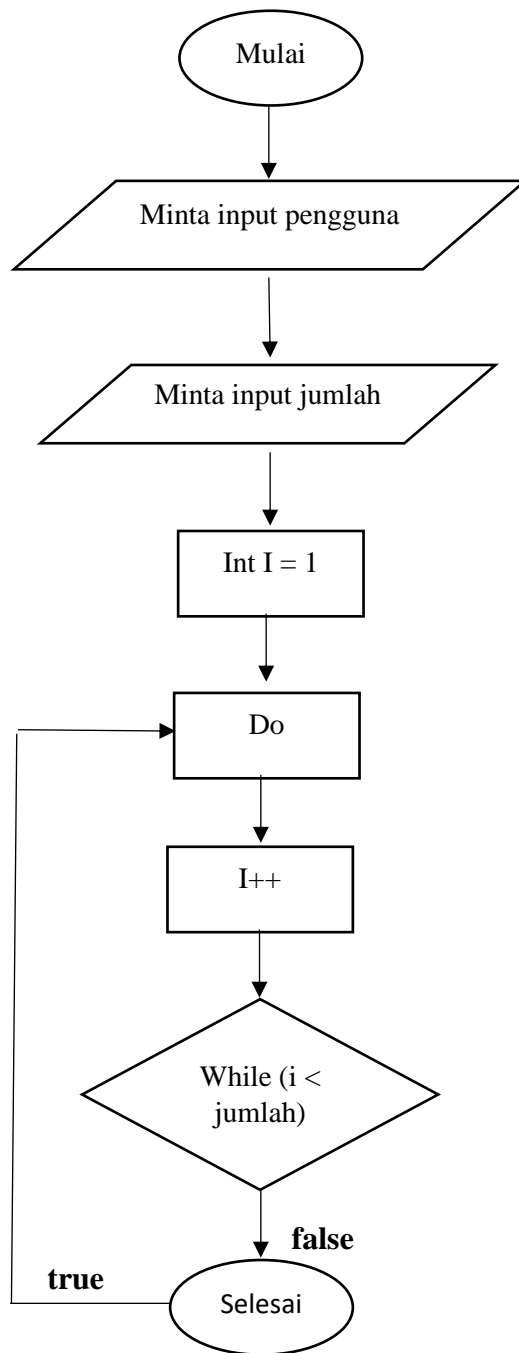
<terminated> Pseudocode [Java Application] C:\Users\ASUS\AppData\Local\Temp\pseudocode\pseudocode.jar

Masukkan Jumlah Pelajaran: 5
Masukkan Nilai Pelajaran 1: 90
Masukkan Nilai Pelajaran 2: 89
Masukkan Nilai Pelajaran 3: 93
Masukkan Nilai Pelajaran 4: 86
Masukkan Nilai Pelajaran 5: 97
Nilai rata-rata: 91.0

