

1. – Selisih Waktu :

```
public class Waktu {
    int detik;
    int menit;
    int jam;

    public Waktu(int jam, int menit, int detik){
        this.jam = jam;
        this.menit = menit;
        this.detik = detik;
    }
    public static void main(String[] args) throws Exception {
        // membuat objek dari class waktu
        Waktu mulai = new Waktu(8, 12,15);
        Waktu selesai = new Waktu(12, 34,55);
        Waktu selisih;

        //memanggil method perbedaan
        selisih = perbedaan(mulai,selesai);
        System.out.printf("Selisih Waktu: %d:%d:%d - ",mulai.jam,mulai.menit,mulai.detik);
        System.out.printf("%d:%d:%d ",selesai.jam, selesai.menit,selesai.detik);
        System.out.printf("= %d:%d:%d\n ",selisih.jam, selisih.menit,selisih.detik);
    }
    private static Waktu perbedaan(Waktu mulai, Waktu selesai) {
        Waktu selisih = new Waktu(0, 0, 0);

        //jika detik mulai lebih dari besar
        //konversi menit selesai ke dalam detik
        // dan tambahkan detik ke detik selesai
        if(mulai.detik > selesai.detik){
            --selesai.menit;
            selesai.detik += 60;
        }
        selisih.detik = selesai.detik - mulai.detik;
        //jika menit mulai lebih besar
        //konversikan jam selesai ke dalam menit
        //kemudian tambahkan ke dalam menit selesai
        if(mulai.menit > selesai.menit){
            --selesai.jam;
            selesai.menit += 60;
        }
        selisih.menit = selesai.menit - mulai.menit;
        selisih.jam = selesai.jam - mulai.jam;

        //return the difference time
        return(selisih);
    }
}
```

- 2 bilangan Kompleks :

```
public class Complex {
    double real;
    double imag;

    public Complex(double real, double imag) {
        this.real = real;
        this.imag = imag;
    }
    public static void main(String[] args) throws Exception {
        Complex n1 = new Complex(2.3, 4.5),
            n2 = new Complex(3.4, 5.0),
            temp;

        temp = add(n1,n2);
        System.out.printf("Sum = %.1f + %.1fi", temp.real,temp.imag);
    }
    public static Complex add(Complex n1, Complex n2){
        Complex temp = new Complex(0.0, 0.0);
        temp.real = n1.real + n2.real;
        temp.imag = n1.imag + n2.imag;

        return(temp);
    }
}
```

2. Ilustrasi untuk alur kerja program Waktu:

1. Program dimulai dengan memanggil method main dalam class Waktu.
2. Dalam method main, dua objek Waktu dibuat menggunakan konstruktor, yaitu objek mulai dan objek selesai.
3. Objek-objek tersebut digunakan sebagai argumen dalam pemanggilan method perbedaan.
4. Pada method perbedaan, objek selisih dibuat menggunakan konstruktor Waktu.
5. Variabel selisih.detik dihitung terlebih dahulu dengan membandingkan nilai detik pada objek mulai dan selesai.
6. Jika nilai detik pada objek mulai lebih besar dari nilai detik pada objek selesai, maka nilai menit pada objek selesai dikurangi 1 dan nilai detik pada objek selesai ditambahkan 60.
7. Selanjutnya, variabel selisih.minut dihitung dengan membandingkan nilai menit pada objek mulai dan selesai.
8. Jika nilai menit pada objek mulai lebih besar dari nilai menit pada objek selesai, maka nilai jam pada objek selesai dikurangi 1 dan nilai menit pada objek selesai ditambahkan 60.
9. Terakhir, variabel selisih.jam dihitung dengan mengurangi nilai jam pada objek mulai dari nilai jam pada objek selesai.

10. Method perbedaan mengembalikan objek selisih, yang kemudian disimpan pada variabel selisih dalam method main.
11. Hasil perhitungan selisih waktu dicetak ke layar.
12. Ilustrasi untuk alur kerja program Complex:

Program dimulai dengan memanggil method main dalam class Complex.

1. Dalam method main, dua objek Complex dibuat menggunakan konstruktor, yaitu objek n1 dan objek n2.
 2. Objek-objek tersebut digunakan sebagai argumen dalam pemanggilan method add.
 3. Pada method add, objek temp dibuat menggunakan konstruktor Complex.
 4. Variabel temp.real dihitung dengan menjumlahkan nilai real pada objek n1 dan n2.
 5. Variabel temp.imag dihitung dengan menjumlahkan nilai imag pada objek n1 dan n2.
 6. Method add mengembalikan objek temp, yang kemudian disimpan pada variabel temp dalam method main.
 7. Hasil penjumlahan bilangan kompleks dicetak ke layar.
3. Variabel detik yang dideklarasikan di dalam class Waktu bersifat global atau instance variable, yang berarti variabel tersebut dapat diakses oleh seluruh method atau konstruktor dalam class Waktu.

Sementara itu, variabel detik yang dideklarasikan di dalam konstruktor Waktu merupakan parameter input yang digunakan untuk menginisialisasi nilai dari variabel detik di dalam class Waktu. Variabel detik di dalam konstruktor hanya bersifat lokal, artinya hanya dapat diakses di dalam konstruktor tersebut. Setelah nilai variabel detik di dalam konstruktor diatribusikan ke variabel detik di dalam class Waktu menggunakan this.detik, variabel detik di dalam konstruktor tidak lagi memiliki pengaruh pada program.