

Template Lembar Kerja Individu dan Kelompok

Nama & NPM Lina Rosalinda	Topik: FOR AND WHILE JAVA	Tanggal: 03 Oktober 2024
[Nomor 1] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variabel</p> <p>Latihan 1</p> <p>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:</p> <pre>0 2 4 6</pre> <p>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:<pre>i = 1; j = 1 i = 1; j = 2 i = 2; j = 1 i = 2; j = 2</pre><p>1.3. Cermati kode contoh 3. Apabila ingin menghasilkan luaran berikut: Luaran berbentuk piramida</p><p>Masukan Input: 7</p><pre> * *** ***** ***** ***** ***** ***** *****</pre><p>Rekomendasikan kode untuk menghasilkan luaran tersebut!</p><p>1.4. Analisa diagram flowchart dari Latihan 1.2 dan 1.3!</p></p>		
[Nomor 1] Analisis dan Argumentasi		
<p>1.1 Penyebab kesalahan pada contoh 1 yaitu tipe data nya pada contoh 1 menggunakan tipe data double sehingga membuat output menjadi double solusinya adalah mengganti tipe data menjadi int. Kemudian pada baris ke-5 angka 15 diganti menjadi 7 saja agar output sesuai dengan yang diinginkan . Pada baris 1 dan 2 bisa dipakai untuk menjalankan program agar sesuai dengan output.</p> <p>1.2 Rekomendasi kode yang saya sarankan yaitu mengganti angka 5 pada baris ke-7 dengan angka 3 sehingga memunculkan output yang sesuai.</p> <p>1.3 Rekomendasi kode yang tepat yaitu merubah arah loop luarnya yang darai menurun dari tinggi ke-1 menjadi menaik dari 1 ke tinggi lalu menambah spasi dan menambah bintang perbarisnya agar output piramida tersebut sesuai ketengah.</p>		

1.4 Pada flowchart 1.2 bisa menggunakan flowchart proses karena sesuai untuk proses alur kerja ataupun Langkah-langkah penyelesaian masalah. Untuk flowchart 1.3 bisa menggambarkan flowchart yang menggambarkan logika untuk mendapatkan keputusan yang jelas dengan menampilkan setiap Langkah dan keputusan.

[Nomor 1] Penyusunan Algoritma dan Kode Program

1) Rancang desain solusi atau algoritma

1.1

1. Mulai

2. Buat kelas dengan nama Forrbersarang

3. Inisialisasi: Set $y = 0$

4. Selama $y < 7$:

4. Jika $y \% 2 == 1$ maka lanjutkan skip ke langkah berikutnya

6. Jika $y == 8$ maka lanjutkan skip ke langkah berikutnya

8. Cetak nilai y dengan `System.out.println(y + " ")`

9. Akhiri program dengan kurung kurawal penutup }

10. Akhiri program

1.2

1. Mulai

2. Buat kelas dengan nama Forrbersarang

3. Buat metode main dengan parameter `public static void main(String[] args)`

4. Deklarasi label pertama dengan membuat perulangan for dengan inisialisasi $i = 1$ dan kondisi $i < 5$:

5. Deklarasi label kedua dengan membuat perulangan for dengan inisialisasi $j = 1$ dan kondisi $j < 3$:

6. Cetak string: `System.out.println("i = " + i + "; j = " + j);`

Jika $i == 2$:

7. Hentikan perulangan dalam (`break` kedua)

8. Akhiri dengan tanda kurung kurawal penutup }

9. Akhiri program

1.3

1. Mulai
2. Baca input tinggi dari pengguna.
3. Untuk setiap baris dari 1 hingga tinggi:
4. Hitung dan cetak spasi.
5. Hitung dan cetak bintang.
6. Pindah ke baris baru setelah mencetak bintang.
7. Akhiri program

1.4

Algoritma flowchart 1.2 dan 1.3

1. Mulai
 2. Inisialisasi $i = 1$.
 3. Impelemtasikan i selama $i < 5$: maka inisialisasi $j = 1$.
 4. Selama $j < 3$:
 5. Cetak i dan j .
 6. Increment j ($j = j + 1$).
 7. Jika $i == 2$, keluar dari loop luar.
 8. Increment i ($i = i + 1$).
 9. Akhiri program
-
1. Mulai
 2. Baca input tinggi dari pengguna.
 3. Untuk setiap baris dari 1 hingga tinggi:
 4. Hitung dan cetak spasi.
 5. Hitung dan cetak bintang.
 6. Pindah ke baris baru setelah mencetak bintang.
 7. Akhiri program.

