

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
PERULANGAN FOR

Disusun Oleh:

Nama: Rifki Maulana

NIM: 2511533007

Dosen Pengampu: Dr. Wahyudi, S.T.,M.T.

Asisten Praktikum: Jovantri Immanuel Gulo



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN 2025

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa karena dengan rahmat-Nya laporan praktikum/makalah ini dapat diselesaikan. Laporan ini disusun sebagai bagian dari pelaksanaan mata kuliah Praktikum Algoritma dan Pemrograman untuk memenuhi tugas dan memperdalam pemahaman praktis terhadap materi yang dibahas.

Terima kasih disampaikan kepada dosen pengampu, asisten praktikum, serta rekan-rekan yang telah memberikan bimbingan, bantuan, dan kerja sama selama kegiatan praktikum berlangsung. Penulis menyadari laporan ini masih jauh dari sempurna, oleh karena itu kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa mendatang.

Semoga laporan ini bermanfaat bagi penulis dan pembaca.

Padang, 01 November 2025

Rifki Maulana

DAFTAR ISI

KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan Praktikum.....	1
1.3 Manfaat Praktikum.....	1
BAB 2 PEMBAHASAN.....	2
2.1 Uraian Kode Program.....	2
2.2 Langkah Kerja.....	4
2.3 Analisis hasil.....	6
BAB 3 KESIMPULAN.....	9
DAFTAR PUSTAKA.....	10

BAB I PENDAHULUAN

1.1 Latar Belakang

Pemrograman merupakan keterampilan dasar yang penting dalam bidang informatika. Salah satu konsep yang harus dipahami oleh mahasiswa adalah perulangan. Perulangan digunakan untuk mengeksekusi blok kode secara berulang dalam kondisi tertentu. Dengan perulangan program menjadi lebih efisien untuk tugas yang harus dilakukan berkali kali.

1.2 Tujuan Praktikum

Praktikum ini memiliki tujuan sebagai berikut:

1. Mahasiswa dapat memahami konsep dasar perulangan for dan nested for pada java.
2. Mahasiswa mampu mengimplementasikan perulangan untuk menyelesaikan masalah

1.3 Manfaat Praktikum

Mahasiswa dapat meningkatkan dan memperkuat pemahaman konsep perulangan, mengoptimalkan kode dengan menggunakan perulangan, dan melatih untuk membuat logika yang kompleks pada perulangan.

BAB 2 PEMBAHASAN

2.1 Uraian Kode Program

```
1 package Pekan5;
2
3 public class PerulanganFor1 {
4     public static void main(String[] args) {
5         for (int i = 1; i <= 10; i++) {
6             System.out.println(i);
7         }
8     }
9 }
10
11
```

Kode Program 2.1 PerulanganFor1

```
1 package Pekan5;
2
3 public class Perulanganfor2 {
4     public static void main(String[] args) {
5         for (int i = 1; i <= 10; i++) {
6             System.out.print(i + " ");
7         }
8     }
9 }
10
11
```

Kode Program 2.2 PerulanganFor2

```
1 package Pekan5;
2
3 public class perulanganFor3 {
4
5     public static void main(String[] args) {
6         int jumlah=0;
7         for (int i=1;i<=10;i++) {
8             System.out.print(i);
9             jumlah= jumlah+i;
10            if (i<10) {
11                System.out.print(" + ");
12            }
13        }
14        System.out.println();
15        System.out.println("Jumlah = " + jumlah);
16    }
17 }
18
19
```

Kode Program 2.3 PerulanganFor3

```

1 package Pekan5;
2 import java.util.Scanner;
3 public class perulanganFor4 {
4
5     public static void main(String[] args) {
6         int jumlah=0;
7         int batas;
8         Scanner input = new Scanner(System.in);
9         System.out.print("Masukkan nilai batas = ");
10        batas = input.nextInt();
11        input.close();
12        for (int i = 1; i<=batas; i++) {
13            System.out.print(i);
14            jumlah = jumlah+i;
15            if (i<batas) {
16                System.out.print(" + ");
17            }
18            else {
19                System.out.print(" = ");
20            }
21        }
22        System.out.println(jumlah);
23    }
24 }

