

**LAPORAN TUGAS ALGORITMA PEMROGRAMAN
KODE PROGRAM, FLOWCHART, PSEUDOCODE
WHILE/DO WHILE
PEKAN 6**



OLEH:

**LEXI MULIA YUNASPI
(2511531006)**

DOSEN PENGAMPU:

DR. WAHYUDI, S.T, M.T

**FAKULTAS TEKNOLOGI INFORMASI
DEPARTEMEN INFORMATIKA
UNIVERSITAS ANDALAS
2025**

KATA PENGANTAR

Segala puji dan syukur saya panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat, karunia, dan kesempatan yang diberikan kepada saya sehingga saya dapat menyelesaikan laporan tugas pekan 6 ini mengenai pembuatan kode program, flowchart, dan pseudocode ini dengan baik dan tepat waktu. Laporan ini disusun sebagai bagian proses pembelajaran untuk mengembangkan kemampuan dalam merancang dan mengimplementasikan suatu program secara sistematis dan terstruktur. Dalam laporan ini, saya memaparkan tahapan penting meliputi pembuatan flowchart sebagai gambaran visual menjelaskan logika program, penyusunan pseudocode sebagai langkah deskriptif merancang algoritma secara rinci, serta implementasi kode program yang merupakan tahap akhir proses perancangan tersebut. Dengan menggabungkan tiga elemen ini, diharapkan dapat memberikan pemahaman menyeluruh tentang bagaimana sebuah program dikembangkan mulai dari konsep hingga eksekusi.

Saya menyadari bahwa dalam proses penyusunan laporan ini masih jauh dari sempurna, oleh karena itu saya sangat mengharapkan kritik dan saran yang membangun guna penyempurnaan karya dan pengetahuan masa mendatang. Saya juga mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah berikan bimbingan, dorongan, dan dukungan, khususnya kepada dosen pengampu dan asisten praktikum yang memotivasi saya dalam menyelesaikan tugas ini. Semoga laporan ini dapat memberikan manfaat dan menjadi referensi yang berguna tidak hanya bagi saya sebagai penyusun, tetapi juga bagi pembaca yang ingin mempelajari lebih dalam mengenai proses pembuatan program secara terstruktur dan sistematis. Harapan saya, pengetahuan yang diperoleh melalui laporan ini dapat digunakan sebagai bekal dalam mengembangkan kemampuan di bidang pemrograman dan teknologi informasi.