2) Tuliskan kode program dan luaran

- a) Screenshot/ Capture potongan kode dan hasil luaran

1.1 Kode Program

```
1
2 package perulangan;
3 public class Forrbersarang{
4     public static void main(String[] args) {
5         for (int y = 0; y <=7; y++) {
6             if (y % 2 == 1) { //kondisi 1
7                 // baris kode kosong 1
8             } else if (y == 8) { //kondisi 2
9                 // baris kode kosong 2
10            } else
11                System.out.println(y + " ");
12        } } }
13
```

1.1 Luaran

```
0
2
4
6
```

Output yang dihasilkan sudah sesuai dengan yang diinginkan dan tidak terjadinya error

1.2 Kode Program

```
1 package perulangan;
2
3 public class Forrbersarang {
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                 System.out.println("i = " + i + "; j = " + j);
11             }
12             if ( i == 2) {
13                 // kode yang hilang
14             } } }
15 }
```

1.2 Luaran

```
<terminated> Forrbersarang [Java Application] C:\Users\user\.p2\pool\plugins\org.eclipse
i = 1; j = 1
i = 1; j = 2
i = 2; j = 1
i = 2; j = 2
```

Luaran yang dihasilkan dari program tersebut telah sesuai dengan yang diinginkan.

1.3 Kode Program

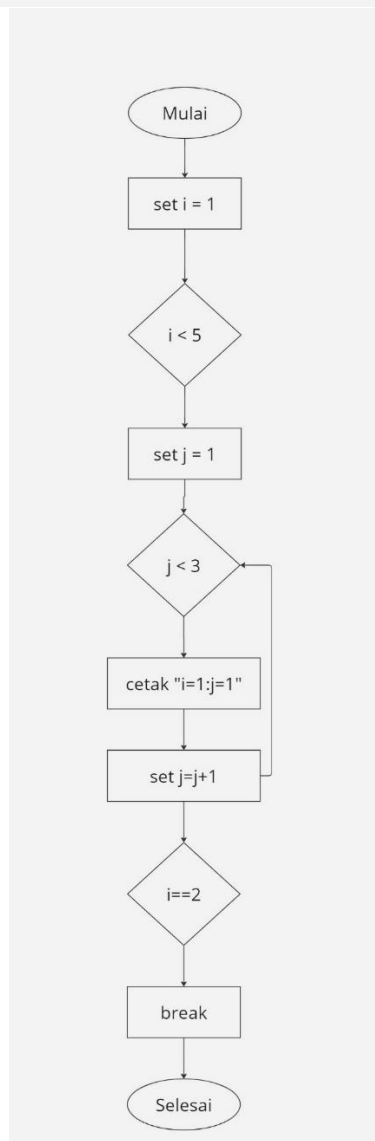
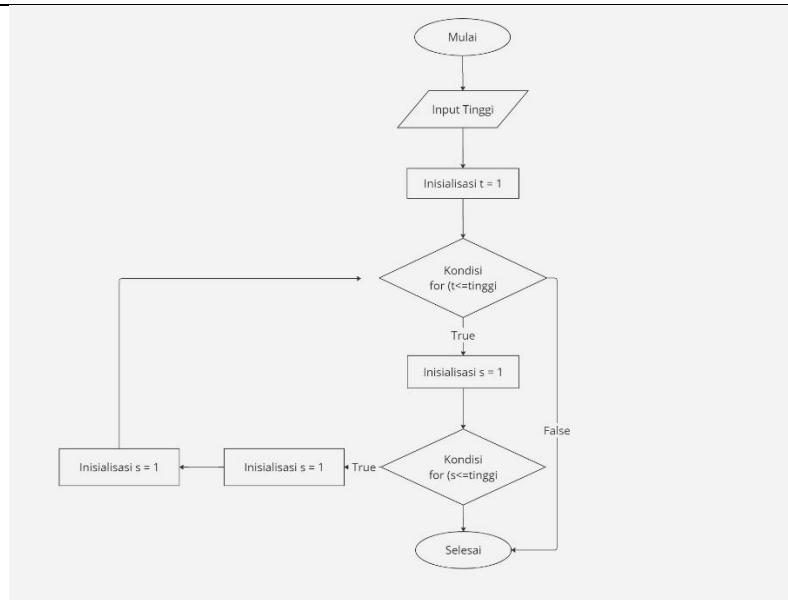
```
Main.java
1 import java.util.Scanner;
2
3 public class perulangan {
4     public static void main(String[] args) {
5
6         Scanner input = new Scanner(System.in);
7         System.out.print("Masukan Input: ");
8         int tinggi = input.nextInt();
9         for (int t = 1; t <= tinggi; t++) {
10
11             for (int s = tinggi; s > t; s--) {
12                 System.out.print(" ");
13             }
14
15             for (int b = 1; b <= (2 * t - 1); b++) {
16                 System.out.print("*");
17             }
18             System.out.println();
19         }
20
21         input.close();
22     }
23 }
```

