Jawaban:

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Moza Violetha Salsabilla**  **G1F024028** | **For and 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!

**[No.1] Analisis dan Argumentasi**

* 1. Pada contoh 1, masih terdapat kesalahan. Kesalahan tersebut dapat diperbaiki dengan cara mengganti tempat y = 0 di sebelah kiri y <= 15. Lalu, agar menghasilkan luaran berupa angka 0, 2, 4, dan 6. Maka, saya merekomendasikan untuk menggunakan kata kunci continue pada kondisi if (y % 2 == 1) untuk mengabaikan semua bilangan ganjil pada kondisi tersebut. Kemudian, saya merekomendasikan untuk menambahkan break pada kondisi else if (y == 8) untuk memastikan bahwa angka 8 tidak akan dicetak. Berikut adalah perbaikan kode dari contoh 1

public class ContohFor {

public static void main(String[] args) {

for (double y = 0; y <= 15; y++) {

if (y % 2 == 1) { // kondisi 1

continue; // baris kode kosong 1

} else if (y == 8) { // kondisi 2

break; // baris kode kosong 2

} else {

System.out.println(y + " ");

}

}

}

}

* 1. Untuk menghasilkan luaran seperti yang diminta, maka saya merekomendasikan kode program sebagai berikut:

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) {

continue pertama; // kode yang hilang

}

}

}

}

* 1. Berikut adalah rekomendasi kode dari saya untuk luaran tersebut:

public class Bintang {

public static void main(String[] args) {

int n = 7;

for (int i = 1; i <= n; i++) {

//print spaces

for (int j = i; j < n; j++) {

System.out.print(" ");

}

//print bintang

for (int j = 1; j <= (2 \* i - 1); j++) {

System.out.print("\*");

}

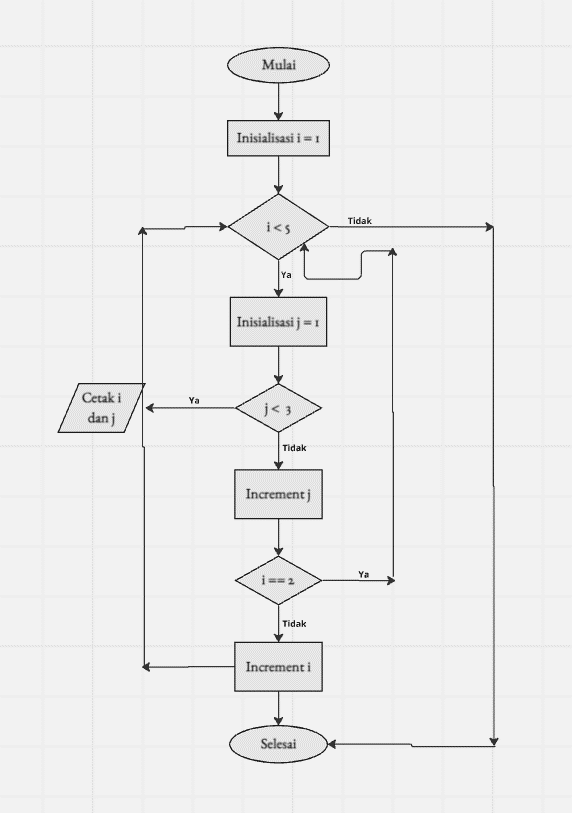
System.out.println();