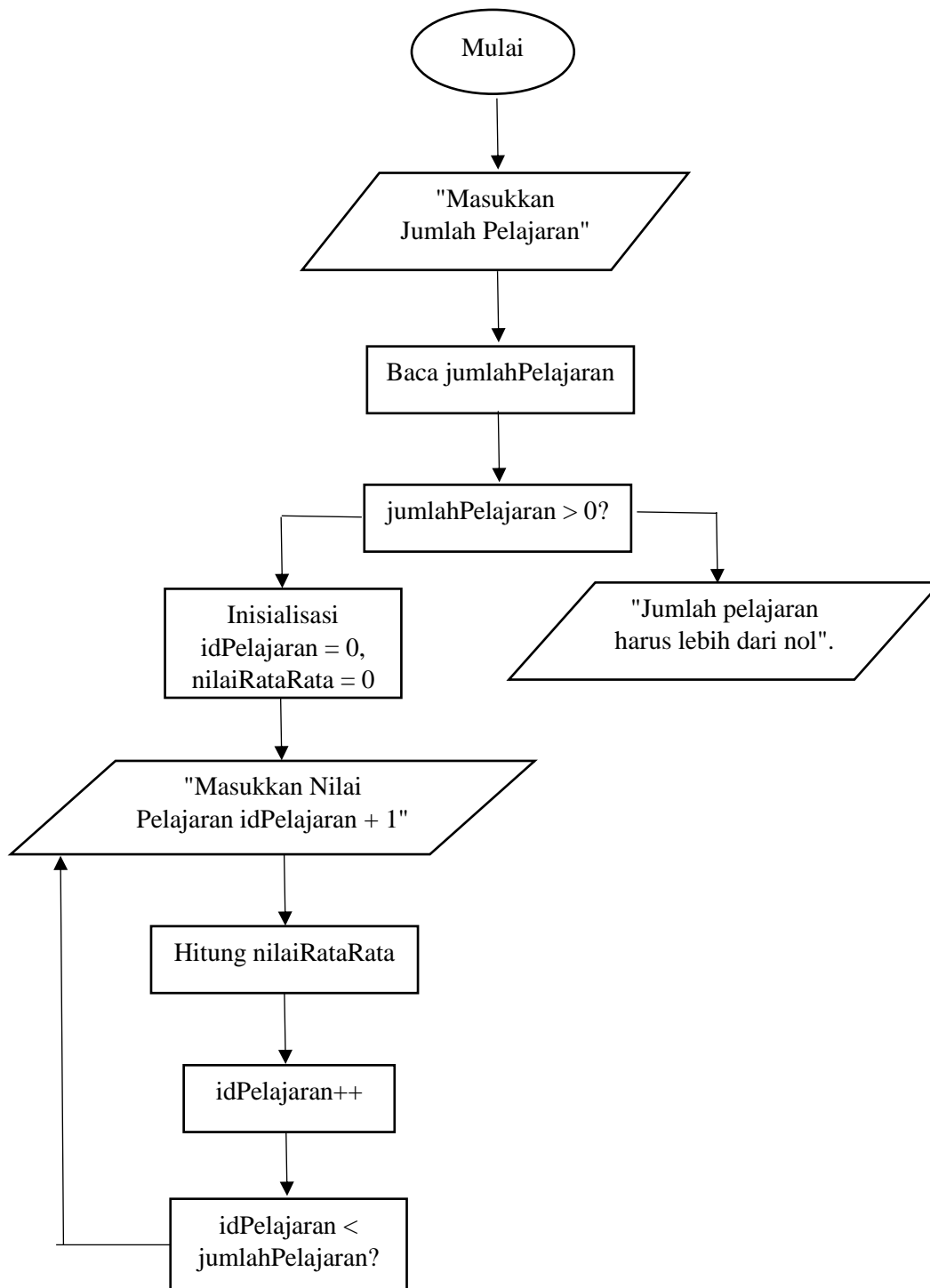
2.4 flowchart 2.1



Flowchart 2.2



Flowchart 2.3



[No.2] Analisis dan Argumentasi

Jawab :

Dalam analisis dua contoh program di atas, kita dapat melihat perbedaan di cara pengulangan dan kontrol aliran saat dijalankan. Pada Contoh 4, penggunaan break berguna untuk menghentikan loop ketika mencapai angka 4, sehingga hanya mencetak angka 1, 2, dan 3. Dengan mengubah kondisi ke $i \% 3 == 0$ dan menggunakan continue akan mencetak angka 1 sampai 6 dan tidak akan melewatkan angka manapun karena continue tidak menghentikan perulangan, tetapi hanya melanjutkan ke iterasi berikutnya. Sementara itu, pada Contoh 5, kita menggunakan loop while untuk mencetak sebuah kata berdasarkan jumlah yang dimasukkan pengguna. Jika jumlahnya 0, tidak ada output yang dihasilkan. Namun, dengan mengubah ke do ... while, kita memastikan bahwa setidaknya kata tersebut akan dicetak sekali, terlepas dari nilai yang dimasukkan.

[No.2] Penyusunan Algoritma dan Kode Program

3) Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

Algoritma contoh 4:

- a) Mulai
- b) inisialisasi $I = 1$
- c) Loop $i \leq 6$
- d) Increment $I++$
- e) Jika i adalah kelipatan 3 (gunakan $i \% 3 == 0$)
- f) Lanjutkan ke iterasi berikutnya (gunakan continue)
- g) Selesai

Algoritma contoh 5:

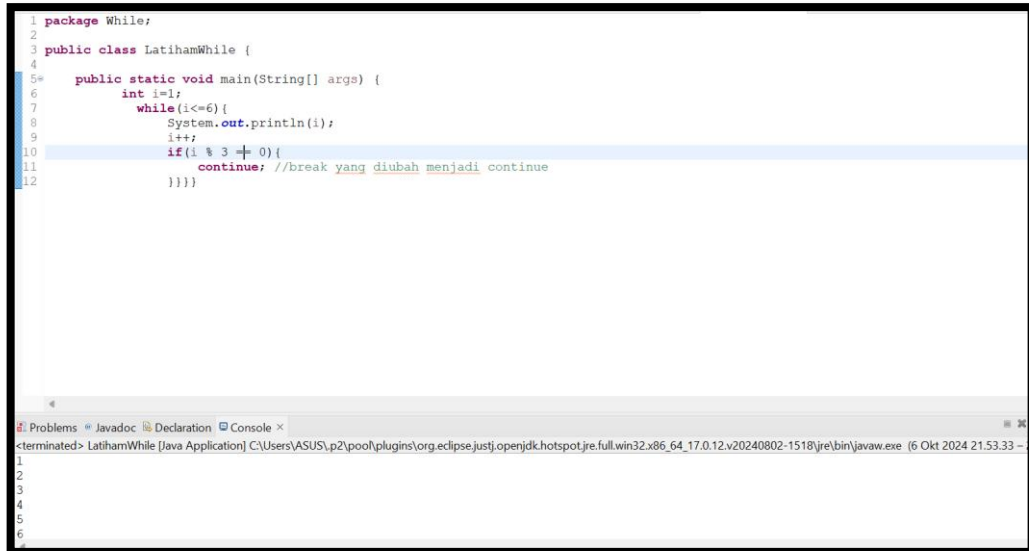
- a. Mulai
- b. Input “masukkan kata yang ingin diulangi”
- c. Input “masukkan jumlah ingin diulangi”
- d. Inisialisasi $I = 0$
- e. Loop do while
- f. Selesai

