

Siakad Kita

START >

KELOMPOK 2

DINA MEI LESTARI (09)
ADINDA
ZAQI

Sistem Informasi Akademik (SIAKAD)

Tujuan Pembuatan

1. Mempermudah pengelolaan data mahasiswa secara sistematis.
2. Meningkatkan efisiensi dalam pencatatan dan pengolahan nilai mahasiswa.
3. Menyediakan laporan IPK dan rata-rata kelas secara otomatis dan akurat.



Mengapa Membuat Sistem Ini?

1. Masalah yang Dihadapi:

- Kesalahan dalam penghitungan nilai mahasiswa secara manual.
- Data mahasiswa sulit dilacak atau tidak terorganisir dengan baik.
- Membutuhkan waktu lebih lama untuk menganalisis data akademik.

2. Solusi yang Ditawarkan:

- Aplikasi berbasis program dapat mencatat data mahasiswa dan nilai dengan struktur yang terorganisir.
- Menghitung nilai rata-rata, IPK, dan kualifikasi secara otomatis berdasarkan input data.
- Memberikan laporan yang mudah dipahami untuk mahasiswa dan dosen.

Flowchart



Coding



Hasil Codingan



Penjelasan Flowchart dan Coding

Flowchart:

1. Input jumlah mahasiswa, nama, NIM, dan nilai (UTS, UAS, Tugas).
2. Validasi input nilai (0-100).
3. Hitung rata-rata nilai dan IPK berdasarkan kriteria.
4. Tampilkan data mahasiswa, nilai, rata-rata, dan IPK.
5. Hitung rata-rata IPK kelas.

Coding:

- Menggunakan array 2D untuk menyimpan data mahasiswa dan nilai.
- Perhitungan rata-rata dan IPK menggunakan struktur pengulangan.
- Validasi input untuk memastikan nilai dalam rentang yang sesuai.

UTS



UAS



TUGAS



IPK



FLOWCHART



For Siakad, Scan me!



For Kafe, Scan me!

CODING



for Siakad, Scan Me!



for Kafe, Scan me!

```
import java.util.Scanner;
```

```
public class SiakadKita {
```

```
    Run | Debug
```

```
    public static void main(String[] args) {
```

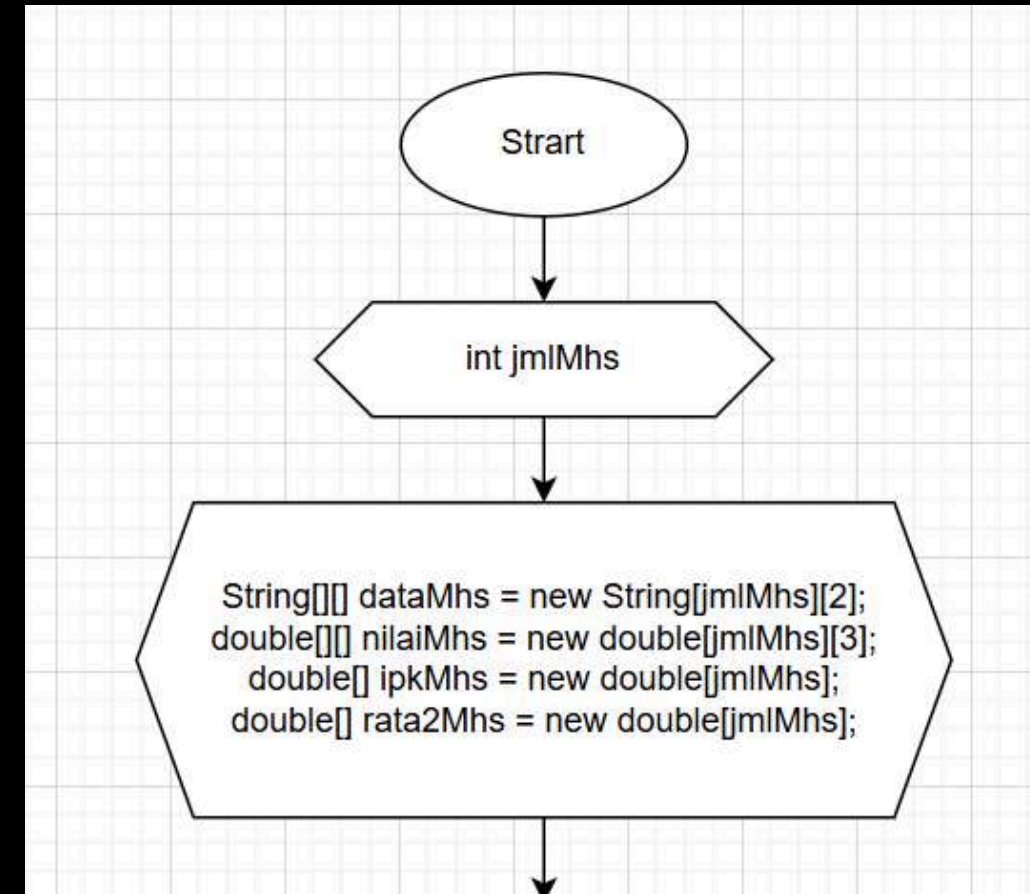
```
        Scanner sc = new Scanner(System.in);
```

```
        System.out.print(s:"Masukkan jumlah mahasiswa: ");  
        int jmlMhs = sc.nextInt();
```

```
        String[][] dataMhs = new String[jmlMhs][2];  
        double[][] nilaiMhs = new double[jmlMhs][3];  
        double[] ipkMhs = new double[jmlMhs];  
        double[] rata2Mhs = new double[jmlMhs];
```

```
        sc.nextLine();  
        String[] tipeNilai = {"UTS", "UAS", "Tugas"};
```