}

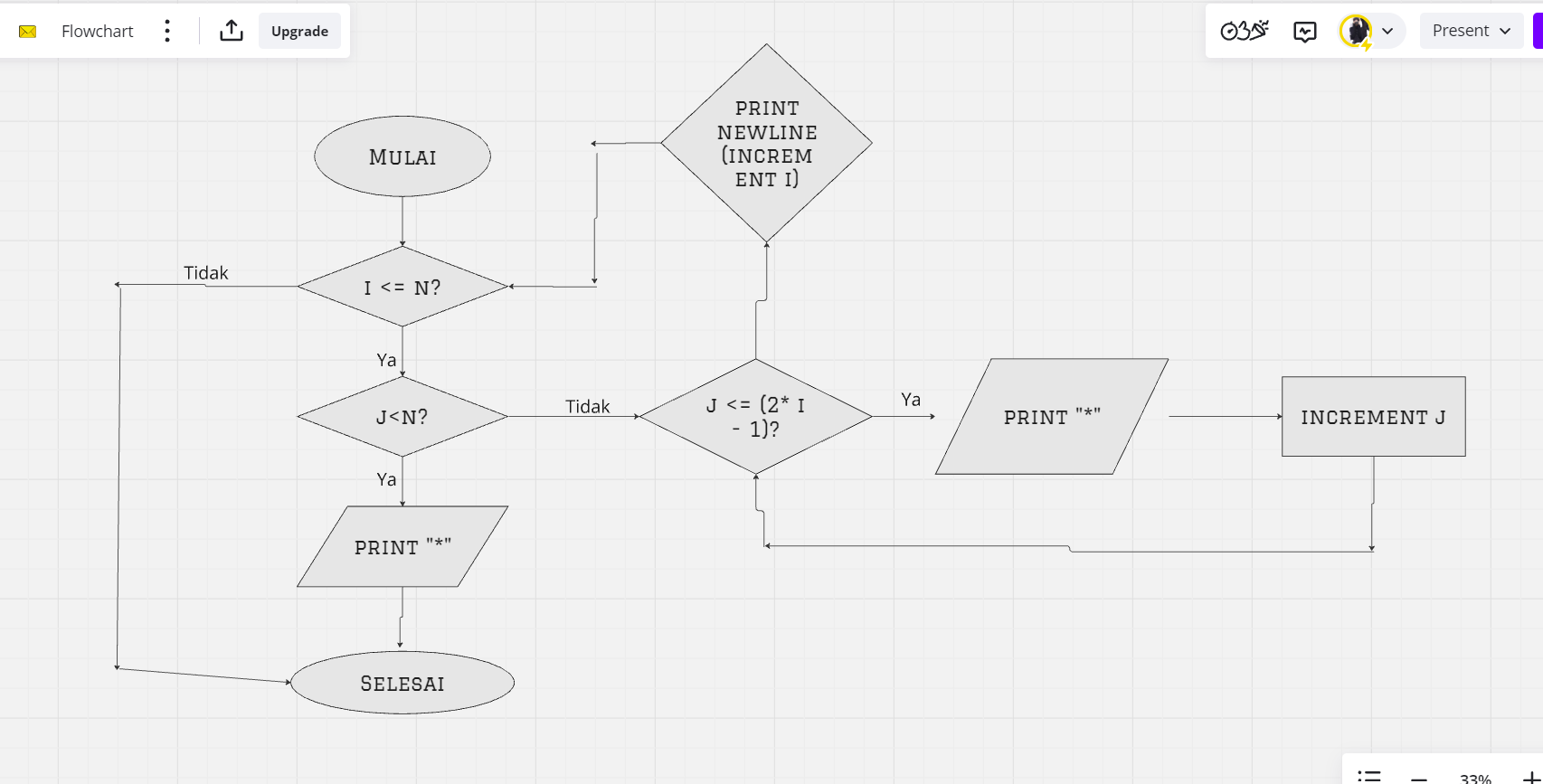
}

}

* 1. Berikut adalah flowchart untuk Latihan 2.3



Berikut adalah flowchart untuk Latihan 1.3



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

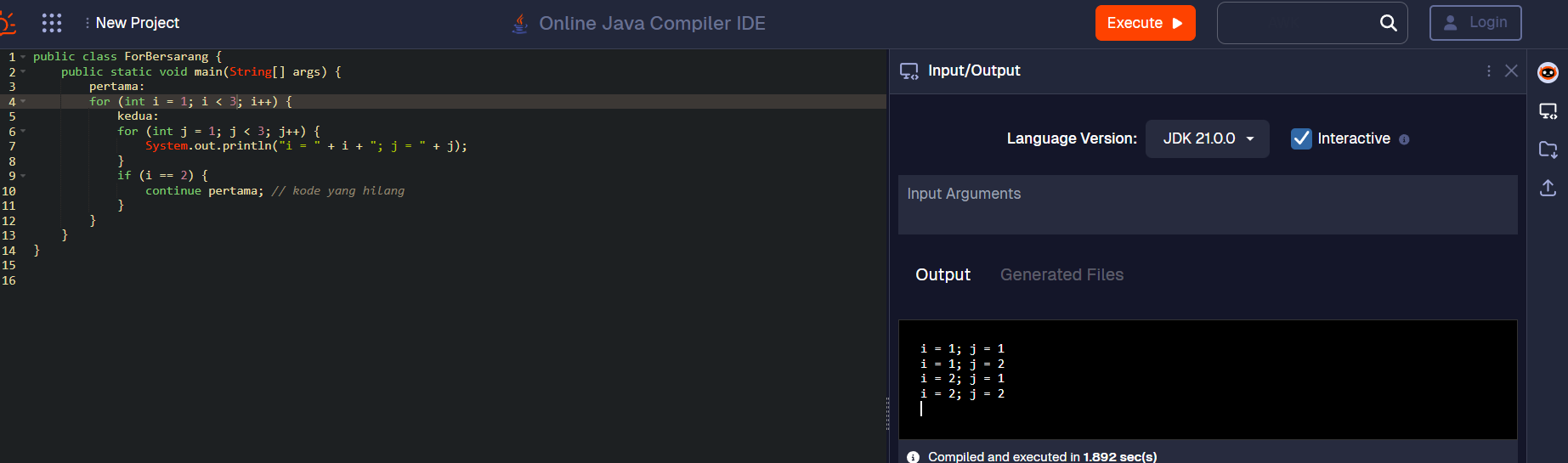
1. Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Misalkan algoritma memasak mi instan:

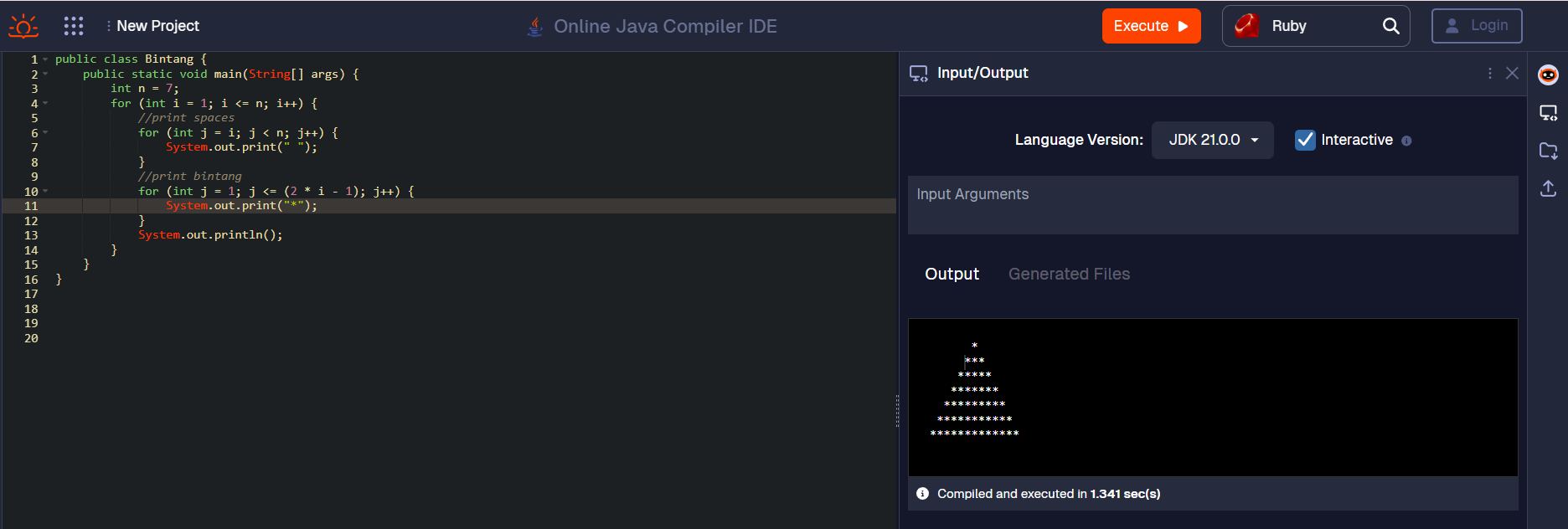
1. Mulai
2. Inisialisasi n
3. Buat loop (perulangan)
4. Cetak newline
5. Selesai
6. Kode program dan luaran
7. Screenshot/ Capture potongan kode dan hasil luaran



Gambar kode program dan luaran 1.1



Gambar kode program dan luaran 1.2



Gambar kode program dan luaran 1.3

1. Analisa luaran yang dihasilkan   
   Contoh:

Luaran sudah sesuai dengan program yang disusun. Luaran telah sesuai dengan hasil yang diminta.

