

**LAPORAN TUGAS PEKAN 6**  
**PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN**  
**“LEMPAR DADU”**



Dosen Pengampu:

DR. Wahyudi, S.T., M.T.

Asisten Lab:

Muhammad Zaki Al Hafiz

Disusun Oleh:

Hamdi Sidqi Alifi

2511531009

Fakultas Teknologi Informasi

Departemen Informatika

Universitas Andalas

2025

## **KATA PENGANTAR**

Puji dan syukur senantiasa penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan laporan ini guna memenuhi penugasan pasca-pratikum mata kuliah Algoritma Pemrograman pekan ke-5 yang dilaksanakan pada hari Rabu tanggal 5 November 2025.

Pada kesempatan ini, Penulis ingin mengucapkan terima kasih kepada Bapak DR. Wahyudi. .S.T.M.T, dosen pengampu mata kuliah Algoritma Pemograman, yang telah memberikan tugas ini. Penulis juga ingin mengucapkan terima kasih kepada pihak-pihak yang telah mendukung serta membantu penulis dalam penyelesaian laporan tugas Algoritma Pemrograman ini.

Penulis sadar bahwa laporan ini masih memiliki beberapa kekurangan dikarenakan terbatasnya pengalaman dan pengetahuan yang penulis miliki. Oleh karena itu, penulis harap adanya bentuk saran dan kritik yang membangun dari berbagai pihak.

Akhir kata, penulis berharap laporan ini dapat bermanfaat bagi perkembangan dunia pendidikan.

## **DAFTAR ISI**

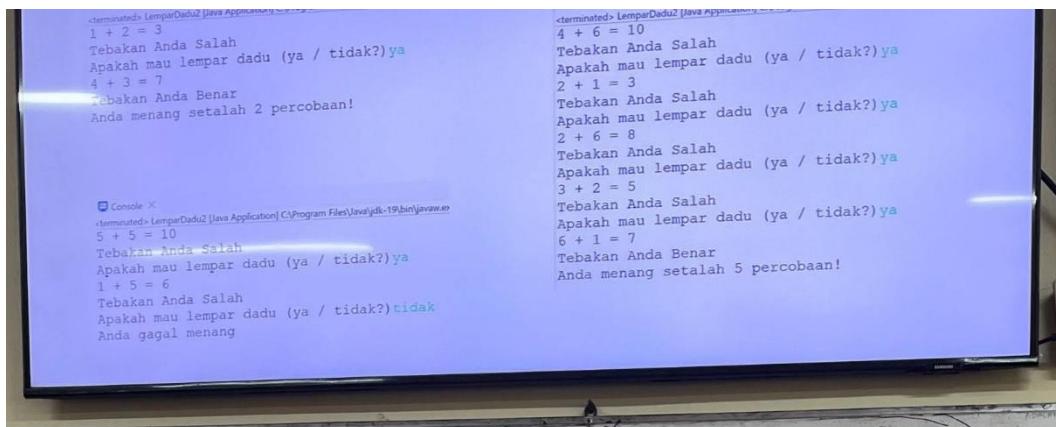
KATA PENGANTAR .....	i
DAFTAR ISI.....	ii
BAB I EKSPLORASI.....	1
1.1 Latar Belakang .....	1
1.2 Tujuan.....	1
1.3 Manfaat Pratikum .....	1
BAB II RANCANGAN.....	2
2.1 Analisis .....	2
BAB III PROGRAM.....	3
3.1 Bahasa Natural .....	3
3.2 Flowchart.....	3
3.3 Pseudo-code.....	4
3.4 Java language .....	6

# BAB I

## EKSPLORASI

### 1.1 Latar Belakang

Pada pekan ke-5 kita telah mempelajari tentang perulangan *while*. Untuk lebih memahami penggunaan fungsi *while-loop*, maka diberikan sebuah hasil output, yang mana mahasiswa Praktikum Algoritma dan Pemograman (Alpro) diharapkan dapat mereplikasi program belah ketupat yang telah dipaparkan pada monitor.



Gambar I.1

### 1.2 Tujuan

Mereplika program lempar dadu di dalam *Java* menggunakan fungsi *while-loop*.

### 1.3 Manfaat Praktikum

Memahami konsep perulangan *while-loop* dalam bahasa pemrograman *Java* secara lebih kompleks.

## **BAB II**

## **RANCANGAN**

### **2.1 Analisis**

Berdasarkan gambar, terdapat dua angka acak yang merupakan hasil simulasi output ruang sampel dari dadu bersisi enam dengan peluang yang sama jika dilempar. Kedua angka tersebut dijumlahkan, apabila hasilnya tujuh maka pemain memenangkan permainan, jika tidak pemain diberikan pilihan untuk lanjut atau tidak.