Our Process



Masukkan jumlah mahasiswa: 2

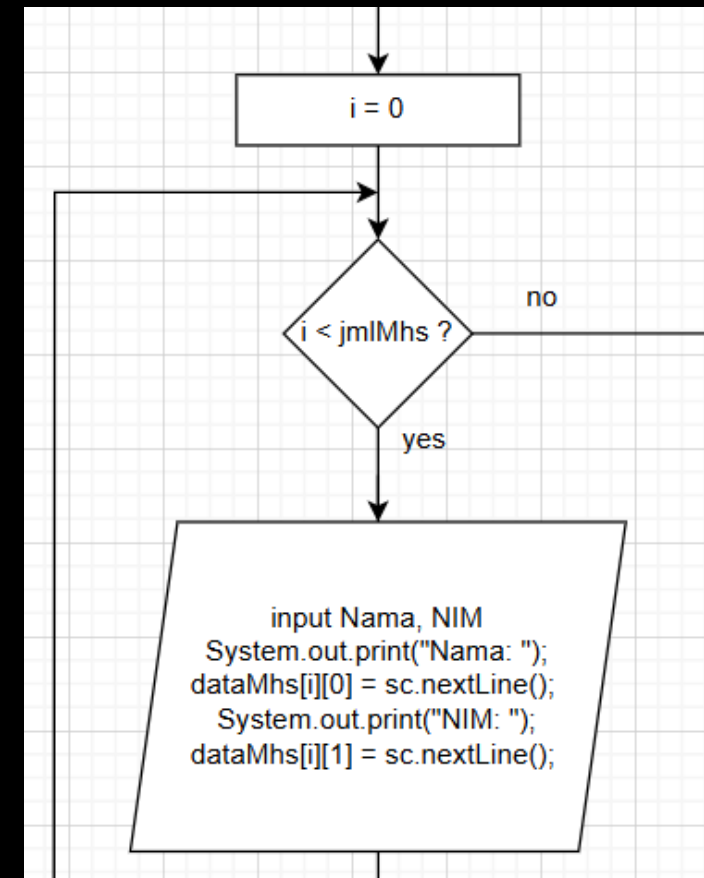
Kode ini berfungsi untuk mempersiapkan input data mahasiswa dan nilai mereka.

- dataMhs: Menyimpan Nama dan NIM.
- nilaiMhs: Menyimpan nilai UTS, UAS, dan Tugas.
- rata2Mhs & ipkMhs: Menghitung rata-rata nilai dan IPK setiap mahasiswa.
- tipeNilai: Label untuk kategori nilai.

Kode ini membangun struktur data yang dibutuhkan untuk pengolahan lebih lanjut.

