

**Lembar Kerja Individu**

<b>Nama &amp; NPM</b>	<b>Topik:</b>	<b>Tanggal:</b>
<b>Rayhan Prabowo G1F024022</b>	<b>For dan While</b>	<b>10 Oktober 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:

### 1) Uraikan permasalahan dan variabel

#### 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 kode pada contoh 1.1 dapat diatasi dengan cara :

- Format yang benar untuk loop for adalah : for (inisialisasi; kondisi; increment)
- for (double y <= 15; y=0; y++) diperbaiki menjadi for (double y = 0 ; y <= 15; y++)\_\_\_\_\_

### 2) Alasan solusi ini karena dengan perbaikan program diatas, program akan menghasilkan output yang diinginkan.

### 3) Perbaiki kode program dengan cara :

mengubah kesalahan sintaks pada contoh 1,berikut kode program perbaikan pada 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 + " ");
    }
}

```

4) Untuk mendapatkan luaran yang diinginkan sesuai dengan soal 1.2 maka sebaiknya penggunaan kode yang tepat menggunakan kode break pertama karena kita perlu menghentikan seluruh proses loop setelah i mencapai nilai 2. Oleh itu, pada kondisi if (i == 2), kita menggunakan break pertama. Jika kita menggunakan kode lain seperti :

- a. Break; (tanpa nama) – hanya akan menghentikan loop dalam
- b. Break kedua; - juga hanya menghentikan loop dalam
- c. Continue pertama; - akan melanjutkan ke literasi berikutnya dari loop luar, yang tidak kita inginkan.

#### [No.1 ] Penyusunan Algoritma, Kode Program, dan Diagram Flowchart

##### 1) Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

Algoritma membuat program yang menghasilkan luaran segitiga:

- (a) Mulai
- (b) Tampilkan “masukkan input”
- (c) Baca input dari pengguna, simpan ke variabel tinggi
- (d) Untuk i= 1 sampai tinggi, lakukan :
  - a. Untuk j 1 sampai( tinggi-i), lakukan :
    - i. Cetak satu karakter spasi
  - b. Untuk k = 1 sampai (2 x i-1), lakukan:
    - i. Cetak karakter bintang “\*”
  - c. Pindah ke garis baru dan cetak new line
  - d. Pindah ke baris baru(cetak newline)
  - e. Selesai.

##### 2) Kode program dan luaran

###### a) Kode Program

- a. Kode program perbaikan :

```

1  public class ContohFor {
2      public static void main(String[] args) {
3          for (double y = 0; y <= 15; y++) {
4              if (y % 2 == 1) {          //kondisi 1
5                  continue;           // baris kode kosong 1
6              } else if (y == 8) {      //kondisi 2
7                  break;               // baris kode kosong 2
8              } else
9                  System.out.println(y + " ");
10         }
11     }
12 }

```

b. Kode program piramida bintang

```

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(s:"Masukan Input: ");
8          int tinggi = input.nextInt(); // Mendapatkan Input Dari User
9
10         for (int i = 1; i <= tinggi; i++) {
11             // Loop untuk spasi
12             for (int j = 1; j <= tinggi - i; j++) {
13                 System.out.print(s:" ");
14             }
15
16             // Loop untuk bintang
17             for (int k = 1; k <= (2 * i) - 1; k++) {
18                 System.out.print(s:"*");
19             }
20
21             System.out.println(); // Membuat baris baru
22         }
23     }
24 }

```

b) Luaran

a. Luaran kode program perbaikan :