```

Kode Program 2.4 PerulanganFor4

```

1 package Pekan5;
2
3 public class nestedFor0 {
4
5     public static void main(String[] args) {
6         for (int line = 1; line <= 5; line++) {
7             for (int j = 1; j <= (-1 * line + 5); j++) {
8                 System.out.print(".");
9             }
10            System.out.print(line);
11            System.out.println();
12        }
13    }
14 }

```

Kode Program 2.5 nestedFor0

```

1 package Pekan5;
2
3 public class nestedFor1 {
4
5     public static void main(String[] args) {
6         for (int i = 1; i <= 5; i++) {
7             for (int j = 1; j <= 5; j++) {
8                 System.out.print("+");
9             }
10            System.out.println();
11            // to end the line
12        }
13    }
14 }

```

Kode Program 2.6 nestedFor1

```
1 package Pekan5;
2
3 public class nestedFor2 {
4
5     public static void main(String[] args) {
6         for (int i = 0; i <= 5; i++) {
7             for (int j = 0; j <= 5; j++) {
8                 System.out.print(i+j+ " ");
9             }
10            System.out.println();
11            // to end the line
12        }
13    }
14 }
```

Kode Program 2.7 nestedFor2

2.2 Langkah Kerja

Langkah-langkah menyusun program:

1. Program PerulanganFor1

- a) Mulai
- b) Inisialisasi int i = 1
- c) Buat perulangan for (int i = 1; i <=10; i++)
- d) Print i menggunakan system.out.println(i);
- e) Selesai

2. Program PerulanganFor2

- a) Mulai
- b) Inisialisasi dan kondisi perulangan sama dengan program PerulanganFor1
for (int i = 1; i <= 10; i++)
- c) Cetak i dalam *loop* menggunakan system.out print(i + " ");
- d) Selesai

3. Program PerulanganFor3

- a) Mulai
- b) Deklarasi int jumlah = 0

- c) *Loop* for (int i = 1; i <= 10; i++)
- d) *Print* i
- e) Proses jumlah = jumlah + i
- f) Buat *if* jika i<10 *print* " + "
- g) Kemudian diluar *loop* buat system.out.println(); supaya *outputnya* dibaris yang baru
- h) Buat system.out.println("jumlah = " + jumlah); untuk menampilkan output
- i) Selesai

4. Program PerulanganFor4

- a) Mulai
- b) Import java.util.Scanner
- c) Deklarasi int jumlah = 0 dan int batas
- d) Baca nilai batas menggunakan scanner dan jangan lupa close scanner
- e) *Loop* for (int i = 1; i <= batas; i++)
- f) *Print* i dan proses jumlah = jumlah + i
- g) Buat *if* i<batas *print* " + " else *print* " = "
- h) Kemudian buat system.out.println(jumlah)
- i) Selesai

5. Program nestedFor0

- a) Mulai
- b) Buat *loop* for (int line = 1; line <= 5; line++)
- c) Kemudian buat lagi *loop* didalamnya dengan *loop* bagian dalam yaitu for (int j = 1; j<= (-1*line + 5); j++)
- d) Pada *loop* bagian dalam buatlah *print* "."
- e) Kemudian pada *loop* bagian luar buatlah system.out.print(line); kemudian sytem.out.println()
- f) selesai

6. Program nestedFor1

- a) Mulai
- b) Buat *loop* for (int i = 1; i <= 5; i++)
- c) Kemudian buat lagi *loop* didalamnya dengan *loop* bagian dalam yaitu for (int j = 1; j <= 5; j++)
- d) Pada *loop* bagian dalam buatlah *print* "+"
- e) Kemudian pada *loop* bagian luar buatlah system.out.println();
- f) selesai

7. Program nestedFor2

- a) Mulai
- b) Buat *loop* for (int i = 0; i <= 5; i++)
- c) Kemudian buat lagi *loop* didalamnya dengan *loop* bagian dalam yaitu for (int j = 0; j <= 5; j++)
- d) Pada *loop* bagian dalam buatlah *print* (i+j+ "+")
- e) Kemudian pada *loop* bagian luar buatlah system.out.println();
- f) selesai

2.3 Analisis Hasil

1. Analisis kode program 2.1 PerulanganFor1

Program mencetak bilangan 1 sampai 10 dengan system.out.println(i); sehingga setiap angka pada *output* pada baris terpisah. Jika nilai batas diubah, *output* menyesuaikan sesuai kondisi $i \leq \text{batas}$.