Our Process

```
for (int i = 0; i < jmlMhs; i++) {  
    System.out.println("\nInput data untuk Mahasiswa ke-" + (i + 1));  
    System.out.print(s:"Nama: ");  
    dataMhs[i][0] = sc.nextLine();  
    System.out.print(s:"NIM: ");  
    dataMhs[i][1] = sc.nextLine();  
}
```



Input data untuk Mahasiswa ke-1
Nama: Dina Mei Lestari
NIM: 244107060105

Kode ini digunakan untuk menginput data setiap mahasiswa:

- Perulangan for: Mengulang proses input berdasarkan jumlah mahasiswa (jmlMhs).
- dataMhs[i][0]: Menyimpan nama mahasiswa.
- dataMhs[i][1]: Menyimpan NIM mahasiswa.

Setiap iterasi akan meminta pengguna memasukkan data untuk mahasiswa ke-(i+1).

Our Process

```
double totalNilai = 0;
```

```
for (int j = 0; j < 3; j++) {  
    boolean validInput = false;  
    while (!validInput) {  
        System.out.print("Masukkan nilai " + tipeNilai[j] + " (0-100): ");  
        nilaiMhs[i][j] = sc.nextDouble();  
        if (nilaiMhs[i][j] >= 0 && nilaiMhs[i][j] <= 100) {  
            totalNilai += nilaiMhs[i][j];  
            validInput = true;  
        } else {  
            System.out.println(x:"Nilai tidak valid, masukkan nilai antara 0 dan 100.");  
        }  
    }  
}
```

- totalNilai: Menyimpan total nilai mahasiswa.
- Perulangan memasukkan nilai UTS, UAS, dan Tugas.
- Validasi memastikan nilai antara 0-100.
- Nilai valid ditambahkan ke totalNilai.

Input data untuk Mahasiswa ke-1

Nama: Dina Mei Lestari

NIM: 244107060105

Masukkan nilai UTS (0-100): 100

Masukkan nilai UAS (0-100): 90

Masukkan nilai Tugas (0-100): 90

Kualifikasi: Sangat Baik (A)