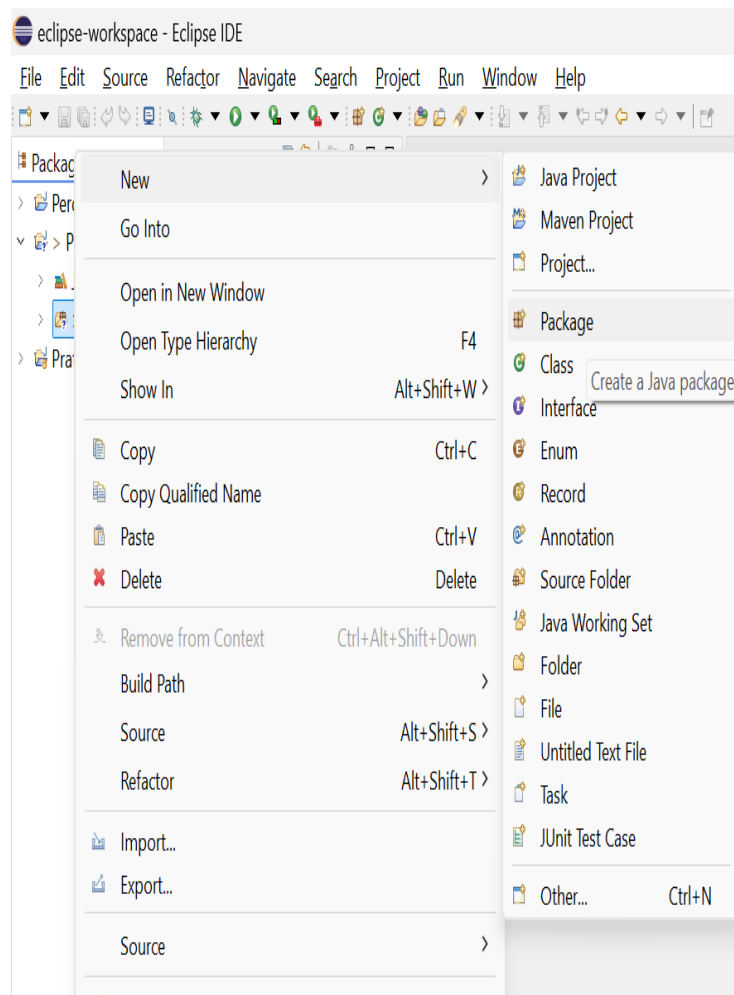
Padang, 05 November 2025

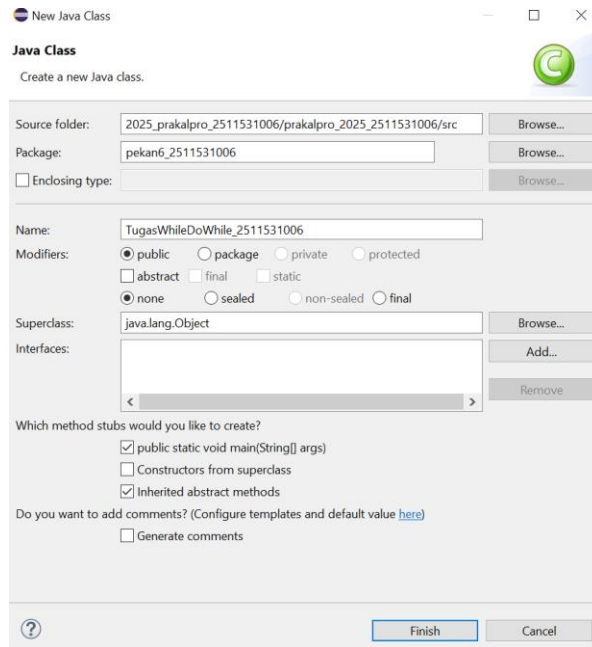
Lexi Mulia Yunaspi

Langkah Langkah Pembuatan Kode Program,Flowchart,Dan Pseudocode

1.Kode Program

A. pilih “New”, lalu pilih class. Buat nama dengan ketentuan nama harus Uppercase pada awal kalimat dan tanpa “space”, lalu centang tanda “public static void main (string[] args)”.





B. Jika sudah di tekan finish, maka Tampilan akan seperti berikut

```

1 package pekan6_2511531006;
2
3 import java.util.Random;
4
5
6 public class TugasWhileDoWhile_2511531006 {
7
8     public static void main(String[] args) {

```

C. Potongan kode pada gambar tersebut merupakan bagian dari program Java yang menyiapkan beberapa variabel untuk digunakan dalam logika permainan atau percobaan. Baris `Scanner input = new Scanner(System.in);` digunakan untuk membaca input dari pengguna melalui keyboard, sedangkan `Random acak = new Random();` berfungsi untuk menghasilkan angka acak. Variabel `String jawab1;` dideklarasikan untuk menyimpan jawaban dari pengguna,

dan `int percobaan = 0;` digunakan untuk menghitung jumlah percobaan yang dilakukan. Selanjutnya, `boolean menang = false;` berfungsi sebagai penanda apakah pemain sudah menang atau belum. Struktur `do { ... }` yang terlihat di bawahnya menandakan bahwa program akan menjalankan perulangan minimal satu kali sebelum memeriksa kondisi berhenti.

