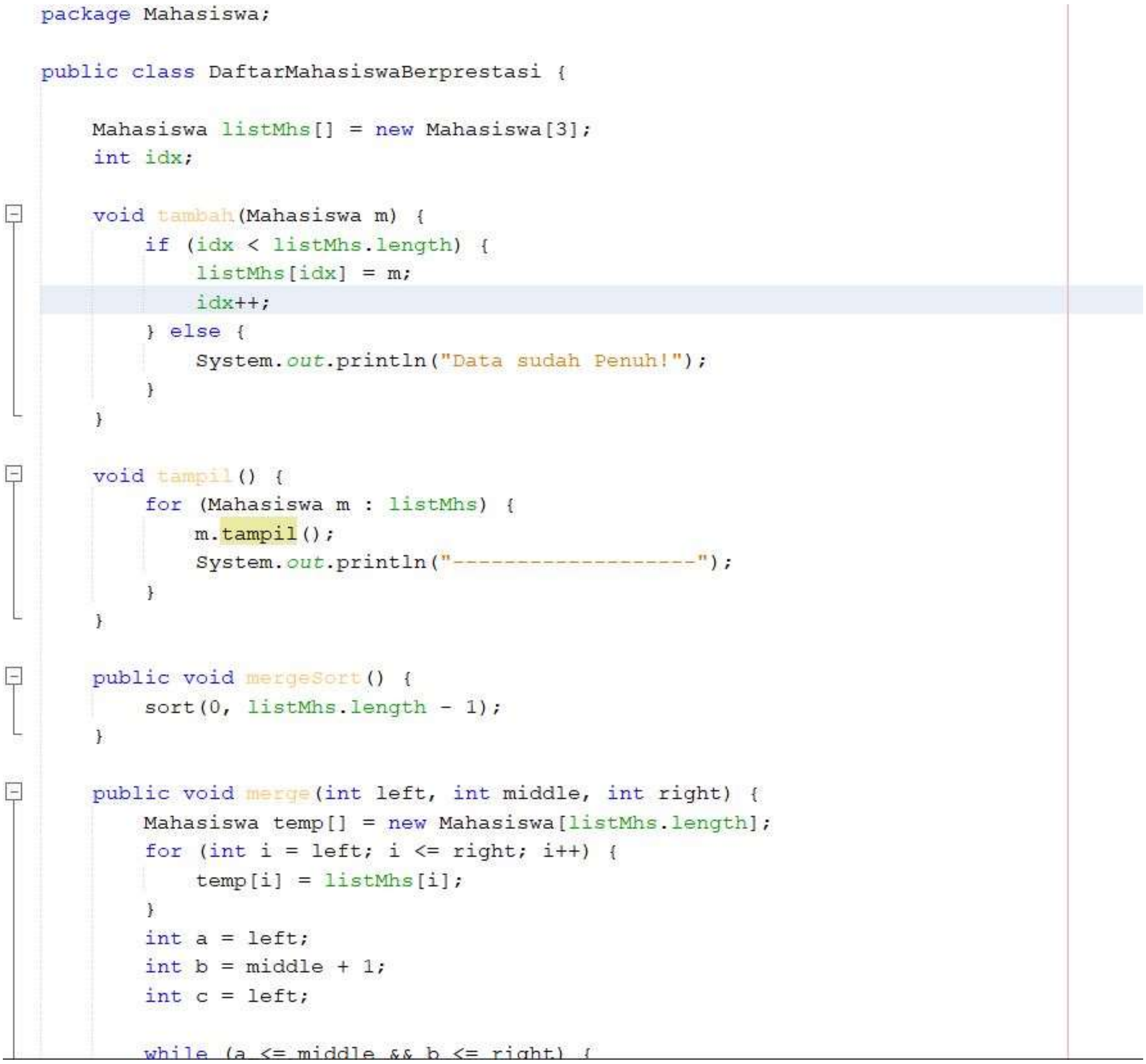
LAPORAN PRAKTIKUM ALGORITMA STRUKTUR DATA

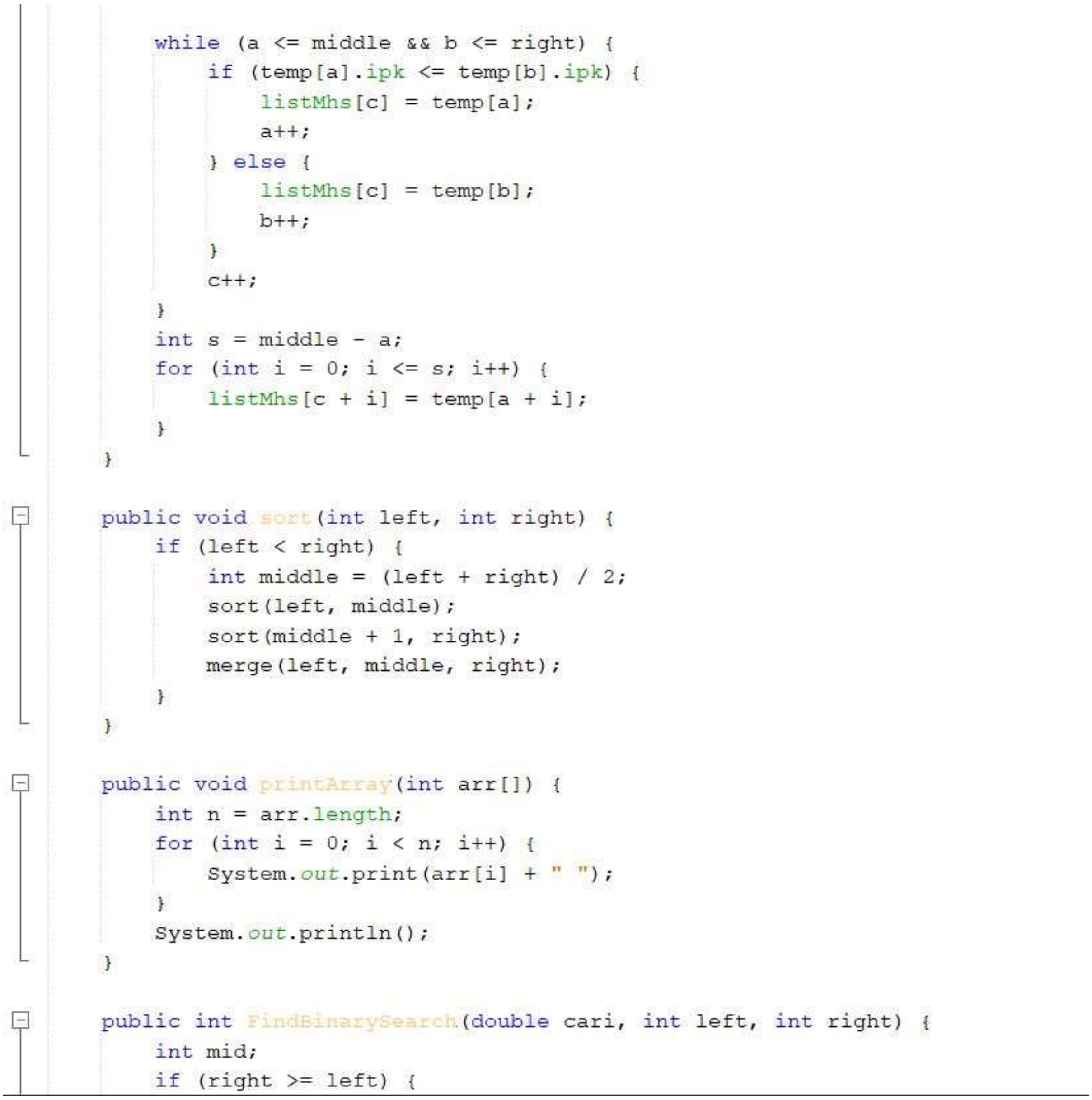
JOB SHEET 8

Oleh:
MOH. ZULFAN AKBAR NIM. 1941720152

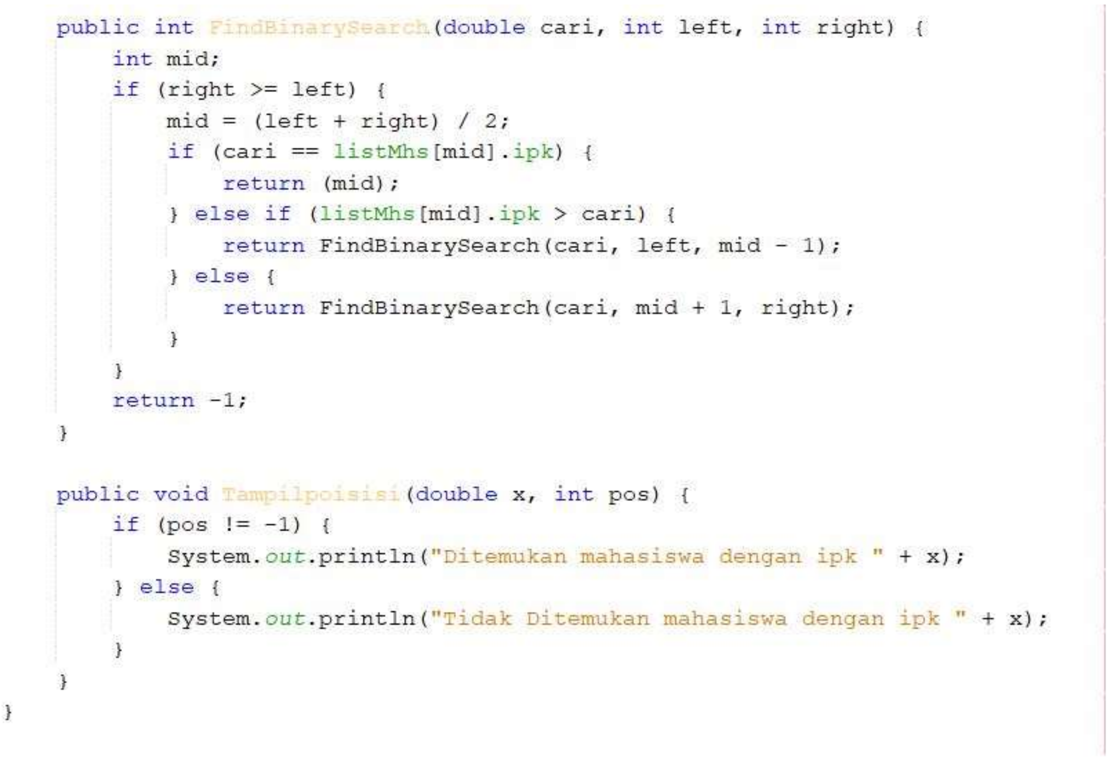
PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
APRIL 2020



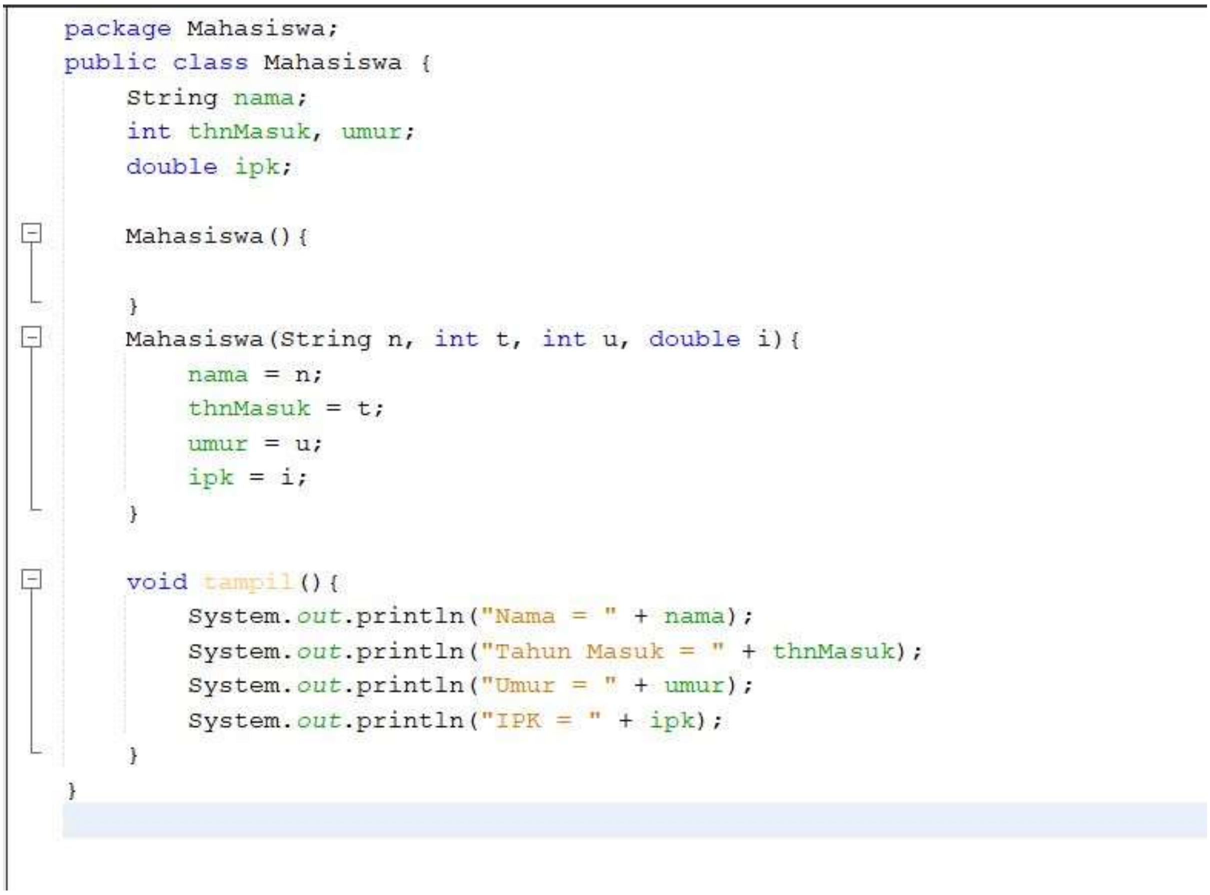
```java
package Mahasiswa;

public class DaftarMahasiswaBerprestasi {

    Mahasiswa listMhs[] = new Mahasiswa[3];
    int idx;

    void tambah(Mahasiswa m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data sudah Penuh!");
        }
    }

    void tampil() {
        for (Mahasiswa m : listMhs) {
            m.tampil();
            System.out.println("-------------------");
        }
    }