Input data untuk Mahasiswa ke-2

Nama: Feronika Wulandari

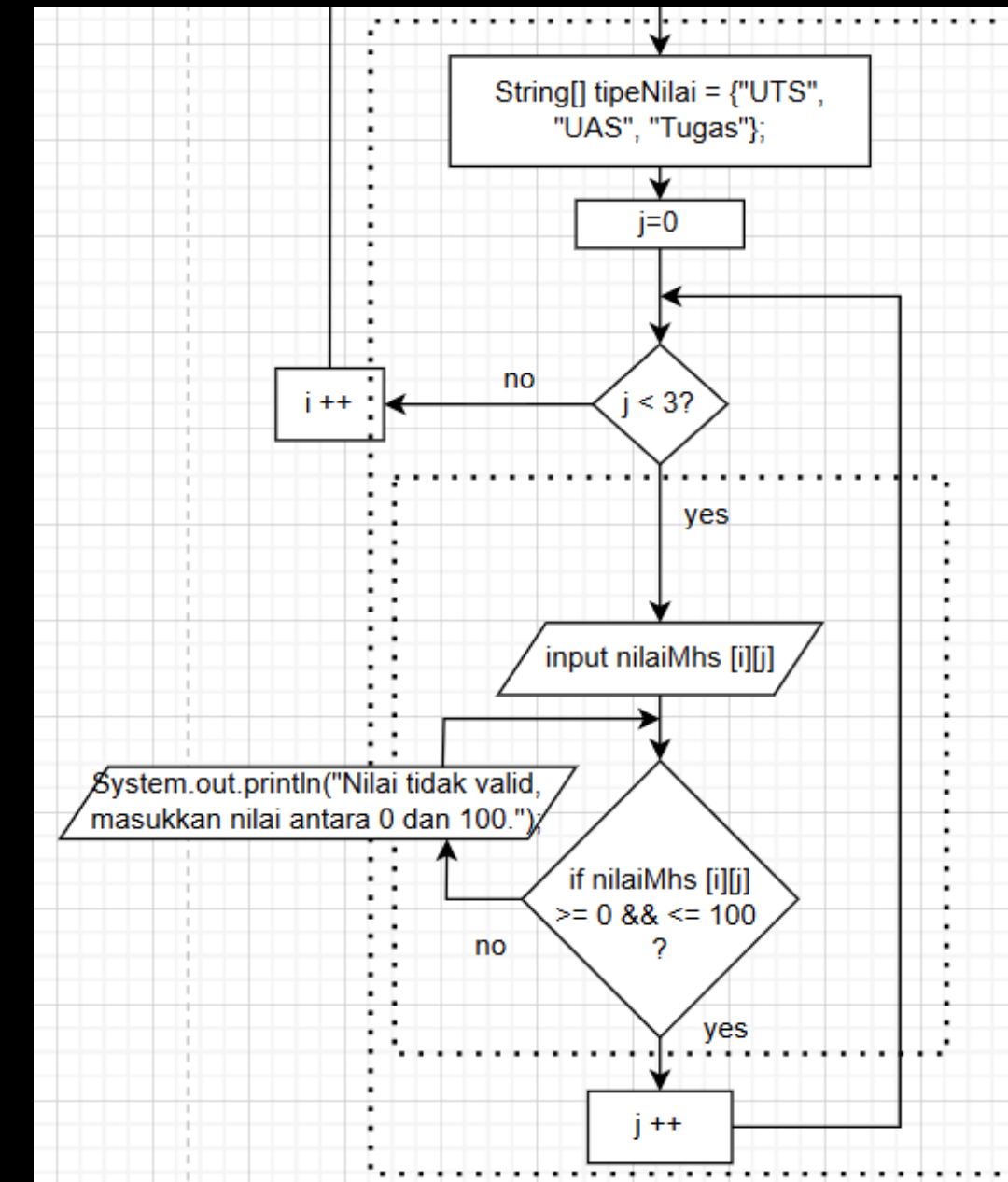
NIM: 244107061111

Masukkan nilai UTS (0-100): 80

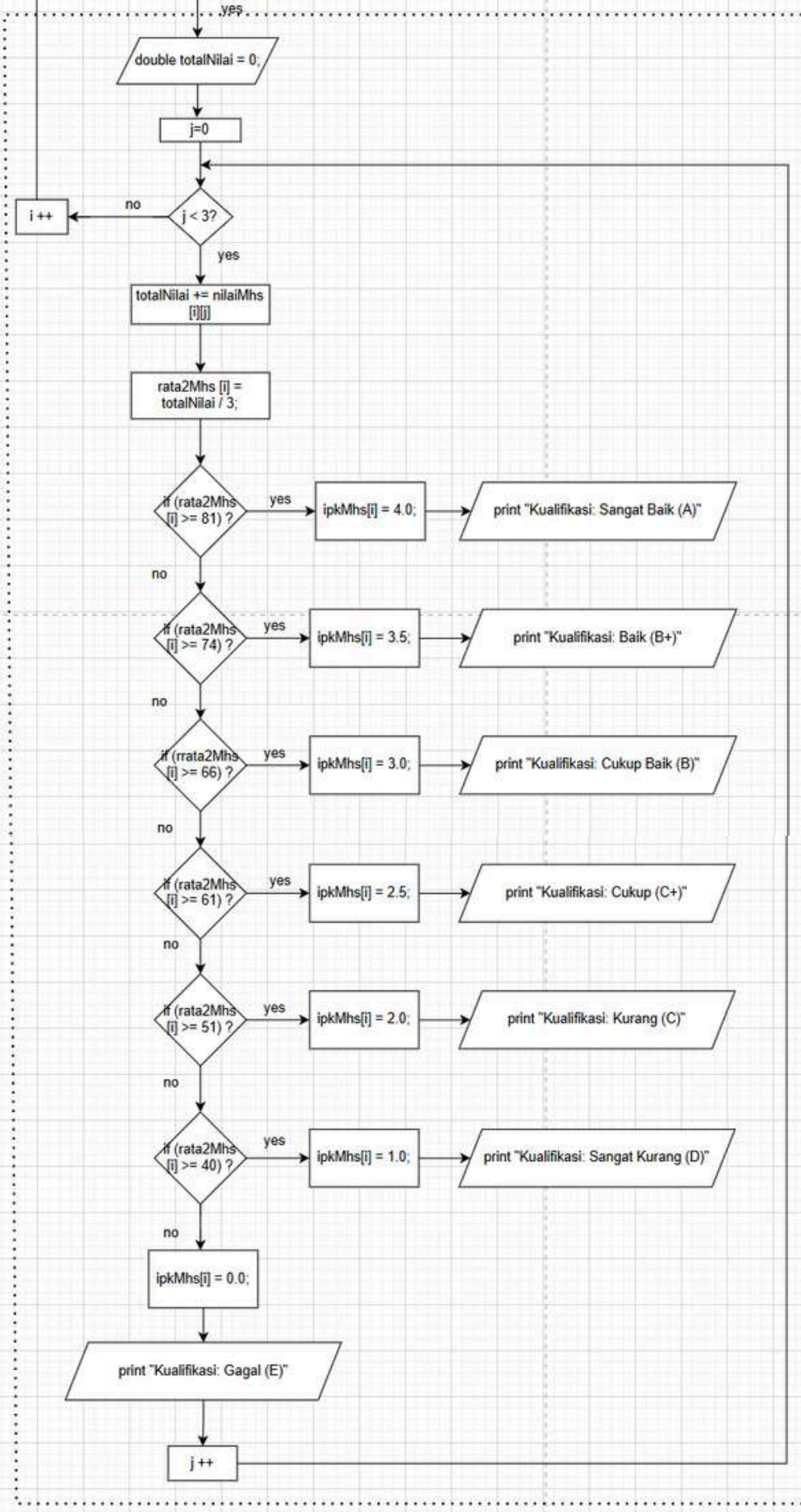
Masukkan nilai UAS (0-100): 70

Masukkan nilai Tugas (0-100): 60

Kualifikasi: Cukup Baik (B)



Our Process



```
rata2Mhs[i] = totalNilai / 3;
```

```
if (rata2Mhs[i] >= 81) {
    ipkMhs[i] = 4.0;
    System.out.println(x:"Kualifikasi: Sangat Baik (A)");
} else if (rata2Mhs[i] >= 74) {
    ipkMhs[i] = 3.5;
    System.out.println(x:"Kualifikasi: Baik (B+)");
} else if (rata2Mhs[i] >= 66) {
    ipkMhs[i] = 3.0;
    System.out.println(x:"Kualifikasi: Cukup Baik (B)");
} else if (rata2Mhs[i] >= 61) {
    ipkMhs[i] = 2.5;
    System.out.println(x:"Kualifikasi: Cukup (C+)");
} else if (rata2Mhs[i] >= 51) {
    ipkMhs[i] = 2.0;
    System.out.println(x:"Kualifikasi: Kurang (C)");
} else if (rata2Mhs[i] >= 40) {
    ipkMhs[i] = 1.0;
    System.out.println(x:"Kualifikasi: Sangat Kurang (D)");
} else {
    ipkMhs[i] = 0.0;
    System.out.println(x:"Kualifikasi: Gagal (E)");
}
```

```
sc.nextLine();
```

Input data untuk Mahasiswa ke-1
Nama: Dina Mei Lestari
NIM: 244107060105