1.3 Luaran

```
Output
java -cp /tmp/kxLfAlbcUF/perulangan
Masukan Input: 7
*
***
*****
*****
*****
*****
*****
=== Code Execution Successful ===
```

Kode program yang dihasilkan sudah sesuai dengan apa yang diinginkan.

1.4 Flowchart 1.2 dan 1.3



[Nomor 1] Kesimpulan

1) Analisa

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

Kesimpulan yang dapat saya ambil dari penyusunan laporan ini yaitu Setiap algoritma dan flowchart yang disusun untuk masing-masing latihan berhasil merepresentasikan logika dari kode program secara jelas dan sistematis, mulai dari analisis permasalahan dan variabel yang terlibat hingga rekomendasi perbaikan kode, sehingga memastikan bahwa output yang diinginkan tercapai tanpa kesalahan, baik dalam mencetak angka genap, melakukan perulangan bersarang, maupun membentuk piramida bintang dengan spasi yang tepat. Hal ini menunjukkan pemahaman yang baik terhadap konsep pemrograman dan algoritma yang efektif dalam menyelesaikan masalah.

- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?
Saya memilih mengambil keputusan ini karena keputusan ini membantu untuk memastikan bahwa permasalahan-permasalahan yang ada dapat dianalisis lebih lanjut dan dapat membantu dalam mengidentifikasi solusi yang efektif dan memastikan bahwa output yang diinginkan tercapai dengan benar.

Template Lembar Kerja Individu dan Kelompok

Nama & NPM Lina Rosalinda	Topik: FOR AND WHILE JAVA	Tanggal: 03 Oktober 2024
[Nomor 2] Identifikasi Masalah:		
<p>2) Uraikan permasalahan dan variabel</p> <p>Latihan 2</p> <p>2.1. Ubahlah baris kode pada Contoh 4 //Ubah1 menjadi <code>if(i % 3 == 0){ ◇ running, periksa hasilnya</code> //Ubah2 menjadi <code>continue; ◇ running, periksa hasilnya</code> Evaluasi perbandingan luaran sebelum dan setelah diubah! Simpulkan maksud dari perubahan tersebut!</p> <p>2.2. Cermati Contoh 5. Periksa luaran, bila ketika di eksekusi, jumlah yang diulang = 0! Evaluasi luaran, bila kode diubah menjadi <code>do ... while</code> dengan masukan sama jumlah yang diulang = 0. Simpulkan perbedaan <code>while</code> dan <code>do ... while</code>!</p> <p>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!</p> <p>2.4. Rancang diagram flowchart dari Latihan 2.1, Latihan 2.2, dan Latihan 2.3!</p>		
[Nomor 1] Analisis dan Argumentasi		
<p>2.1 Pada program 2.1 luaran yang sebelumnya 1 2 3 pada saat <code>if(i % 3 == 0){</code> dan <code>break</code> diubah menjadi <code>continue</code> maka luaran berubah menjadi 1 2 3 4 5 6</p> <p>2.2 Pada program 2.2 bila kode diubah menjadi <code>do</code> maka program akan mengeksekusi minimal sekali eksekusi sedangkan <code>while</code> langsung menginput program sehingga luaran <code>while</code> dan <code>do</code> akan berbeda</p> <p>2.3 Pada program 2.3 dapat menggunakan kode yang mengimplmentasikan inisiasi id pelajaran, inisiasi nilai pelajaran dan nilai rata-rata dengan kode program <code>while</code>.</p> <p>2.4 Flowchart yang digunakan yaitu flowchart yang menjelaskan bagaimana langkah-langkah proses pemecahan masalah dari program sehingga menghasilkan luaran yang diinginkan.</p>		
[Nomor 1] Penyusunan Algoritma dan Kode Program		
<p>1 Rancang desain solusi atau algoritma</p> <p>1.1</p> <p>1. Mulai</p> <p>2. Inisialisasi variabel i dengan nilai 2</p>		