2. Kode program dan luaran

a) Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

Contoh 4:

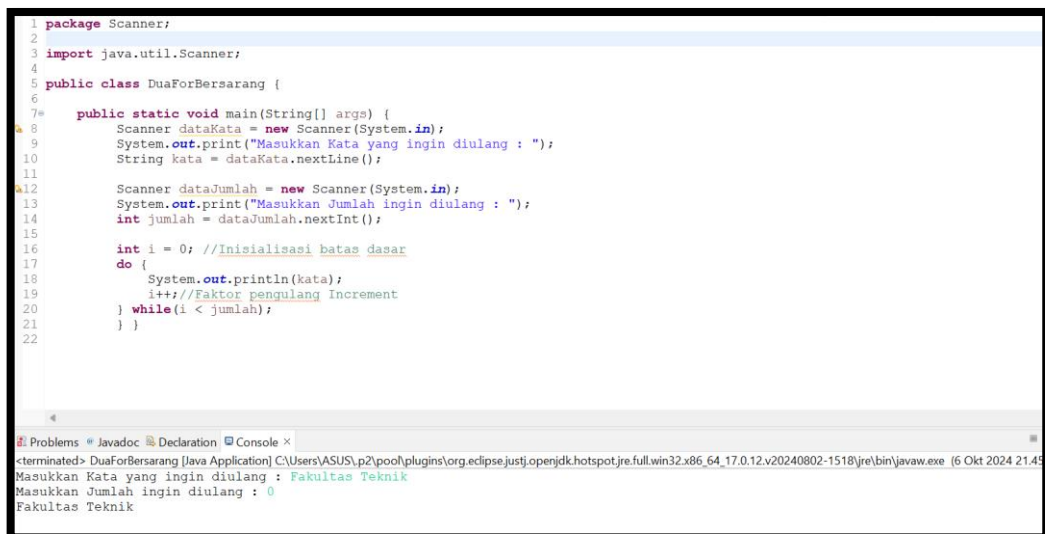


```
1 package While;
2
3 public class LatihanWhile {
4
5     public static void main(String[] args) {
6         int i=1;
7         while(i<=6){
8             System.out.println(i);
9             i++;
10            if(i % 3 == 0){
11                continue; //break yang diubah menjadi continue
12            }
13        }
14    }
15 }
```

Console Output:

```
<terminated> LatihanWhile [Java Application] C:\Users\ASUS\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.12.v20240802-1518\jre\bin\javaw.exe (6 Okt 2024 21:53:33 --
1
2
3
4
5
6
```

Contoh 5:



```
1 package Scanner;
2
3 import java.util.Scanner;
4
5 public class DuaForBersarang {
6
7     public static void main(String[] args) {
8         Scanner dataKata = new Scanner(System.in);
9         System.out.print("Masukkan Kata yang ingin diulang : ");
10        String kata = dataKata.nextLine();
11
12        Scanner dataJumlah = new Scanner(System.in);
13        System.out.print("Masukkan Jumlah ingin diulang : ");
14        int jumlah = dataJumlah.nextInt();
15
16        int i = 0; //Inisialisasi batas dasar
17        do {
18            System.out.println(kata);
19            i++; //Faktor pengulang Increment
20        } while(i < jumlah);
21    }
22 }
```

Console Output:

```
<terminated> DuaForBersarang [Java Application] C:\Users\ASUS\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.12.v20240802-1518\jre\bin\javaw.exe (6 Okt 2024 21:45:
Masukkan Kata yang ingin diulang : Fakultas Teknik
Masukkan Jumlah ingin diulang : 0
Fakultas Teknik
```

b) Analisa luaran yang dihasilkan

jawab :

Luaran sudah sesuai dengan program yang disusun.

[No.2] Kesimpulan

Analisa

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

Jawab : Dari analisis kedua contoh diatas, bisa dilihat bahwa kontrol aliran seperti break dan continue mempengaruhi output. Contoh 4 menggunakan break untuk menghentikan loop pada nilai tertentu, menghasilkan output terbatas, sementara

continue memungkinkan semua nilai dicetak dengan melewati kondisi tertentu.
Di Contoh 5, penggunaan do...while memastikan setidaknya satu output tercetak,

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

Jawab: keputusan diambil berdasarkan hasil analisa. Di Contoh 4, break digunakan untuk membatasi output, sedangkan di Contoh 5, do...while memastikan kata tercetak setidaknya sekali.

Refleksi

Minggu ini, saya belajar tentang percabangan dan perulangan menggunakan for, while, dan do...while. Pengalaman ini meningkatkan pemahaman saya tentang kontrol alur program. Tantangan yang dihadapi adalah memahami kapan menggunakan masing-masing loop, terutama memilih antara while dan do...while. Namun, dengan latihan, saya semakin percaya diri. Pengetahuan ini menunjukkan betapa pentingnya struktur kontrol dalam meningkatkan efisiensi dan keterbacaan kode.