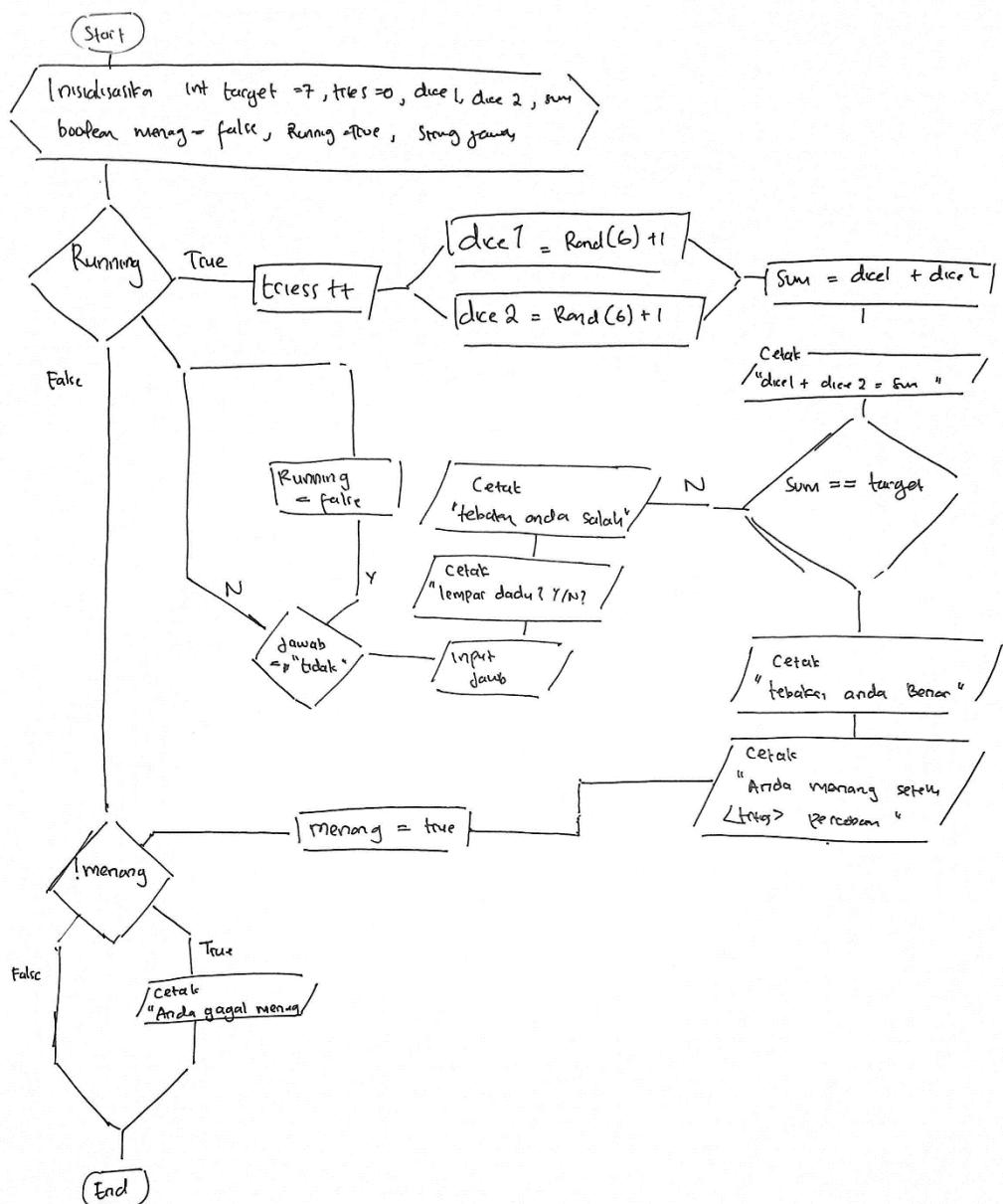
## **BAB III**

### **PROGRAM**

#### **3.1 Bahasa Natural**

- 1) Inisialisasikan target = 7, tries = 0, menang = *false*, *Running* = *true*, *dice1*, *dice2*, *sum*, jawab.
- 2) Cek nilai *Running*, apakah nilainya *true* atau *false*?
  - a. Jika true. maka *value tries* bertambah 1, kemudian :
    - i. Acak angka dengan rentang 0 – 5, tambahkan dengan 1, masukkan ke dalam variabel *dice1*
    - ii. Acak angka dengan rentang 0 – 5, tambahkan dengan 1, masukkan ke dalam variabel *dice2*
    - iii. Jumlahkan nilai *dice1* dengan *dice2* dan masukkan nilai tersebut ke dalam variabel *sum*
    - iv. Cetak line untuk menunjukkan hasil penjumlahan kedua dadu
    - v. Bandingkan sum dengan target, apakah nilainya sama
      - Jika ya, maka cetak “tebakan anda benar” dan dalam berapa kali percobaan. Terapkan nilai true pada variabel menang, kemudian hentikan perulangan.
      - Jika tidak, maka “cetak “tebakan anda salah” dan tawarkan untuk mengulangi permainan. Apakah jawabannya tidak?
        - Jika jawabannya tidak, maka terapkan nilai false untuk variabel *Running*. Ulangi nomor (2)
        - Jika jawabannya bukan tidak, ulangi nomor (2)
  - b. Jika false, maka periksa apakah nilai menang berupa *false*, jika ya maka cetak “anda gagal menang”

#### **3.2 Flowchart**



### 3.3 Pseudo-code

#### Judul

Program Lempar Dadu

{ menciptakan program simulasi penjumlahan dua dadu dengan target hasil pejumlahan angka tujuh }

#### Deklarasi

int target = 7, tries = 0, dice1, dice2, sum,

Boolean menang = false, Running = true,  
String jawab

### Algoritma dan Logika

1. While (running)
2.     Tries++
3.     Dice1 = Rand(6)+1
4.     Dice2 = Rand(6)+1
5.     Sum = dice1 + dice2
6.     Print("d1+d2=sum")
7.     If (sum==target)  
        Println("tebakan benar")
8.         Println("anda menang  
setelah"+tries+"percobaan")
9.     Menang = true
10.    Break
11. Else  
12.     Println("tebakan anda salah")  
13.     Print("apakah mau lempar dadu?")  
14.     Input jawab  
15.     If (jawab =ignorecase "tidak")  
        Running = false
16. If (!menang)  
    Println("anda gagal menang")

### 3.4 Java language

```
1 package pekan6_2511531009;
2 import java.util.*;
3 public class TugasPekan6_LemparDadu_2511531009 {
4
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7         Random rand = new Random();
8