3. Lakukan perulangan selama nilai I lebih kecil atau sama dengan 6 lalu cetak i
4. Periksa apakah nilai i jika nilai habis dibagi 3 yang artinya $i \% 3 \neq 0$ maka lanjutkan ke Langkah berikutnya dengan menggunakan continue
5. Tambahkan nilai i dengan 1 setiap literasi berikutnya
6. Perulangan akan terus berhenti Ketika nilai I sudah lebih besar
7. Akhiri program

2.2

1. Mulai program
2. Inisialisasi variabel scanner untuk menerima input dari pengguna
3. Pengguna memasukkan kata yang ingin diulang kemudian simpan input dalam variabel kata
4. Minta pengguna untuk memasukkan jumlah perulangan, simpan input dalam variabel jumlah
5. Inisialisasi variabel I dengan 0
6. Lakukan perulangan selama nilai i kurang dari jumlahnya
7. Cetak kata yang sudah diinputkan oleh pengguna lalu tambahkan nilai I dengan 1
8. Perulangan akan berhenti Ketika nilai udah sama atau lebih esar dari jumlah
9. Akhiri program

2.3

1. Mulai program.
2. Inisialisasi variabel idPelajaran untuk menghitung indeks pelajaran, dan nilaiRataRata untuk menyimpan nilai rata-rata, keduanya diset dengan nilai awal 0.
3. Minta pengguna memasukkan jumlah pelajaran, simpan hasil input ke dalam variabel jumlahPelajaran.
4. Lakukan perulangan selama nilai idPelajaran lebih kecil dari jumlahPelajaran:
5. Minta pengguna untuk memasukkan nilai pelajaran ke-idPelajaran+1.
6. Simpan nilai pelajaran yang dimasukkan ke dalam variabel nilaiPelajaran.
7. Hitung rata-rata sementara dengan menambahkan nilai pelajaran yang baru dimasukkan ke dalam total rata-rata, lalu bagi dengan jumlah pelajaran yang telah diinputkan.
8. Tambahkan idPelajaran dengan 1 untuk beralih ke pelajaran berikutnya.
9. Setelah semua nilai pelajaran diinputkan, cetak hasil perhitungan nilai rata-rata.
10. Tutup scanner untuk menghentikan input.
11. Program selesai.

2.4 Algoritma Flowchart 2.1

1. Mulai
2. Inisialisai variabel i
3. Periksa apakah $i \leq 6$?
4. Jika ya lanjutkan tampilkan i jika tidak maka flowchart selesai
5. Proses apakah $i = i + 1$
6. Periksa apakah $i \% 3 == 0$?
7. Jika ya lanjut ke continue dan jika tidak Kembali ke pemeriksaan $i \leq 6$?
8. Setelah diperiksa lagi dan menyatakan tidk maka flowchart berakhir

Algoritma Flowchart 2.2

1. Mulai
2. Inisialisasi scanner
3. Input kata yang ingin diulang
4. Input jumlah perulangan
5. Inisialisasi variabel i
6. Periksa apakah i=jumlah?, jika tidak maka flowchart selesai
7. Tampilkan kata dari input
8. Increment i
9. Kembali ke pengecekan
10. Selesai
11. Lalu ulangi memeriksa dari apakah i=jumlah?

Algoritma Flowchart 2.3

1. Mulai
2. Inisialisasi scanner
3. Input jumlah pelajaran
4. Inisialisasi variabel idPelajaran
5. Inisialisasi nilai rata-rata
6. Periksa apakah pelajaran=jumlah pelajaran,
7. Jika ya input nilai pelajaran dan jikatidak tampilkan rata-rata lalu selesai
8. Hitung nilai rata-rata
9. Incrememt id Pelajaran
10. Tampilkan rata-rata
11. selesai

3) Tuliskan kode program dan luaran

- b) Screenshot/ Capture potongan kode dan hasil luaran

2.1 Kode Program

```

Main.java
1- public class ContohWhile{
2- public static void main(String[] args) {
3-     int i=1;
4-     while(i<=6){
5-         System.out.println(i);
6-         i++;
7-         if(i % 3 == 0){
8-             continue; //ubah1
9-         }
10-    }
11- }

```

2.1 Luaran

```

Output
Clear

java -cp /tmp/QYqnHmAXGv/ContohWhile
1
2
3
4
5
6

=== Code Execution Successful ===

```

Luaran yang dihasilkan sudah sesuai dengan soal yang diinginkan.