**[No.1] Kesimpulan** **Analisa**

1. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
2. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban :

Pada program itu saya menggunakan continue pada kondisi jika y adalah bilangan ganjil (hasil y % sama dengan 1), maka program akan menjalankan perintah continue, yang berarti tidak akan mencetak nilai y dan akan melanjutkan ke nilai y berikutnya. Sedangkan, saya menggunakan perintah break pada kondisi, jika y sama dengan 8, program akan menjalankan perintah break, yang berarti loop akan dihentikan sepenuhnya, dan eksekusi akan melanjutkan ke baris kode setelah loop.

**[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 = (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!

**[No.2] Analisis dan Argumentasi**

2.1 Sebelum diperbaiki luaran dari kode program tersebut adalah 1,2, dan 3. Karena, Ketika i mencapai 4, perintah break dijalankan, menghentikan loop.Sedangkan, setelah mengganti syntax menjadi if(i % 3 == 0) {, dengan perubahan ini, tidak ada perubahan dalam hasil, karena i hanya mencapai 3 sebelum loop berakhir. Ketika I menjadi 3, tidak ada tindakan yang diambil, dan setelah itu, i meningkat menjadi 4. Lalu, dengan mengganti break menjadi continue dengan perubahan ini, ketika I mencapai 3, perintah continue dijalankan, yang menyebabkan loop melanjutkan ke iterasi berikutnya tanpa mencetak nilai i (dalam hal ini, tidak mencetak 4). Loop kemudian melanjutkan untuk mencetak 5 dan 6. Selain itu, kita harus menambahkan kurung kurawal di baris akhir kode program agar program dapat di run.

2.2 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();

System.out.print("Masukkan Jumlah ingin diulang: ");

int jumlah = dataKata.nextInt();

int i = 0; // Inisialisasi batas dasar

while(i < jumlah) {

System.out.println(kata);

i++; // Faktor pengulang Increment

}

}

}

Saat pengguna memasukkan 0 sebagai nilai, maka program tidak akan mencetak apapun karena kondisi while tidak terpenuhi (0 tidak kurang dqari 0). Maka dari itu, outputnya akan kosong. Saat kode diubah menjadi do.. while maka kode programnya sebagai berikut:

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();

System.out.print("Masukkan Jumlah ingin diulang: ");

int jumlah = dataKata.nextInt();

int i = 0; // Inisialisasi batas dasar

do {

System.out.println(kata);

i++; // Faktor pengulang Increment

} while(i < jumlah);

}

}

Jika pengguna memasukkan 0 sebagai nilai, maka output akan tetap kosong, karena perintah do.. while.. akan memeriksa kondisi perulangn setelah eksekusi minimal satu kali. Tetapi, jika nilai adalah 0, maka tidak akan ada iterasi yang dilakukan dalam loop tersebut. Perbedaan antara while dan do while adalah bahwa while memeriksa kondisi sebelum program di eksekusi, sedangkan do.. while menjalankan iterasi setidaknya sekali sebelum memeriksa kondisi belakangan.

2.3 Berikut adalah kode program yang dapat saya rekomendasikan

import java.util.Scanner;

public class RataRataNilai {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Inisialisasi

int jumlahPelajaran;

double totalNilai = 0;

// Minta pengguna untuk menuliskan jumlah pelajaran

System.out.print("Masukkan jumlah pelajaran: ");

jumlahPelajaran = scanner.nextInt();

// Minta pengguna untuk menuliskan nilai pelajaran

for (int i = 0; i < jumlahPelajaran; i++) {

System.out.print("Masukkan nilai pelajaran ke-" + (i + 1) + ": ");

double nilai = scanner.nextDouble();

totalNilai += nilai; // Akumulasi total nilai

}

// Hitung dan tampilkan nilai rata-rata

double rataRata = totalNilai / jumlahPelajaran;

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

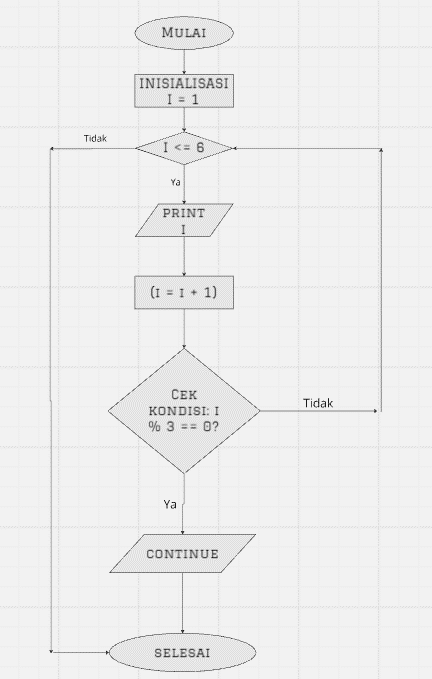
// Menutup scanner

scanner.close();