```

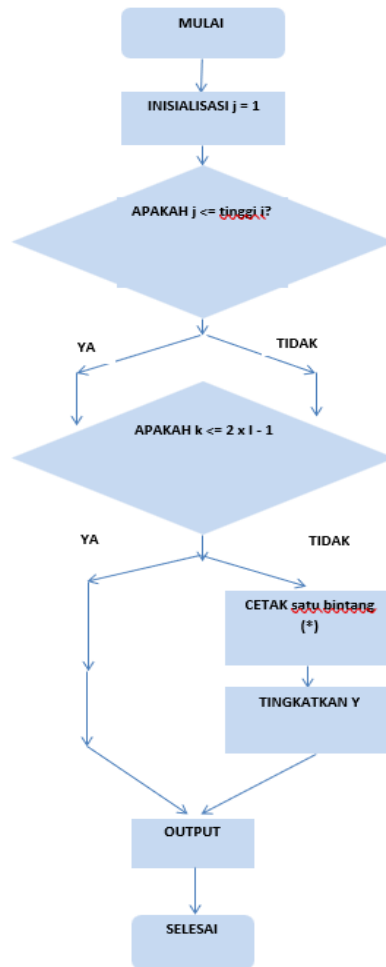
PS E:\TIPE DATA> & 'C:\Program Files\Java\jdk1.8.0_341\bin\j
Storage\fa5bb4322bcb69e63c79fb4c679be663\redhat.java\jdt_ws\T
0.0
2.0
4.0
6.0
PS E:\TIPE DATA>

```

b. Luaran kode program piramida bintang :

c) Analisa luaran yang dihasilkan  
Luaran yang dihasilkan sudah sesuai dengan perbaikan kode pada soal yang diberikan, kode yang mulanya pada soal ada baris yang kosong dan tanpa break atau continue. Pada kode program piramida bintang juga jika di inputkan nilai berapa pun akan menghasilkan luaran piramida berbentuk menyerupai segitiga dengan kumpulan symbol bintang.

b) Flowchart code program piramida



## [No.1] Kesimpulan

### 1) Analisa

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

Jawab :

- Program untuk membuat segitiga bintang dengan menggunakan struktur kontrol for bersarang merupakan cara yang efektif untuk memahami konsep pengulangan dalam pemrograman. Dalam program ini, kita memanfaatkan dua loop for: satu loop luar yang mengatur jumlah baris segitiga dan satu loop dalam yang bertugas mencetak bintang pada setiap baris. Setiap kali loop luar beriterasi, kita menentukan jumlah bintang yang akan dicetak berdasarkan nomor baris saat itu. Sebagai contoh, pada baris pertama, kita mencetak satu bintang, pada baris kedua dua bintang, dan seterusnya hingga mencapai tinggi segitiga yang diinginkan. Loop dalam bertanggung jawab untuk mencetak bintang-

bintang tersebut, dan biasanya diakhiri dengan karakter newline untuk memindahkan kursor ke baris berikutnya.

- b) Dasar pengambilan keputusan dalam konteks program ini melibatkan beberapa pertimbangan. Pertama, penting untuk menganalisis tujuan dari program, yaitu untuk mencetak pola segitiga bintang yang simetris. Keputusan untuk menggunakan loop bersarang diambil karena metode ini memungkinkan kontrol yang lebih baik atas jumlah elemen yang dicetak pada setiap baris. Selain itu, pemilihan struktur kontrol ini juga mempertimbangkan kemudahan pemahaman dan implementasi, terutama bagi pemula dalam pemrograman.

## [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

1) Evaluasi perubahan kode program pada contoh 4 :

- a. Pada perubahan pertama, saya mengganti kode `if(i==)` dengan `if(i % 3 == 0)`. Dengan perubahan ini , program akan memeriksa apakah nilai `i` adalah kelipatan dari 3. Jika ya, maka pernyataan `break` akan di eksekusi, menghentikan loop.
- b. Pada perubahan kedua, saya mengganti `break` dengan `continue`. Dengan menggunakan `continue`, program tidak akan menghentikan loop, tetapi akan melewati sisa iterasi saat kondisi `if (i % 3 == 0)` terpenuhi. Ini berarti bahwa Ketika `i` adalah 3, program akan mencetak angka tersebut dan langsung melanjutkan ke iterasi berikutnya tanpa mencetak angka 4. Pada perubahan pertama program mencetak angka 1,2, dan 3 , tetapi setelah perubahan kedua angkanya di lanjutkan ke iterasi berikutnya tanpa mencetak 4, dan kemudian mencetak 5 dan 6.

2) Alasan solusi ini karena penggunaan struktur control yang tepat dalam pemrograman sangat penting untuk mencapai hasil yang diinginkan. Dalam contoh program yang menggunakan loop while, kita melihat bahwa kondisi dievaluasi sebelum eksekusi blok kode. Ini berarti jika jumlah pengulangan yang dimasukkan adalah 0, maka tidak ada output yang dihasilkan, karena kondisi awal tidak terpenuhi.\_\_\_\_\_

3) Perbaiki kode program dengan cara :

- a. Menggunakan Satu Scanner : Daripada menggunakan dua objek Scanner untuk input, kita bisa menggunakan satu objek Scanner untuk kedua input. Ini akan membuat kode lebih efisien.
- b. Menangani Input Jumlah yang Negatif : Kita bisa menambahkan logika untuk menangani kasus dimana pengguna memasukkan jumlah pengulangan yang negative. Misalnya, jika jumlah yang dimasukkan kurang dari 0, kita bisa memberikan pesan kesalahan dan meminta input ulang.

- c. Menggunakan Do While untuk menjamin Eksekusi : Jika kita ingin memastikan bahwa kata selalu dicetak setidaknya sekali, kita bisa menggunakan struktur do While untuk mencetak kata, meskipun jumlah yang dimasukkan adalah 0.

## **[No.2 ] Penyusunan Algoritma ,Kode Program, dan diagram Flowchart**

### **1) Algoritma**

Algoritma adalah langkah-langkah penyelesaian masalah.

Algoritma program menghitung Rata Rata Nilai:

- (a) Mulai.
- (b) inisiasi idPelajaran
- (c) inisiasi nilai pelajaran
- (d) inisiasi nilai rata-rata
- (e) Minta pengguna untuk menuliskan jumlah pelajaran
- (f) Ketika idPelajaran lebih kecil dari jumlah pelajaran
- (g) Minta pengguna untuk menuliskan nilai pelajaran
- (h) Hitung nilai rata-rata =  $(\text{nilai pelajaran} + \text{nilai rata-rata}) / 2$
- (i) Tambah satu ke idPelajaran
- (j) Tampilkan nilai rata-rata
- (k) Selesai.

### **2) Kode program dan luaran**

