

Nama & NPM	Topik:	Tanggal:
Iqbal Tri Agustian G1F024043	Unit 1 Kelas (class)	12 September 2024
[Nomor Soal 1] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variabel</p> <pre> public class Manusia { // deklarasi kelas //deklarasi atribut Manusia dalam variabel String nama, rambut; //deklarasi constructor public Manusia1 (String nama) { System.out.println(" Nama saya : "+ nama + "\n Warna Rambut : " + rambut); } //deklarasi method utama public static void main(String[] args) { Manusia1 satu = new Manusia1("Putri", "hitam"); } } </pre> <p>Luaran 1:</p> <pre> Exception in thread "main" java.lang.Error: Unresolved compilation problem: The constructor Manusia1(String, String) is undefined at Manusia1.main(Manusia1.java:13) </pre> <p>Latihan 1:</p> <ol style="list-style-type: none"> 1.1. Perbaiki pesan kesalahan Contoh 1! 1.2. Analisa ciri-ciri lain Kelas Manusia yang dapat menjadi <ol style="list-style-type: none"> a. atribut variabel, dan b. perilaku/ behavior! <p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <p>Video Materi 1 tentang Kelas, Objek, Method https://www.youtube.com/watch?v=60IdOc8m8Es</p>		
[Nomor Soal 1] Analisis dan Argumentasi		
<p>1) Uraikan rancangan solusi yang diusulkan.</p> <ul style="list-style-type: none"> ➤ 1.1 Dengan mengganti codingan deklarasi konstrutor dan deklarasi method utama, karena pada deklarasi konstrutor diberi nama Manusia1 yang mana namanya tidak sama dengan nama class dan juga tidak mendelarasri variabel rambut. Pada method utama juga sama, nama yang digunakan berbeda dengan nama claass. ➤ 1.2 Atribut variable ➤ a. Atribut variable merupakan karakteristik yang dimiliki suatu objek, atribut ini mendefinisikan keadaan atau informasi yang berkaitan dengan objek tersebut. Pada soal terdapat atribut variable berupa nama dan rambut. 		

- Nama : Atribut ini menyimpan nama orang. Tipe data yang digunakan adalah String.
 - Rambut : Atribut ini menyimpan informasi tentang warna rambut. Tipe data yang digunakan juga String
 - Jenis kelamin: Atribut ini menyimpan informasi tentang jenis kelamin. Tipe data yang digunakan juga String
 - Alamat: Atribut ini menyimpan informasi tentang alamat. Tipe data yang digunakan juga String
 - Umur: Atribut ini menyimpan informasi tentang umur. Tipe data yang digunakan int
 - Tinggi badan : Atribut ini menyimpan informasi tentang . Tipetinggi badan data yang digunakan double
 - Berat badan : Atribut ini menyimpan informasi tentang umur. Tipe data yang digunakan juga double
- b. Perilaku/ behavior untuk method Perilaku atau behavior merujuk pada tindakan atau fungsi yang dapat dilakukan oleh objek. Hal ini diimplementasikan dalam bentuk method dalam kelas, metode ini memungkinkan objek untuk melakukan operasi tertentu dan berinteraksi dengan atribut atau objek lainnya.
- 2) Analisis solusi, kaitkan dengan permasalahan.
- Karena pada pemrograman nama method harus sama dengan nama class, jika nama method berbeda dengan nama class maka program tidak dapat berjalan.

[Nomor Soa 1] Penyusunan Algoritma dan Kode Program

- 1) Rancang desain solusi atau algoritma
 - Membuat nama class
 - Mendeklarasi konstruktor
 - Mendeklarasi method utama
- 2) Tuliskan kode program dan luaran

(Codingan 1.1)

```

1  public class Manusia { // deklarasi kelas
2      //deklarasi atribut Manusia dalam variabel
3      String nama, rambut;
4
5      //deklarasi constructor
6      public Manusia (String nama, String rambut) {
7          System.out.println(" Nama saya : "+ nama + "\n Warna Rambut : " + rambut);
8      }
9
10     //deklarasi method utama
11     public static void main( String[] args) {
12         Manusia Satu = new Manusia("Putri", "hitam");
13         Manusia Dua = new Manusia("Budi", "kuning");
14     }
15 }

```

(Luaran 1.2)

```
[Running] cd "d:\praktikumkoprog\" && javac Manusia.java && java Manusia
Nama saya : Putri
Warna Rambut : hitam
Nama saya : Budi
Warna Rambut : kuning
[Done] exited with code=0 in 7.243 seconds
```