Masukkan nilai UTS (0-100): 100
Masukkan nilai UAS (0-100): 90
Masukkan nilai Tugas (0-100): 90
Kualifikasi: Sangat Baik (A)

Input data untuk Mahasiswa ke-2
Nama: Feronika Wulandari
NIM: 244107061111
Masukkan nilai UTS (0-100): 80
Masukkan nilai UAS (0-100): 70
Masukkan nilai Tugas (0-100): 60
Kualifikasi: Cukup Baik (B)

- Menghitung rata-rata nilai: rata2Mhs[i] adalah hasil pembagian totalNilai dengan 3.
- Menentukan IPK dan kualifikasi: Berdasarkan rata-rata, IPK diberikan nilai sesuai skala (4.0 untuk Sangat Baik hingga 0.0 untuk Gagal).
- Menampilkan kualifikasi: Kualifikasi ditampilkan berdasarkan kategori nilai.


```

System.out.println(x:"\n=====");
System.out.println(x:"Data Mahasiswa dengan Nilai Akhir:");
for (int i = 0; i < jmlMhs; i++) {
    System.out.println("Nama: " + dataMhs[i][0] + " (NIM: " + dataMhs[i][1] + ")");
    System.out.println(x:"Kelas: SIB 1F");
    System.out.println(x:"Nilai:");
    for (int j = 0; j < 3; j++) {
        System.out.println(" - " + tipeNilai[j] + ": " + nilaiMhs[i][j]);
    }
    System.out.printf(format:"Rata-rata: %.2f | IPK: %.2f\n", rata2Mhs[i], ipkMhs[i]);
    System.out.println(x:"-----");
}