9         int target = 7;
10        int tries = 0;
11        boolean menang = false;
12        boolean Running = true;
13
14    while (Running) {
15        tries++;
16
17        int dice1 = rand.nextInt(6) + 1;
18        int dice2 = rand.nextInt(6) + 1;
19        int sum = dice1 + dice2;
20
21        System.out.println(dice1 + " + " + dice2 + " = " + sum);
22
23        if (sum == target) {
24            System.out.println("Tebakan Anda Benar");
25            System.out.println("Anda menang setelah " + tries + " 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            String jawab = input.nextLine();
32
33            if (jawab.equalsIgnoreCase("tidak")) {
34                Running = false;
35            }
36        }
37    }
38
39    if (!menang) {
40        System.out.println("Anda gagal menang");
41    }
42
43    input.close();
44}
45
```

Disini kita menginisialisasikan bilangan yang kita inginkan (target), pada kasus ini kita mencari angka tujuh, sehingga nilai target kita masukkan dengan tujuh. Kemudian *tries* sebagai penunjuk sudah berap kali kita melakukan percobaan, diinisialisasikan dengan bilangan nol karena variabel *tries* akan diitterasikan nantinya. Kita juga menyatakan variabel *menang* untuk indikasi kemenangnana. Kemudian variabel *Running* untuk indikator jalannya program.

Karena bilangan dadu hanya dimulai dari satu, maka hasil generatif kedua bilangan random masing-masing kita tambah dengan satu. Ketika kita telah menjumlahkan kedua dadu (yang kita masukkan ke dalam variable *sum*). Jika

nilai *sum* sesuai dengan target maka pemain menang dan perulangan berhenti yang dinyatakan oleh *syntax break* pada *line* (27). Jika tidak, maka pemain ditawarkan untuk bermain kembali atau tidak, jawaban yang hanya mempengaruhi program hanya “tidak”, tak peduli apakah besar atau kecil karena kita menggunakan *syntax equalsIgnoreCase* seperti yang dapat dilihat pada *line* (33).

Output :

```
4 + 3 = 7
Tebakan Anda Benar
Anda menang setelah 1 percobaan!
```

```
2 + 3 = 5
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak?) ya
6 + 1 = 7
Tebakan Anda Benar
Anda menang setelah 2 percobaan!
```

```
4 + 4 = 8
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak?) ya
3 + 5 = 8
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak?) ya
1 + 2 = 3
Tebakan Anda Salah
Apakah mau lempar dadu (ya / tidak?) tidak
Anda gagal menang
```