- Luaran yang dihasilkan sudah sesuai dengan perintah

(Codingan 1.2)

```
1 public class Manusia {
2     // Deklarasi atribut Manusia dalam variabel
3     String nama, rambut, jenisKelamin, alamat;
4     int umur;
5     double tinggiBadan, beratBadan;
6
7     // Deklarasi constructor
8     public Manusia(String nama, String rambut, String jenisKelamin, int umur, double tinggiBadan, double beratBadan, String alamat) {
9         this.nama = nama;
10        this.rambut = rambut;
11        this.jenisKelamin = jenisKelamin;
12        this.umur = umur;
13        this.tinggiBadan = tinggiBadan;
14        this.beratBadan = beratBadan;
15        this.alamat = alamat;
16    }
17
18    // Metode untuk memperkenalkan diri
19    public void perkenalanDiri() {
20        System.out.println("Halo, nama saya " + nama + ", umur saya " + umur + " tahun, rambut saya berwarna " + rambut + ".");
21        System.out.println("Jenis kelamin: " + jenisKelamin + ", tinggi badan: " + tinggiBadan + " cm, berat badan: " + beratBadan + " kg.");
22        System.out.println("Saya berasal dari " + alamat + ".\n");
23    }
24
25    // Metode untuk berbicara
26    public void berbicara(String kata) {
27        System.out.println(nama + " berkata: " + kata);
28    }
29
30    // Metode untuk berjalan
31    public void mengerjakanTugas() {
32        System.out.println(nama + " sedang mengerjakan tugas.");
33    }
34
35    // Metode untuk makan
36    public void makan(String makanan) {
37        System.out.println(nama + " sedang makan " + makanan + ".");
38    }
39
40    // Metode untuk tidur
41    public void minumJusBuah() {
42        System.out.println(nama + " sedang minum jus buah.");
43    }
44
45    // Deklarasi method utama
46    public static void main(String[] args) {
47        // Membuat objek Manusia
48        Manusia manusia1 = new Manusia("Zahra", "Hitam", "Perempuan", 18, 155.5, 55.0, "Bengkulu");
49        Manusia manusia2 = new Manusia("Iqbal", "Pink", "Laki-laki", 18, 174.3, 65.4, "Lubuklinggau");
50
51        // Memanggil metode untuk setiap objek
52        manusia1.perkenalanDiri();
53        manusia1.berbicara("Halo guys, salam kenal");
54        manusia1.mengerjakanTugas();
55        manusia1.makan("Mie Soto");
56        manusia1.minumJusBuah();
57
58        System.out.println(); // Untuk memberikan jarak antara output
59
60        manusia2.perkenalanDiri();
61        manusia2.berbicara("Salam kenal juga guys");
62        manusia2.mengerjakanTugas();
63        manusia2.makan("Mie Benda");
64        manusia2.minumJusBuah();
65    }
66 }
67
```

(Luaran 1.2)

```
[Running] cd "d:\praktikumkoprog\" && javac Manusia.java && java Manusia
Halo, nama saya Zahra, umur saya 18 tahun, rambut saya berwarna Hitam.
Jenis kelamin: Perempuan, tinggi badan: 155.5 cm, berat badan: 55.0 kg.
Saya berasal dari Bengkulu.

Zahra berkata: Hallo guyss, salam kenal
Zahra sedang mengerjakan tugas.
Zahra sedang makan Mie Soto.
Zahra sedang minum jus buah.

Halo, nama saya Iqbal, umur saya 18 tahun, rambut saya berwarna Pink.
Jenis kelamin: Laki-laki, tinggi badan: 174.3 cm, berat badan: 65.4 kg.
Saya berasal dari Lubuklinggau.

Iqbal berkata: Salam kenal juga guysss
Iqbal sedang mengerjakan tugas.
Iqbal sedang makan Mie Rendang.
Iqbal sedang minum jus buah.

[Done] exited with code=0 in 5.427 seconds
```