D. Potongan kode tersebut merupakan bagian dari program simulasi lempar dua buah dadu. Di dalam blok `do`, variabel `dadu1` dan `dadu2` diisi dengan bilangan acak antara 1 sampai 6 menggunakan `acak.nextInt(6) + 1`, yang merepresentasikan hasil lemparan masing-masing dadu. Nilai kedua dadu tersebut kemudian dijumlahkan dan disimpan pada variabel `total`. Setelah itu, variabel `percobaan` ditambah satu untuk menghitung berapa kali percobaan lempar dilakukan. Baris `System.out.println(dadu1 + " + " + dadu2 + " = " + total);` menampilkan hasil lemparan kedua dadu beserta jumlah totalnya ke layar.

```
16         int dadu1 = acak.nextInt(6) + 1;
17         int dadu2 = acak.nextInt(6) + 1;
18         int total = dadu1 + dadu2;
19         percobaan++;
20
21     System.out.println(dadu1 + " + " + dadu2 + " = " + total);
22
```

E. Potongan kode tersebut merupakan bagian dari logika pengecekan hasil lemparan dua dadu. Baris `if (total == 7)` digunakan untuk memeriksa apakah jumlah kedua dadu yang dilempar menghasilkan angka 7. Jika kondisi tersebut benar, maka program menampilkan pesan “Tebakan Anda Benar” dan “Anda menang setelah X percobaan!”, di mana X merupakan jumlah percobaan yang disimpan dalam variabel `percobaan`. Setelah itu, variabel `menang` diubah menjadi `true` untuk menandakan bahwa permainan sudah dimenangkan, dan perintah `break;` digunakan untuk menghentikan perulangan agar program tidak melanjutkan ke percobaan berikutnya.

```

22
23 if (total == 7) {
24     System.out.println("Tebakan Anda Benar");
25     System.out.println("Anda menang setelah " + percobaan + " percobaan!");
26     menang = true;
27     break;

```

G. Potongan kode tersebut merupakan bagian dari percabangan else yang dijalankan jika kondisi `if (total == 7)` sebelumnya tidak terpenuhi, artinya jumlah kedua dadu tidak sama dengan 7. Dalam kasus ini, program akan menampilkan pesan “Tebakan Anda Salah” untuk memberi tahu bahwa hasil lemparan belum sesuai dengan tebakan. Setelah itu, baris `System.out.print("Apakah mau lempar dadu (ya / tidak)? ");` digunakan untuk menanyakan kepada pengguna apakah mereka ingin melanjutkan permainan dengan melempar dadu lagi atau berhenti. Program kemudian menunggu input dari pengguna sebelum melanjutkan ke langkah berikutnya.

```

28 } else {
29     System.out.println("Tebakan Anda Salah");
30     System.out.print("Apakah mau lempar dadu (ya / tidak)? ");
31

```

H. Potongan kode tersebut berfungsi untuk membaca jawaban pengguna dan menentukan apakah permainan akan dilanjutkan atau dihentikan. Baris `String jawab = input.nextLine();` digunakan untuk menyimpan input dari pengguna ke dalam variabel `jawab`. Kemudian, kondisi `if (!jawab.equalsIgnoreCase("ya"))` memeriksa apakah pengguna tidak menjawab “ya” (dengan mengabaikan huruf besar atau kecil). Jika pengguna mengetik selain “ya”, maka program menampilkan pesan “Anda gagal menang” dan perintah `break;` menghentikan perulangan, yang berarti permainan berakhir karena pengguna memilih untuk tidak melanjutkan lemparan dadu.

```

31
32     String jawab = input.nextLine();
33 if (!jawab.equalsIgnoreCase("ya")) {
34     System.out.println("Anda gagal menang");
35     break;

```

I. Potongan kode `} while (true);` pada gambar tersebut menunjukkan struktur perulangan `do-while` tanpa kondisi berhenti (infinite loop). Artinya, blok kode di dalam `do { ... } while (true);` akan terus dijalankan berulang-ulang tanpa henti karena kondisi `true` selalu bernilai benar. Biasanya, perulangan ini digunakan agar program terus berjalan hingga ada perintah `break` di dalam blok tersebut untuk menghentikan loop, misalnya saat pengguna memilih keluar dari menu atau ketika kondisi tertentu terpenuhi.

```
36         }
37     }
38     } while (true);
39
40
41     }
42 }
```