```

```

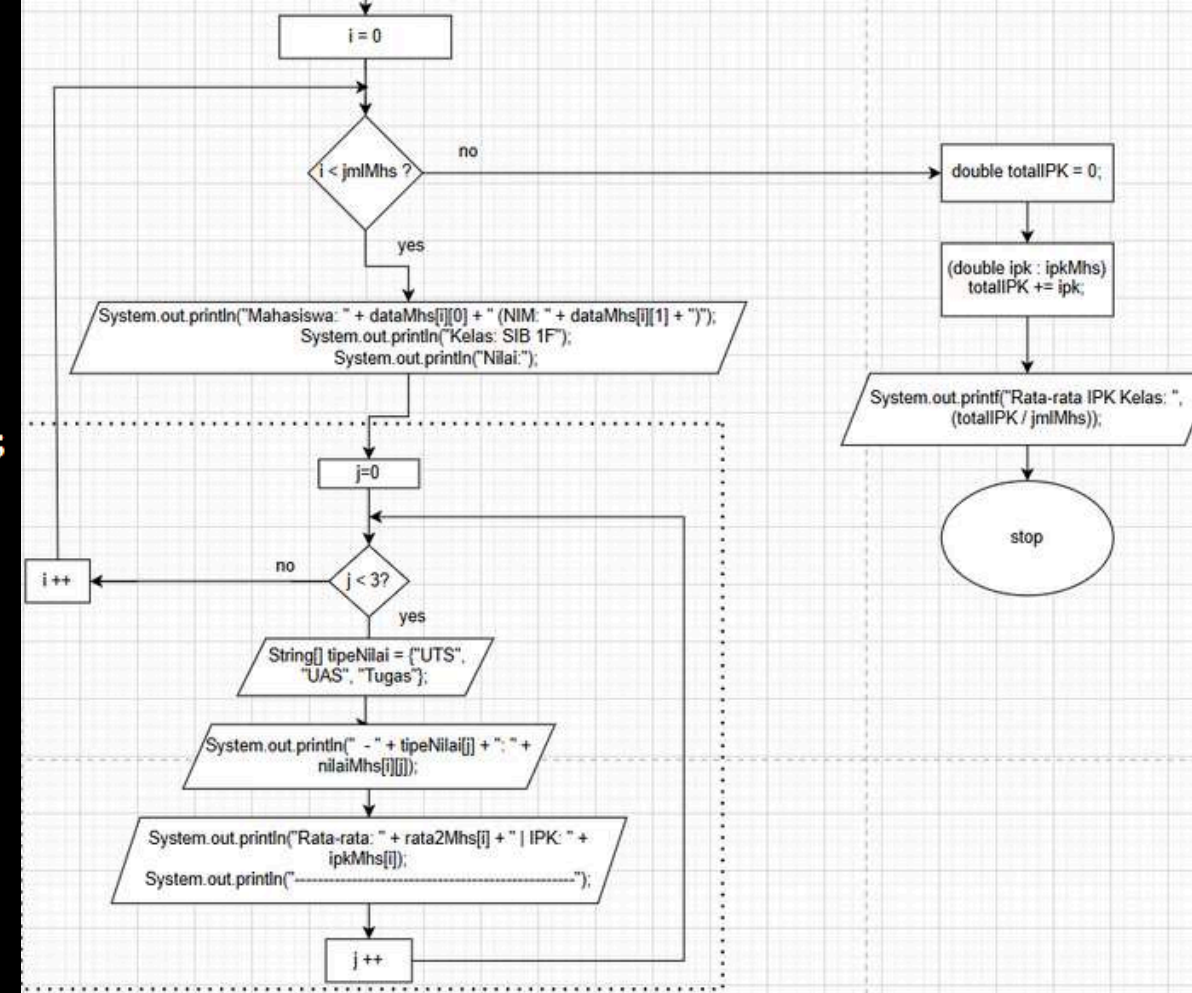
double totalIPK = 0;
for (double ipk : ipkMhs) {
    totalIPK += ipk;
}