    public void mergeSort() {
        sort(0, listMhs.length - 1);
    }

    public void merge(int left, int middle, int right) {
        Mahasiswa temp[] = new Mahasiswa[listMhs.length];
        for (int i = left; i <= right; i++) {
            temp[i] = listMhs[i];
        }
        int a = left;
        int b = middle + 1;
        int c = left;

        while (a <= middle && b <= right) {
```



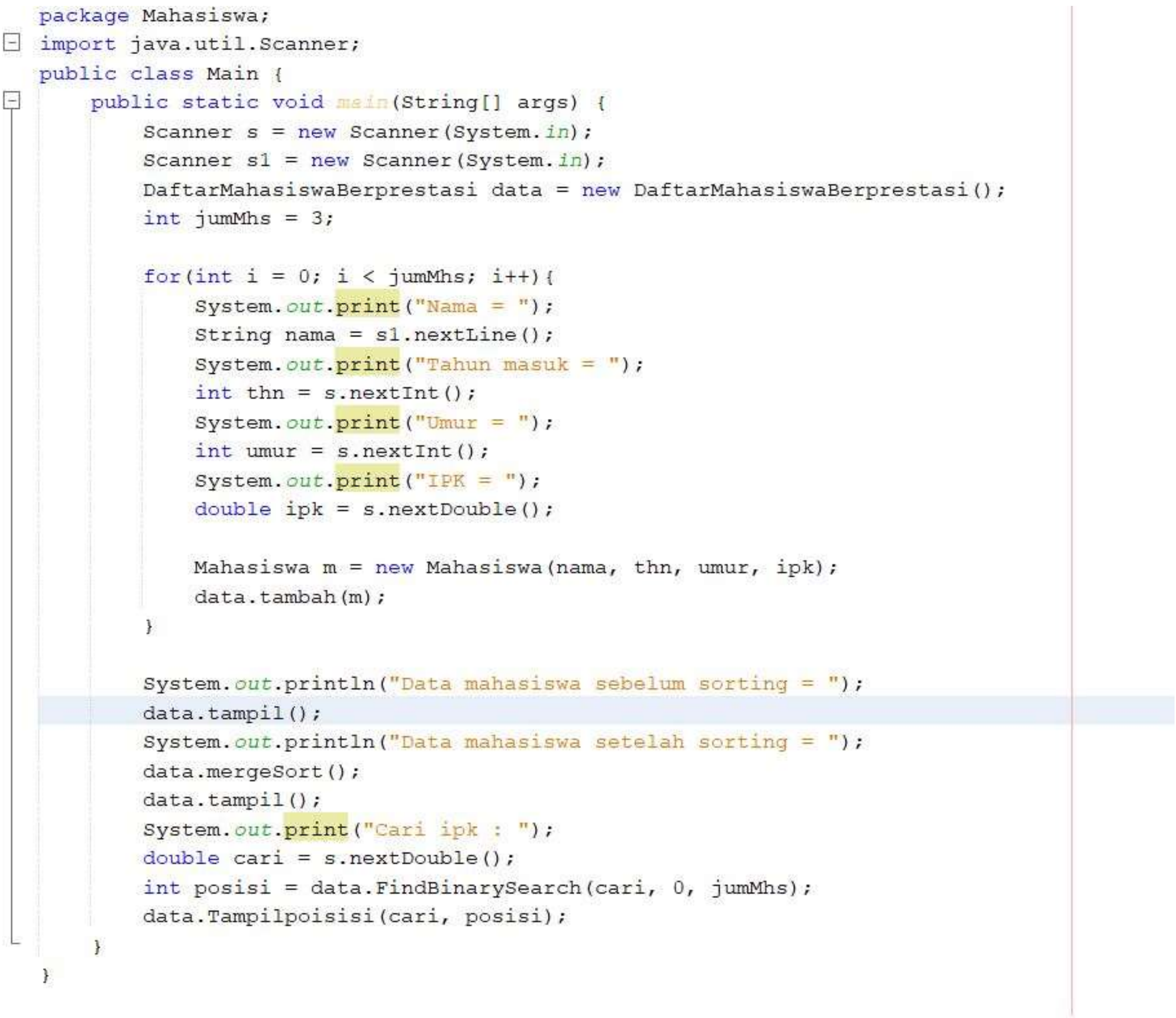
```
        while (a <= middle && b <= right) {
            if (temp[a].ipk <= temp[b].ipk) {
                listMhs[c] = temp[a];
                a++;
            } else {
                listMhs[c] = temp[b];
                b++;
            }
            c++;
        }
        int s = middle - a;