2.2 Kode program

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

2.2 Luaran

```
Output
Clear

java -cp /tmp/9r00TGNbTj/ForBersarang
Masukkan Kata yang ingin diulang : fakultas teknik
Masukkan Jumlah ingin diulang : 0
fakultas teknik

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

2.2Luaran

Luaran yang dihasilkan sudah sesuai sehingga tidak adanya terjadi error

2.3 Kode program

```
Main.java
1- import java.util.Scanner;
2
3- public class RataRataNilai {
4-     public static void main(String[] args) {
5         // Inisialisasi variabel
6         int idPelajaran = 0;
7         double nilaiRataRata = 0.0;
8         Scanner scanner = new Scanner(System.in);
9
10        // Minta pengguna untuk menuliskan jumlah pelajaran
11        System.out.print("Masukkan jumlah pelajaran: ");
12        int jumlahPelajaran = scanner.nextInt();
13
14        // Ketika idPelajaran lebih kecil dari jumlah pelajaran
15        while (idPelajaran < jumlahPelajaran) {
16            // Minta pengguna untuk menuliskan nilai pelajaran
17            System.out.print("Masukkan nilai pelajaran ke-" + (idPelajaran + 1) + ": ");
18            double nilaiPelajaran = scanner.nextDouble();
19
20            // Hitung nilai rata-rata
21            nilaiRataRata = (nilaiPelajaran + nilaiRataRata * idPelajaran) / (idPelajaran + 1);
22
23            // Tambah satu ke idPelajaran
24            idPelajaran++;
25        }
26
27        // Tampilkan nilai rata-rata
28        System.out.printf("Nilai rata-rata: %.2f\n", nilaiRataRata);
29
30        // Tutup scanner
31        scanner.close();
32    }
33 }
```

2.3 Luaran

Output

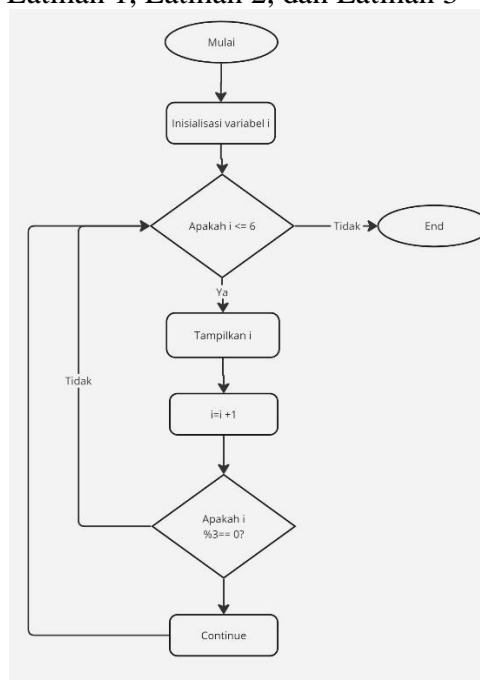
[Clear](#)

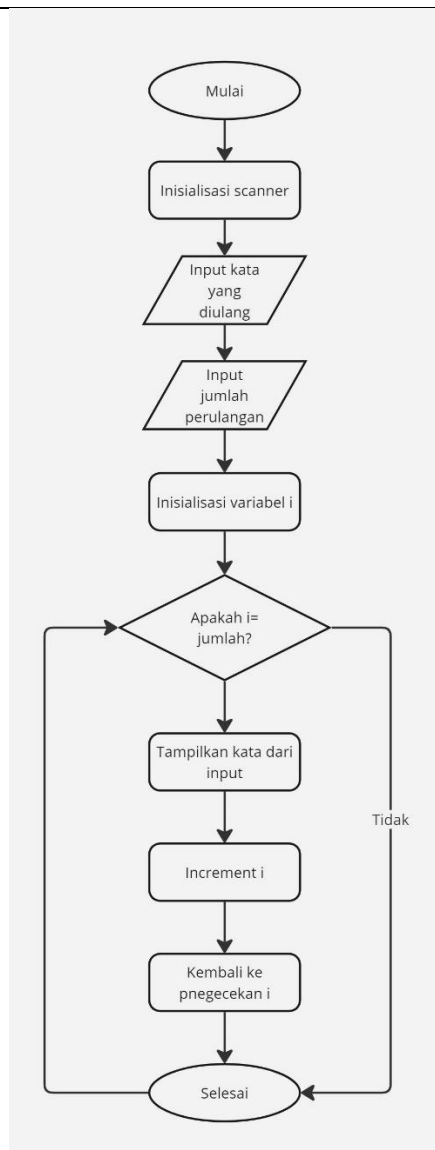
```
java -cp /tmp/a8BPkqaK8S/RataRataNilai  
Masukkan jumlah pelajaran: 8  
Masukkan nilai pelajaran ke-1: 80  
Masukkan nilai pelajaran ke-2: 85  
Masukkan nilai pelajaran ke-3: 87  
Masukkan nilai pelajaran ke-4: 90  