2. Analisis kode program 2.2 PerulanganFor2

Sama fungsi dasar dengan kode program 2.1 tetapi menggunakan System.out.print(i + " "); sehingga semua angka *diprint* pada satu baris dipisah spasi. Tanpa println() akhir, *output* tetap di *line* yang sama.

3. Analisis kode program 2.3 PerulanganFor3

Program menampilkan deret $1+2+\dots+10$ dan menghitung total menggunakan accumulator jumlah. *Outputnya* akan menjadi $1 + 2 + 3 + \dots + 10$

Jumlah = 55

Logikanya yaitu setiap iterasi menambah i ke jumlah dan mencetak pemisah " + " jika bukan elemen terakhir. Jika hanya perlu nilai jumlah deret $1..n$, bisa menggunakan rumus aritmetika $n*(n+1)/2$.

4. Analisis kode program 2.4 PerulanganFor4

Program menerima *input user* batas, mencetak deret $1 + 2 + \dots + \text{batas} = \text{jumlah}$. Kita perlu memerhatikan bahwa validasi input (misal $\text{batas} < 1$) agar tidak membingungkan ketika *user* memasukkan nol/negatif. Dan jangan lupa untuk menutup scanner.

5. Analisis kode program 2.5 nestedFor0

Pada kode program ini kita menggunakan *nested for* yaitu *for* didalam *for* atau *loop* didalam *loop*. Program ini menghasilkan pola di mana jumlah titik (".") berkurang tiap baris lalu diikuti angka baris. Untuk line 1 sampai 5 *outputnya*:

```
....1
...2
..3
.4
5
```

Dengan menggunakan logika yang mana jumlah titik memakai $(-1 * \text{line} + 5)$.

6. Analisis kode program 2.6 nestedFor1

Pada program ini kita ingin mencetak persegi 5×5 dari tanda (+). Setiap baris terdiri dari 5 tanda +. Contoh *outputnya*:

```
+++++
+++++
+++++
+++++
+++++
```

Pola ini memakai nested for (baris \times kolom).

7. Analisi kode program 2.7 nestedFor2

Pada kode program ini kita mengharapkan *output* yaitu tabel hasil penjumlahan indeks baris dan kolom untuk variabel *i* dan *j* dari 0 sampai 5, menghasilkan matriks 6 \times 6:

```
0 1 2 3 4 5
1 2 3 4 5 6
2 3 4 5 6 7
3 4 5 6 7 8
4 5 6 7 8 9
5 6 7 8 9 10
```

Berguna untuk memahami nested loop dan hubungan indeks.

BAB III KESIMPULAN

Laporan praktikum ini membahas implementasi dan analisis perulangan *for* dan *nested for* di Java melalui tujuh program contoh. Dari percobaan dan pengamatan terhadap *output* program dapat disimpulkan bahwa perulangan *for* adalah struktur kontrol yang efektif untuk mengulangi eksekusi kode ketika jumlah iterasi diketahui, sedangkan *nested for* diperlukan untuk mengolah atau mencetak struktur dua dimensi.

Secara keseluruhan, penguasaan perulangan *for* dan *nested for* merupakan hal yang penting dalam logika pemrograman karena banyak masalah algoritma dan pemrosesan data dapat diselesaikan menggunakan perulangan ini.

DAFTAR PUSTAKA

- [1] P. J. Deitel and H. M. Deitel, *Java™ How to Program*, 10th ed. Pearson Education, 2017.
- [2] J. Bloch, *Effective Java*, 3rd ed. Addison-Wesley, 2018.
- [3] Oracle, “The Java™ Tutorials,” 2025. [Online]. Available: <https://docs.oracle.com/javase/tutorial/>. [Accessed: 02-Nov-2025].