```

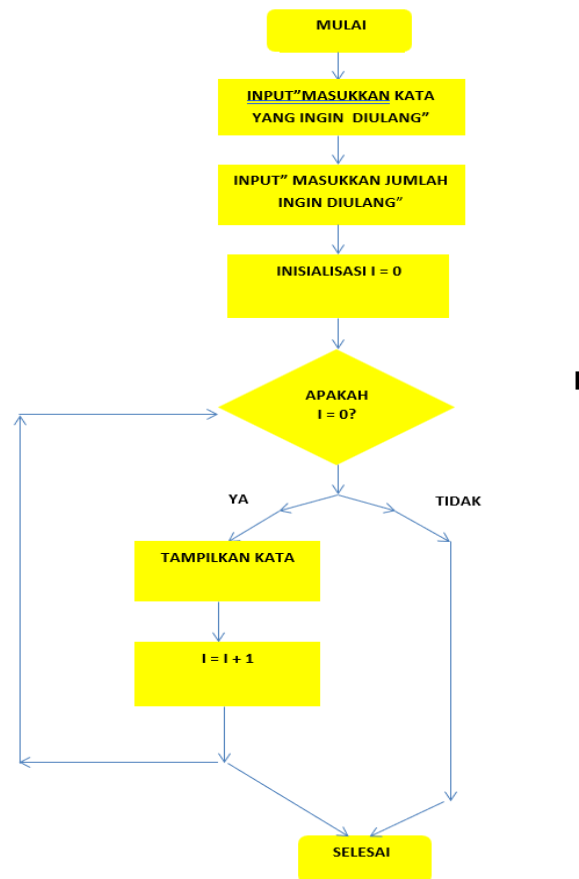
1  import java.util.Scanner;
2
3  public class RataRataNilai {
4      public static void main(String[] args) {
5          // Inisiasi variabel
6          int idPelajaran = 0;
7          double nilaiRataRata = 0;
8          Scanner scanner = new Scanner(System.in);
9
10         // Minta pengguna untuk menuliskan jumlah pelajaran
11         System.out.print("Masukkan jumlah pelajaran: ");
12         int jumlahPelajaran = scanner.nextInt();
13
14         // Proses untuk menghitung nilai rata-rata
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.println("Nilai rata-rata adalah: " + nilaiRataRata);
29
30         // Tutup scanner
31         scanner.close();
32     }
33 }
34

```

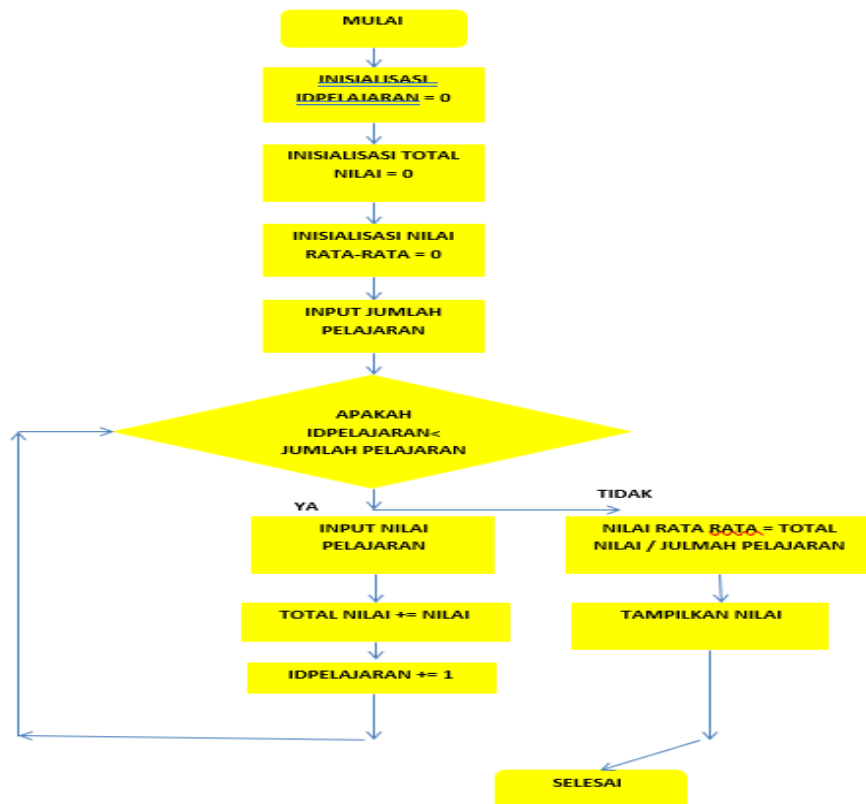
```
PS E:\TIPE DATA> e.; cd 'e:\TIPE DATA'; & 'C:\Program Files\Java\jdk-8.0.660\bin\java.exe' -jar '%cd%\src\main\resources\Code\User\workspaceStorage\fa5bb4322bcb69e63c79fb4c679be663\redhat\workspace\workspaceMetadata\workspaceMetadata.json'
Masukkan jumlah pelajaran: 5
Masukkan nilai pelajaran ke-1: 80
Masukkan nilai pelajaran ke-2: 88
Masukkan nilai pelajaran ke-3: 90
Masukkan nilai pelajaran ke-4: 95
Masukkan nilai pelajaran ke-5: 87
Nilai rata-rata adalah: 88.0
PS E:\TIPE DATA>
```

Luaran sudah sesuai dengan program yang disusun. Secara Keseluruhan program ini efektif dalam menghitung dan menampilkan nilai rata rata, tetapi perlu penanganan kesalahan untuk meningkatkan kualitasnya dan cukup dapat diandalkan untuk perhitungan.

a. Diagram Flowchart dai Latihan 2.1 dan 2.2



b. Diagram Flowchart dari Latihan 2.3



## **[No.1] Kesimpulan**

### **1) Analisa**

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawab :

Dari permasalahan diatas saya mengambil keputusan untuk menyimpulkan perbedaan antara while dan do while : Perbedaan utama antara perulangan while dan do while terletak pada cara mereka memeriksa kondisi untuk melanjutkan iterasi . Pada perulangan while, kondisi diperiksa sebelum eksekusi blok kode, sehingga jika kondisi tidak terpenuhi sejak awal, blok kode tidak akan dijalankan sama sekali. Sebaliknya, pada perulangan do while blok kode dieksekusi terlebih dahulu sebelum kondisi diperiksa, menjamin bahwa setidaknya satu iterasi akan selalu terjadi, bahkan jika kondisi awalnya tidak terpenuhi. Dengan demikian, do while lebih cocok digunakan Ketika kita ingin memastikan bahwa sesuatu aksi dilakukan setidaknya sekali, sementara while lebih tepat digunakan Ketika kita ingin Lanjutkan aksi hanya jika kondisi tertentu terpenuhi sejak awal.

Dasar alasan saya mengambil topik perbedaan antara while dan do while adalah sesuai dengan apa yang diminta oleh soal.

### **Refleksi**

Saya menyelesaikan tugas ini dengan hanya satu malam, karena tugas ini termasuk tugas yang dikategorikan tugas dengan tingkat kesulitan menengah. Saya memberikan usaha yang terbaik saat mengerjakan tugas ini, banyak hal baru yang saya pelajari saat mengerjakan tugas ini . Saya bisa meningkatkan kemampuan saya saat mengerjakan tugas ini dengan cara mengganti variabel pada contoh data atau ciri ciri yang di instruksikan. Pada materi yang mendatang saya harap saya bisa mendapatkan materi yang lebih sulit karena jika saya mendapatkan materi yang lebih sulit maka akan lebih banyak ilmu yang saya dapatkan dari tugas tersebut.