J.kode programnya

```
1  package pekan6_2511531006;
2
3+ import java.util.Random;
4
5
6  public class TugasWhileDoWhile_2511531006 {
7
8      public static void main(String[] args) {
9          Scanner input = new Scanner(System.in);
10         Random acak = new Random();
11         String jawab1;
12         int percobaan = 0;
13         boolean menang = false;
14
15         do {
16             int dadu1 = acak.nextInt(6) + 1;
17             int dadu2 = acak.nextInt(6) + 1;
18             int total = dadu1 + dadu2;
19             percobaan++;
20
21             System.out.println(dadu1 + " + " + dadu2 + " = " + total);
22
23             if (total == 7) {
24                 System.out.println("Tebakan Anda Benar");
25                 System.out.println("Anda menang setelah " + percobaan + " percobaan!");
26                 menang = true;
27                 break;
28             } else {
29                 System.out.println("Tebakan Anda Salah");
30                 System.out.print("Apakah mau lempar dadu (ya / tidak)? ");
31
32                 String jawab = input.nextLine();
33                 if (!jawab.equalsIgnoreCase("ya")) {
34                     System.out.println("Anda gagal menang");
35                     break;
36                 }
37             }
38         }
```

```

38         } while (true);
39
40
41     }
42 }

```

K.Hasil Output

a.Hasil run pertama

```

Console X
<terminated> TugasWhileDoWhile_2511531006 [Java Applica
6 + 3 = 9
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? ya
3 + 4 = 7
Tebakan Anda Benar
Anda menang setelah 2 percobaan!

```

b.Hasil run kedua

```

Console X
<terminated> TugasWhileDoWhile_2511531006 [Java Applicatio
3 + 5 = 8
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? ya
6 + 4 = 10
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? tidak
Anda gagal menang