        for (int i = 0; i <= s; i++) {
            listMhs[c + i] = temp[a + i];
        }
    }

    public void sort(int left, int right) {
        if (left < right) {
            int middle = (left + right) / 2;
            sort(left, middle);
            sort(middle + 1, right);
            merge(left, middle, right);
        }
    }

    public void printArray(int arr[]) {
        int n = arr.length;
        for (int i = 0; i < n; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.println();
    }

    public int FindBinarySearch(double cari, int left, int right) {
        int mid;
        if (right >= left) {
```



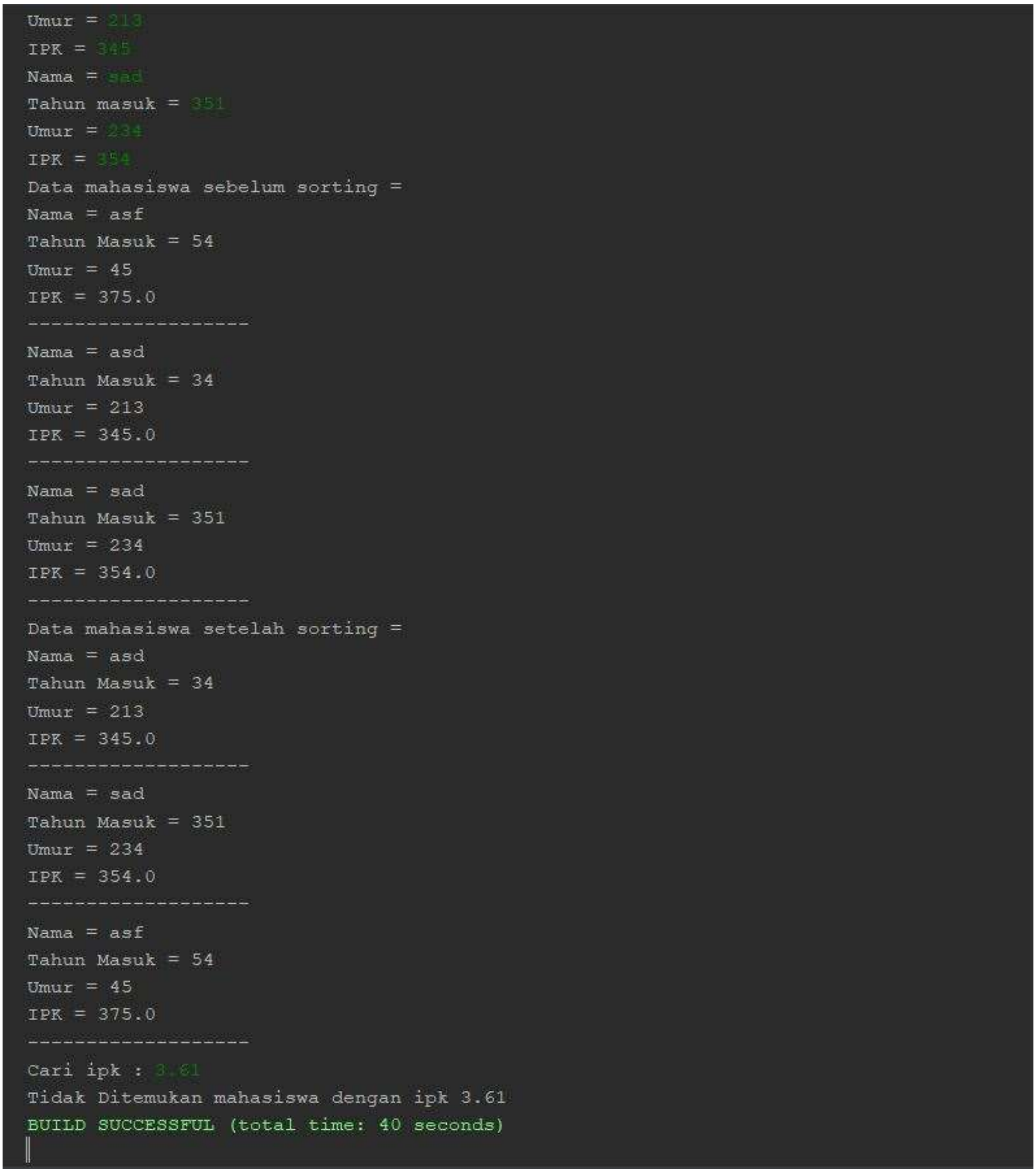
```java
    public int FindBinarySearch(double cari, int left, int right) {