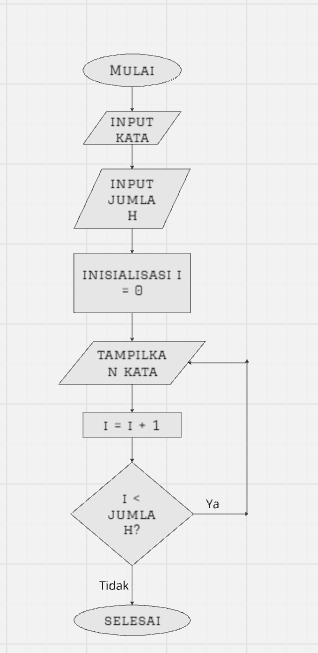
}

}

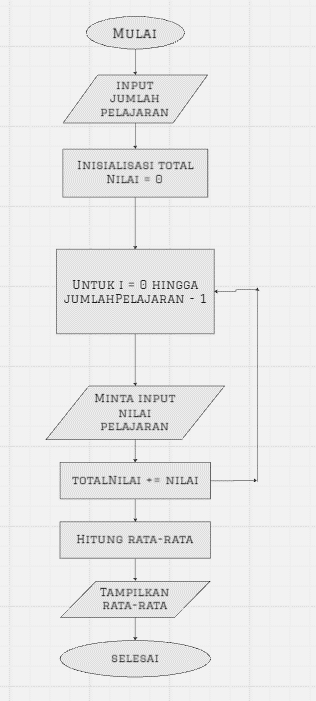
2.4 Berikut adalah flowchart 2.1



Berikut adalah flowchart 2.2



Berikut adalah flowchart 2.3



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

1. Algoritma 2.1

Algoritma adalah langkah-langkah penyelesaian masalah.  
(a) Mulai

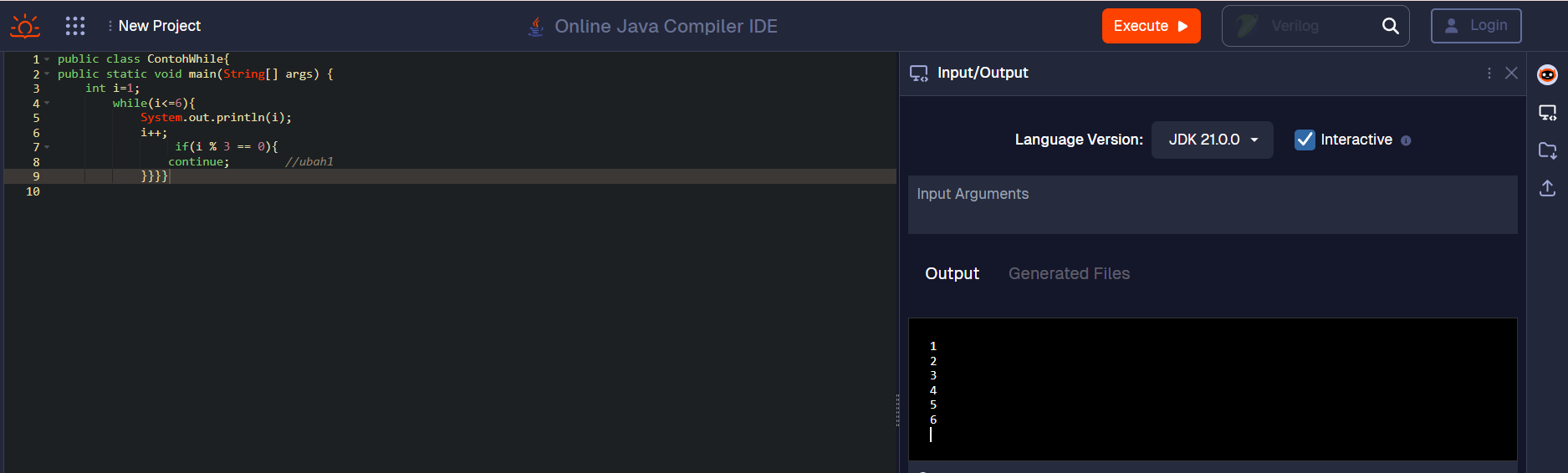
1. Inisialisasi i
2. Buat loop (perulangan)
3. Cetak luaran
4. Selesai
5. Algoritma 2.2