Masukkan nilai pelajaran ke-5: 88  
Masukkan nilai pelajaran ke-6: 86  
Masukkan nilai pelajaran ke-7: 85  
Masukkan nilai pelajaran ke-8: 83  
Nilai rata-rata: 85.50
```

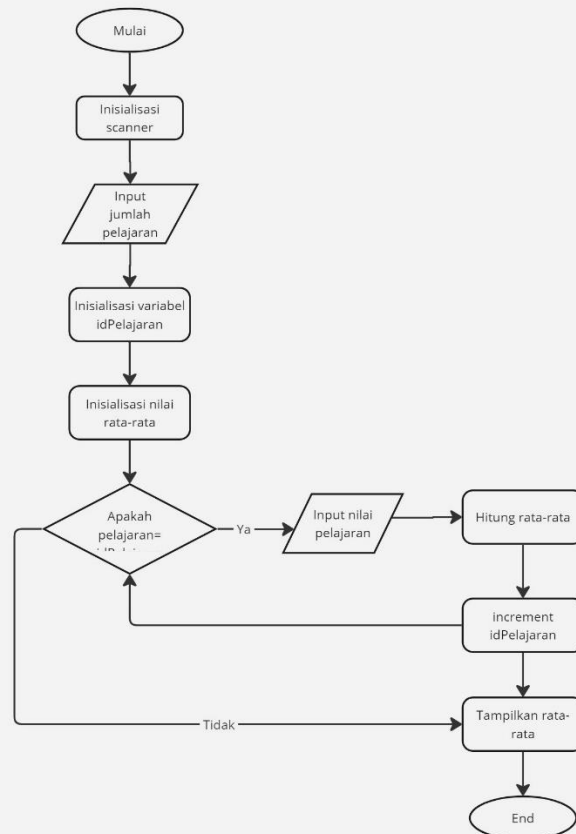
=== Code Execution Successful ===

Luaran yang dihasilkan sudah menuruti perintah dari soal sehingga sesuai dengan apa yang diinginkan.

2.4 Flowchart Latihan 1, Latihan 2, dan Latihan 3







[Nomor 1] Kesimpulan

2 Analisa

- c) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program! Kesimpulan yang dapat saya ambil dari penyusunan laporan ini yaitu alam ketiga program ini, penggunaan struktur kontrol seperti if, continue, do...while, dan while sangat penting untuk menentukan alur logika program. Flowchart berfungsi sebagai panduan visual untuk memahami dan merancang algoritma secara sistematis, memastikan semua langkah pemecahan masalah terstruktur dengan baik. Dengan begitu pemrograman menjadi lebih terorganisir dan efisien.
- d) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini? Saya memilih mengambil keputusan ini karena keputusan ini mempertimbangkan tujuan yang ingin dicapai oleh program, efektivitas penggunaan struktur kontrol yang tepat, dan kemudahan dalam memahami alur logika melalui diagram alir. Ini bertujuan agar program tidak hanya berfungsi dengan baik, tetapi juga memberikan pengalaman yang baik bagi pengguna serta memudahkan pengembangan di masa depan.

Refleksi:

Hasil dari penyusunan laopran kali ini yaitu memberikan saya beberapa pemahaman pentingnya merancang program dengan baik. Memikirkan tujuan, cara kerja yang efisien, dan pengalaman pengguna membuat program lebih bermanfaat. Selain itu, menggunakan diagram alir membantu menjelaskan ide-ide kita dengan lebih jelas. Semua ini tidak hanya meningkatkan kualitas program, tetapi juga mempersiapkan kita untuk pengembangan di masa depan. Pada

akhirnya, perencanaan yang matang dan pemikiran kritis sangat penting dalam setiap proyek pemrograman.