```

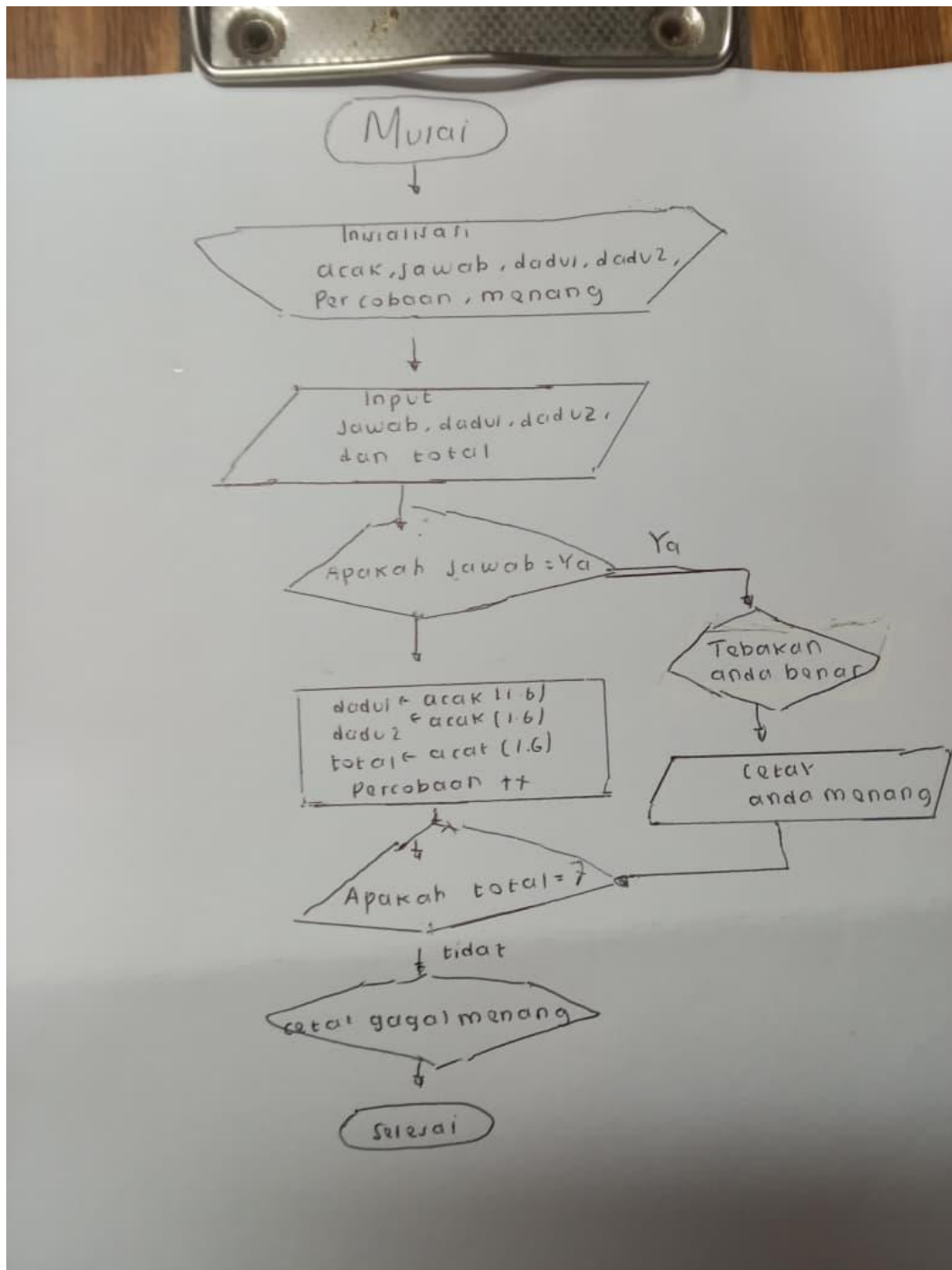
c.Hasil run ketiga

```

Console X
<terminated> TugasWhileDoWhile_2511531006 [Java App
4 + 1 = 5
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? ya
3 + 1 = 4
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? ya
6 + 6 = 12
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? ya
1 + 5 = 6
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak)? ya
1 + 6 = 7
Tebakan Anda Benar
Anda menang setelah 5 percobaan!

```


2. Flowchart



3. Pseudocode

A. Judul

Lempar Dadu

B. Deklarasi

Input : Scanner

Acak : Random

Jawab : String

dadu1,dadu 2,percobaan : Integer

menang : Boolean

C.Algoritma

- 1.Inisiaalisasi acak,jawab,dadu1,dadu2,dan percobaaan,menang
- 2.Input jawab,dadu1,dadu2,total
- 3.Baca(jawab)
- 4.Jika jawab \neq "ya" maka
 - a.Cetak("Program dibatalkan oleh user.")
 - b. Akhiri program
- 5.menang \leftarrow false
- 6.percobaan \leftarrow 0
7. Selama menang = false lakukan
 9. dadu1 \leftarrow acak(1..6)
 - 10.dadu2 \leftarrow acak(1..6)
 11. total \leftarrow dadu1 + dadu2
 12. percobaan \leftarrow percobaan + 1
- 13.Cetak("Lemparan ke-", percobaan, ": ", dadu1, " + ", dadu2, " = ", total)
14. Jika total = 7 maka
 - Cetak("Tebakan Anda Benar!")
 - Cetak("Anda menang setelah ", percobaan, " percobaan.")
 - menang \leftarrow true
 - a.Jika tidak
 - Cetak("Tebakan Anda Salah.")
 - Cetak("Apakah mau lempar dadu lagi (ya/tidak)?")

bJika jawab \neq "ya" maka

 Cetak("Anda gagal menang.")

 Akhir program

15.Print