- Luaran yang dihasilkan sudah sesuai dengan perintah

[Nomor Soal 1] Kesimpulan

- 1) Analisa
 - a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
- Jika mendeklarasi method nama yang digunakan harus nama dengan nama class, jika nama yang digunakan berbeda dengan nama class maka program tersebut tidak akan berjalan.
 - b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Nama & NPM	Topik:	Tanggal:
Iqbal Tri Agustian G1F024043	Unit 2 Objek	14 September 2024
[Nomor Soal 2] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variabel</p> <pre> public class Ortu { //deklarasi constructor (variabel constructor) public Ortu { //nama dan rambut adalah variabel constructor System.out.println(" Nama saya : "+ nama + "\n Warna Rambut : " + rambut); } public static void main (String[] args) { Ortu satu = new Ortu("Putri", "hitam"); } } </pre> <p>Luaran 2:</p> <pre> Exception in thread "main" java.lang.Error: Unresolved compilation problem: The constructor Ortu(String, String) is undefined at Ortu.main(Ortu.java:9) </pre> <p>Latihan 2:</p> <p>2.1. Evaluasi penyebab kesalahan dan perbaiki kode tersebut!</p> <p>2.2. Apabila Ortu memiliki data variabel umur = 25 dan jenis kelamin = P (untuk Perempuan), rekomendasikan constructor dengan parameter yang baru untuk ditambahkan dalam program</p>		
<p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <p>Video Materi 1 tentang Kelas, Objek, Method https://www.youtube.com/watch?v=60IdOc8m8Es</p>		
[Nomor Soal] Analisis dan Argumentasi		
<p>1) Uraikan rancangan solusi yang diusulkan.</p> <ul style="list-style-type: none"> ➤ 2.1 Dengan mengganti codingan deklarasi konstruktor , karena pada deklarasi konstruktor diberi nama Ortu yang mana namanya tidak sama dengan nama class dan juga tidak mendeklarasi variabel String nama , String rambut. ➤ 2.2 Menambahkan tipe data int untuk umur dan char untuk jenis kelamin(p) <p>2) Analisis solusi, kaitkan dengan permasalahan.</p> <ul style="list-style-type: none"> ➤ Karena pada pemrograman saat kita membuat nama pada deklarasi yang dibuat nama harus sama dengan nama kelas 		
[Nomor Soal 2] Penyusunan Algoritma dan Kode Program		
<p>1) Rancang desain solusi atau algoritma</p> <ul style="list-style-type: none"> ➤ Membuat class ➤ Mendeklarasi konstruktor ➤ Mendeklarasi method utama 		

2) Tuliskan kode program dan luaran

(Codingan 2.1)

```
1 public class Ortu {
2     //deklarasi constructor (variabel constructor)
3     public Ortu (String nama, String rambut) {
4         //nama dan rambut adalah variabel constructor
5         System.out.println(" Nama saya : "+ nama +
6             "\n Warna Rambut : " + rambut);
7     }
8     public static void main (String[] args) {
9         Ortu satu = new Ortu("Putri", "hitam");
10    }
11 }
```

(Luaran 2.1)

```
[Running] cd "d:\praktikumkoprog\" && javac Ortu.java && java Ortu
Nama saya : Putri
Warna Rambut : hitam
[Done] exited with code=0 in 9.242 seconds
```

➤ Luaran yang dihasilkan sudah sesuai dengan perintah

(Codingan 2.2)

```
1 public class Ortu {
2     //deklarasi constructor (variabel constructor)
3     public Ortu (String nama, String rambut, int umur, char jenisKelamin) {
4         //nama dan rambut adalah variabel constructor
5         System.out.println(" Nama saya : "+ nama +
6             "\n Warna Rambut : " + rambut + "\n Umur : " + umur + "\n Jenis kelamin : " + jenisKelamin);
7     }
8     public static void main (String[] args) {
9         Ortu satu = new Ortu("Putri", "hitam", 25, 'p');
10    }
11 }
```

(Luaran 2.2)

```
[Running] cd "d:\praktikumkoprog\" && javac Ortu.java && java Ortu
Nama saya : Putri
Warna Rambut : hitam
Umur : 25
Jenis kelamin : p
[Done] exited with code=0 in 6.98 seconds
```

➤ Luaran yang dihasilkan sudah sesuai dengan perintah

[Nomor Soal 2] Kesimpulan

1) Analisa

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
- Jika mendeklarasi method nama yang digunakan harus sama dengan nama class, jika nama yang digunakan berbeda dengan nama class maka program tersebut tidak akan berjalan.
- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Nama & NPM	Topik:	Tanggal:
Iqbal Tri Agustian G1F024043	Unit 3 Method	14 September 2024

[Nomor Soal 3] Identifikasi Masalah:

1) Uraikan permasalahan dan variabel

```
public class Manusia {
    //deklarasi atribut Manusia dalam variabel
    String nama, rambut;

    //deklarasi constructor
    public Manusia(String nama, String rambut) {
        System.out.println(" Nama saya : "+ nama +
            "\n Warna Rambut : " + rambut);
    }

    //deklarasi method
    void sukaNonton {
        System.out.println(" Hobi Menonton : " + film);
    }

    int sukaNonton {
        episode*durasi;
    }

    //deklarasi method utama
    public static void main( String[] args) {
        Manusia satu = new Manusia("Putri", "hitam");
        satu.sukaNonton("Drakor");
        int jumlahJam = satu.sukaNonton(2, 2);
        System.out.println("Jam nonton = " +jumlahJam + " jam");
    }
}
```