        int mid;
        if (right >= left) {
            mid = (left + right) / 2;
            if (cari == listMhs[mid].ipk) {
                return (mid);
            } else if (listMhs[mid].ipk > cari) {
                return FindBinarySearch(cari, left, mid - 1);
            } else {
                return FindBinarySearch(cari, mid + 1, right);
            }
        }
        return -1;
    }

    public void Tampilpoisisi(double x, int pos) {
        if (pos != -1) {
            System.out.println("Ditemukan mahasiswa dengan ipk " + x);
        } else {
            System.out.println("Tidak Ditemukan mahasiswa dengan ipk " + x);
        }
    }
}
```



```java
package Mahasiswa;
public class Mahasiswa {
    String nama;
    int thnMasuk, umur;
    double ipk;

    Mahasiswa(){

    }
    Mahasiswa(String n, int t, int u, double i){
        nama = n;
        thnMasuk = t;
        umur = u;
        ipk = i;
    }

    void tampil(){
        System.out.println("Nama = " + nama);
        System.out.println("Tahun Masuk = " + thnMasuk);
        System.out.println("Umur = " + umur);
        System.out.println("IPK = " + ipk);
    }
}
```



```
package Mahasiswa;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        Scanner s1 = new Scanner(System.in);
        DaftarMahasiswaBerprestasi data = new DaftarMahasiswaBerprestasi();
        int jumMhs = 3;

        for(int i = 0; i < jumMhs; i++){
            System.out.print("Nama = ");