```

```

System.out.println(x:"\n=====");
System.out.printf(format:"Rata-rata IPK Kelas: %.2f\n", (totalIPK / jmlMhs));

```



```

=====
Data Mahasiswa dengan Nilai Akhir:
Nama: Dina Mei Lestari (NIM: 244107060105)
Kelas: SIB 1F
Nilai:
    - UTS: 100.0
    - UAS: 90.0
    - Tugas: 90.0
Rata-rata: 93.33 | IPK: 4.00
-----

Nama: Feronika Wulandari (NIM: 244107061111)
Kelas: SIB 1F
Nilai:
    - UTS: 80.0
    - UAS: 70.0
    - Tugas: 60.0
Rata-rata: 70.00 | IPK: 3.00
-----

=====
Rata-rata IPK Kelas: 3.50

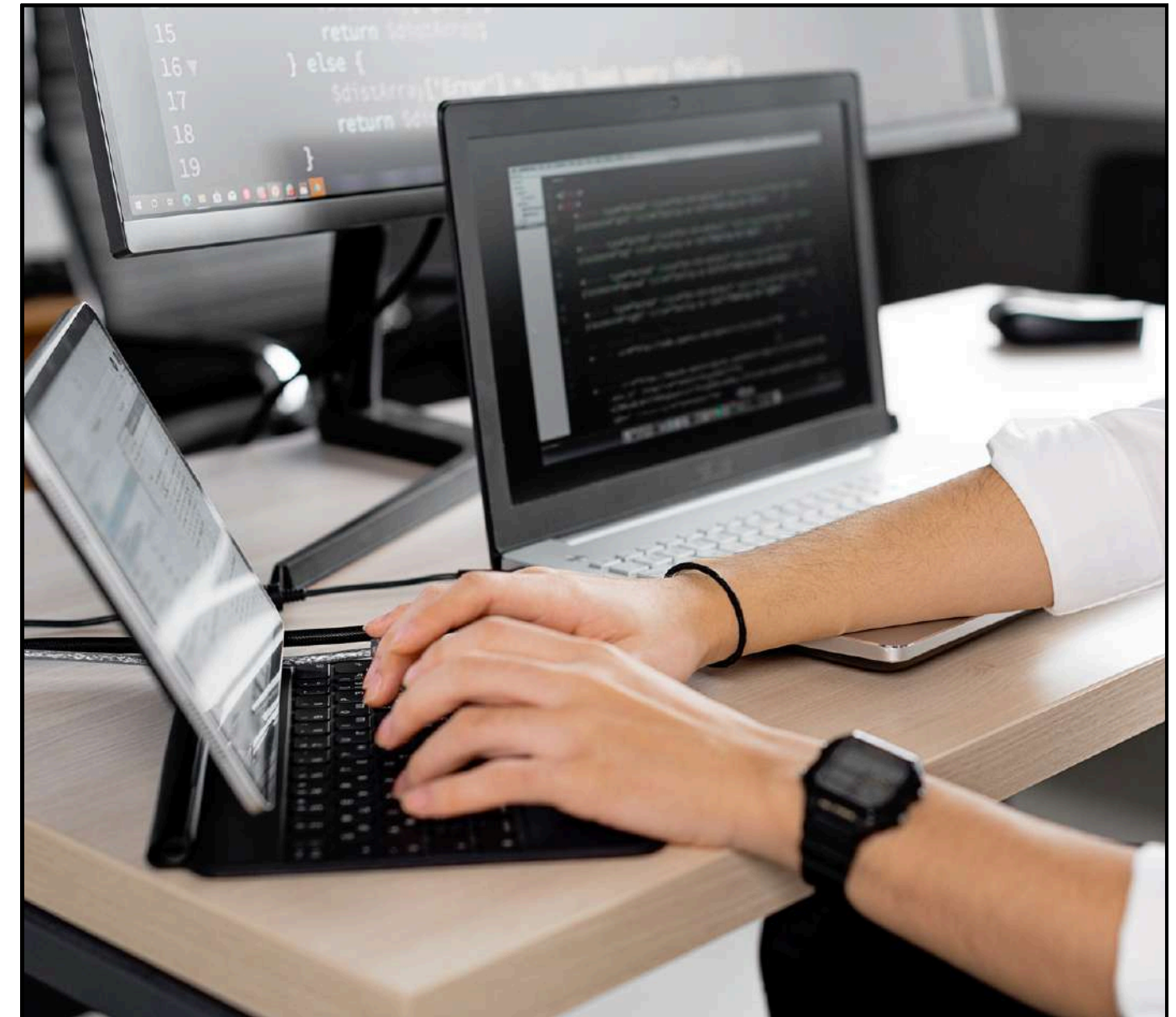
```