Luaran 3:

```
Exception in thread "main" java.lang.Error: Unresolved compilation problems:
The method sukaNonton(String) is undefined for the type Manusia1
The method sukaNonton(int, int) is undefined for the type Manusia1
at Manusia1.main(Manusia1.java:23)
```

Latihan 3:

- 3.1. Evaluasi penyebab kesalahan dan perbaiki kode tersebut!
- 3.2. Ubahlah method dan constructor Contoh 3 sesuai dengan perilaku/ behavior anda
- 3.3. Berdasarkan Contoh 3 dan Latihan 3.2. simpulkan perbedaan:
 - a) constructor overloading dan overriding
 - b) method overloading, dan method overriding
 - c) method yang mengembalikan nilai dan method tidak mengembalikan nilai

2) Rincikan sumber informasi yang relevan (buku / webpage)

[Video Materi 1 tentang Kelas, Objek, Method https://www.youtube.com/watch?v=60IdOc8m8Es](https://www.youtube.com/watch?v=60IdOc8m8Es)

[Nomor Soal] Analisis dan Argumentasi

1) Uraikan rancangan solusi yang diusulkan.

- 3.1 Mengganti nama deklarasi konstruktor sama dengan nama kelas
- mendefinisikan dua metode overload untuk sukaNonton:
 - Satu metode yang menerima parameter String dan mencetak genre film favorit.
 - Metode lainnya yang menerima dua parameter int, menghitung total waktu, dan mengembalikan hasilnya.

- 3.3 Nama Kelas dan Constructor:
 - Kode pertama menggunakan kelas Manusia dan constructor yang sesuai (Manusia(String nama, String rambut)).
 - Kode kedua menggunakan kelas Manusia2 dengan constructor yang juga sesuai (Manusia2(String nama, String rambut)).
- Nama Metode:
 - Kode pertama menggunakan metode sukaNonton.
 - Kode kedua menggunakan metode sukaMakan.
- Parameter Metode:
 - Pada kode pertama, sukaNonton memiliki dua versi:
 - sukaNonton(String film) untuk mencetak judul film.
 - sukaNonton(int episode, int durasi) untuk menghitung total jam nonton berdasarkan episode dan durasi.
 - Pada kode kedua, sukaMakan juga memiliki dua versi:
 - sukaMakan(String mie) untuk mencetak jenis makanan.
 - sukaMakan(int bungkus, int minggu) untuk menghitung jumlah bungkus makanan yang dimakan per minggu.
- Fokus Aktivitas:
 - Kode pertama fokus pada **hobi menonton** (sukaNonton).
 - Kode kedua fokus pada **hobi makan** (sukaMakan).
- Output Teks:
 - Pada kode pertama, teks yang dicetak berhubungan dengan **film dan jam nonton**.
 - Pada kode kedua, teks yang dicetak berhubungan dengan **makanan dan jumlah bungkus yang dimakan**.

[Nomor Soal 3] Penyusunan Algoritma dan Kode Program

- 1) Rancang desain solusi atau algoritma
 - Membuat class
 - Mendeklarasi konstruktor
 - Mendeklarasi metthod utama

2) Tuliskan kode program dan luaran

(Codingan 3.1)

```
1 public class Manusia2 {
2     //deklarasi atribut Manusia dalam variabel
3     String nama, rambut;
4
5     //deklarasi constructor
6     public Manusia2(String nama, String rambut) {
7         System.out.println(" Nama saya : " + nama +
8             "\n Warna Rambut : " + rambut);
9     }
10
11     //deklarasi method
12     void sukaNonton (String film) {
13         System.out.println(" Hobi Menonton : " + film);
14     }
15
16     int sukaNonton (int episode, int durasi) {
17         return episode*durasi;
18     }
19
20     //deklarasi method utama
21     public static void main( String[] args) {
22         Manusia2 satu = new Manusia2("Putri", "hitam");
23         satu.sukaNonton("Drakor");
24         int jumlahJam = satu.sukaNonton(2, 2);
25         System.out.println("Jam nonton = " +jumlahJam + " jam");
26     }
27 }
28
29
```

(Luaran 3.1)

```
[Running] cd "d:\praktikumkoprog\" && javac Manusia2.java && java Manusia2
Nama saya : Putri
Warna Rambut : hitam
Hobi Menonton : Drakor
Jam nonton = 4 jam
```