            String nama = s1.nextLine();
            System.out.print("Tahun masuk = ");
            int thn = s.nextInt();
            System.out.print("Umur = ");
            int umur = s.nextInt();
            System.out.print("IPK = ");
            double ipk = s.nextDouble();

            Mahasiswa m = new Mahasiswa(nama, thn, umur, ipk);
            data.tambah(m);
        }

        System.out.println("Data mahasiswa sebelum sorting = ");
        data.tampil();
        System.out.println("Data mahasiswa setelah sorting = ");
        data.mergeSort();
        data.tampil();
        System.out.print("Cari ipk : ");
        double cari = s.nextDouble();
        int posisi = data.FindBinarySearch(cari, 0, jumMhs);
        data.Tampilpoisisi(cari, posisi);
    }
}
```



```
Umur = 213
IPK = 345
Nama = sad
Tahun masuk = 351
Umur = 234
IPK = 354
Data mahasiswa sebelum sorting =
Nama = asf
Tahun Masuk = 54
Umur = 45
IPK = 375.0
-------------------
Nama = asd
Tahun Masuk = 34
Umur = 213
IPK = 345.0
-------------------
Nama = sad
Tahun Masuk = 351
Umur = 234
IPK = 354.0
-------------------
Data mahasiswa setelah sorting =
Nama = asd
Tahun Masuk = 34
Umur = 213
IPK = 345.0
-------------------
Nama = sad
Tahun Masuk = 351
Umur = 234
IPK = 354.0
-------------------
Nama = asf
Tahun Masuk = 54
Umur = 45
IPK = 375.0
-------------------
Cari ipk : 3.61
Tidak Ditemukan mahasiswa dengan ipk 3.61
BUILD SUCCESSFUL (total time: 40 seconds)
```