Our Process

- Menampilkan data mahasiswa: Loop ini mencetak informasi setiap mahasiswa, termasuk nama, NIM, kelas, dan nilai per kategori (UTS, UAS, Tugas).
- Rata-rata dan IPK: Hasil rata-rata nilai dan IPK ditampilkan dengan format desimal dua angka di belakang koma.
- Pemisah data: Garis horizontal digunakan untuk membedakan data antar mahasiswa.

Keunggulan Sistem

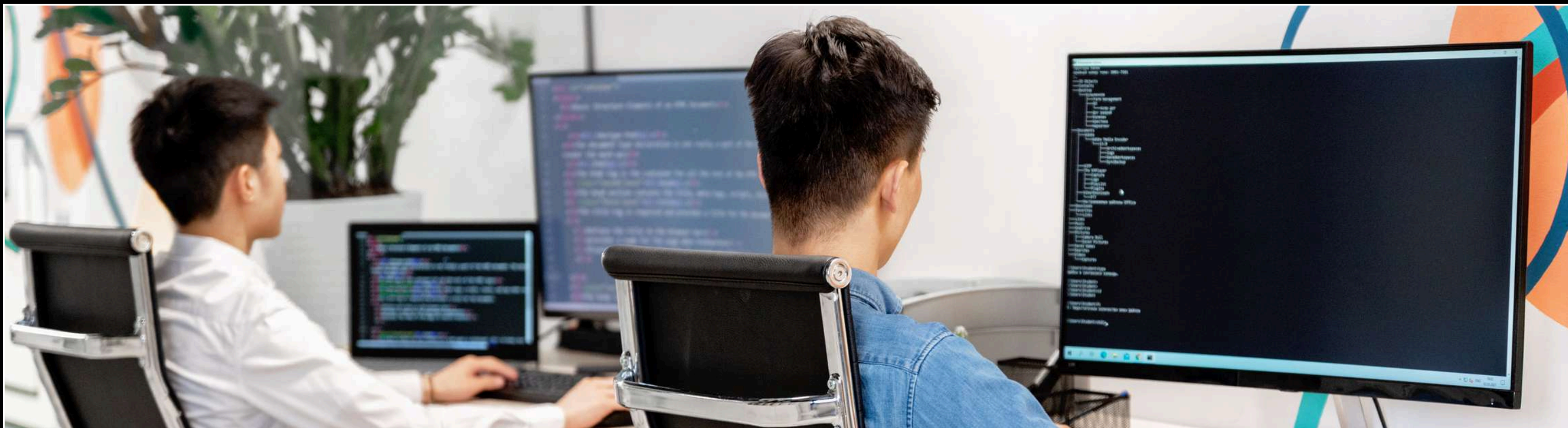
1. Efisiensi: Memproses data dengan cepat dan mengurangi beban kerja manual.
2. Akurasi: Mengurangi risiko kesalahan dalam perhitungan nilai dan IPK.
3. Fleksibilitas: Dapat diterapkan di berbagai perangkat komputer.
4. Kemudahan Analisis: Laporan yang dihasilkan memudahkan dalam evaluasi akademik.



Kesimpulan

- Sistem ini adalah solusi untuk manajemen data mahasiswa yang lebih cepat, akurat, dan efisien.
- Dengan implementasi yang sederhana, dapat memberikan dampak signifikan pada kelancaran administrasi akademik.





Grateful for Your Time!

FINISH



Let's Build Something Amazing Together.

KELOMPOK 2