Algoritma adalah langkah-langkah penyelesaian masalah.  
(a) Mulai

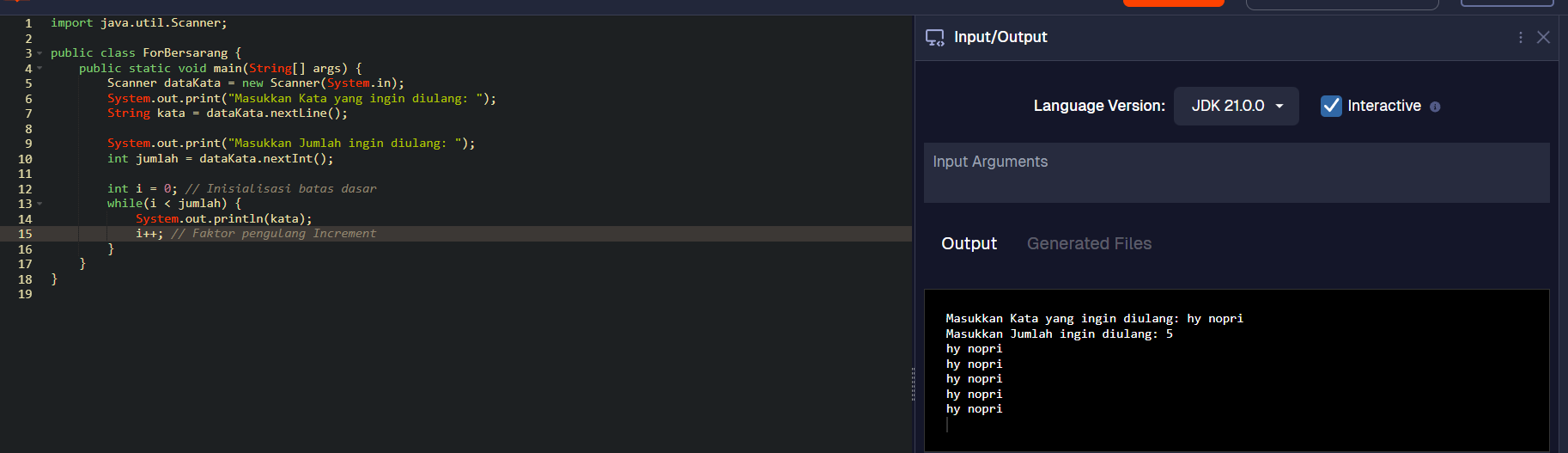
1. Deklarasi variabel
2. **Minta Input dari Pengguna**
3. Minta Input Jumlah
4. Mulai perulangan
5. Akhiri perulangan
6. Selesai
7. Algoritma 2.3

Algoritma adalah langkah-langkah penyelesaian masalah.  
(a) Mulai

1. Deklarasi variabel
2. **Minta Input dari Pengguna**
3. Mulai loop
4. Hitung rata- rata
5. Tampilkan hasil
6. Selesai
7. Kode program dan luaran
8. Screenshot/ Capture potongan kode dan hasil luaran



Gambar kode program dan luaran 2.1



Gambar kode program dan luaran 2.2



Gambar kode program dan luaran 2.3

1. Analisa luaran yang dihasilkan   
   Luaran sudah sesuai dengan program yang disusun. Luaran telah sesuai dengan hasil yang diminta.

**[No.2] Kesimpulan** **Analisa**

1. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
2. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban :

Pada program tersebut, saya mengganti beberapa syntax seperti i%3 == 0 dan juga menambahkan perintah continue pada kode program agar bisa dijalankan. Lalu, saya berhasil mengetahui bahwa perbedaan antara while dan do while adalah bahwa while memeriksa kondisi sebelum program di eksekusi, sedangkan do.. while menjalankan iterasi setidaknya sekali sebelum memeriksa kondisi belakangan.

Refleksi

Pada subbab for while ini saya mempelajari perulangan do, do while, for, dan juga for while. Saya juga belajar banyak mengenai algoritma dan flowchart. Saya juga berhasil membuat kode program untuk menghitung rata rata dari nilai mata Pelajaran.