➤ Luaran yang dikeluarkan sudah sesuai dengan perintah

(Codingan 3.2)

```
1 public class Manusia2 {
2     //deklarasi atribut Manusia dalam variabel
3     String nama, rambut;
4
5     //deklarasi constructor
6     public Manusia2(String nama, String rambut) {
7         System.out.println(" Nama saya : " + nama +
8             "\n Warna Rambut : " + rambut);
9     }
10
11     //deklarasi method
12     void sukaMakan (String mie) {
13         System.out.println(" Suka makan : " + mie);
14     }
15
16     int sukaMakan (int bungkus, int minggu) {
17         return bungkus*minggu;
18     }
19
20     //deklarasi method utama
21     public static void main( String[] args) {
22         Manusia2 satu = new Manusia2("Iqbal", "Pink");
23         satu.sukaMakan("Mie rendang");
24         int jumlahJam = satu.sukaMakan(2, 2);
25         System.out.println("Mie yang dimakan perminggu : " + jumlahJam + " Bungkus");
26     }
27 }
28
29
```

(Luaran 3.2)

```
[Running] cd "d:\praktikumkoprog\" && javac Manusia2.java && java Manusia2
Nama saya : Iqbal
Warna Rambut : Pink
Suka makan : Mie rendang
Mie yang dimakan perminggu : 4 Bungkus
[Done] exited with code=0 in 7.796 seconds
```

- Luaran yang dihasilkan sudah sesuai dengan perintah

[Nomor Soal 3] Kesimpulan

- 1) Analisa
 - a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
- Permasalahan yang dihadapi adalah bagaimana membuat sebuah kelas Java yang dapat menampilkan informasi tentang seseorang, seperti nama dan hobi mereka. Kode juga perlu menghitung total aktivitas berdasarkan parameter tertentu, jumlah episode yang ditonton atau jumlah bungkus makanan yang dimakan. Tantangan lainnya adalah menggunakan metode dengan parameter yang berbeda, sehingga konsep method overloading dapat diterapkan.

Nama & NPM	Topik:	Tanggal:
Iqbal Tri Agustian G1F024043	Unit 4 Extend	14 September 2024

[Nomor Soal 4] Identifikasi Masalah:

1) Uraikan permasalahan dan variabel

```

public class Ortu { // membuat kelas induk
    void sukaMenonton(String a) { // method induk spesifik
        System.out.println("Menonton " + a);
    }

    void sukaMembaca(String a) { // method induk umum bisa diubah anak
        System.out.println("Suka Baca " + a);
    }
}

public static void main(String [] args) {
    System.out.println("Sifat Orang Tua :");
    Ortu objekO = new Ortu(); // memanggil objek induk
    objekO.sukaMenonton("Berita"); // memanggil sifat spesifik induk
    objekO.sukaMembaca("Koran"); // memanggil method dengan variabel dapat diubah

    System.out.println("\n Sifat Anak :");
    Anak objekA = new Anak(); //memanggil objek anak
    objekA.sukaMenonton(9, "Film Drakor"); //memanggil sifat spesifik anak yang diturunkan induk
    objekA.sukaMembaca("Komik One Piece"); //memanggil method ke induk yang otomatis diturunkan tanpa deklarasi ulang di anak
}

class Anak extends Ortu {
    void sukaMenonton(int a, String b) {
        System.out.println("Menonton Jam " + a + " Malam " + b);
    }

    void sukaMenonton(String a) { // method Induk spesifik
        System.out.println("Menonton " + a);
    }

    void sukaMembaca(String a) { // method induk umum bisa diubah anak
        System.out.println("Suka Baca " + a);
    }
}

public static void main(String [] args) {
    System.out.println("Sifat Orang Tua :");
    Ortu objekO = new Ortu(); // memanggil objek Induk
    objekO.sukaMenonton("Berita"); // memanggil sifat spesifik induk
    objekO.sukaMembaca("Koran"); // memanggil method dengan variabel dapat diubah

    System.out.println("\n Sifat Anak :");
    Anak objekA = new Anak(); //memanggil objek anak
    objekA.sukaMenonton(9, "Film Drakor"); //memanggil sifat spesifik anak yang diturunkan induk
    objekA.sukaMembaca("Komik One Piece"); //memanggil method ke induk yang otomatis diturunkan tanpa deklarasi ulang di anak
}
}

```

Luaran 4:
 Sifat Orang Tua :
 Menonton Berita
 Suka Baca Koran

 Sifat Anak :
 Menonton Jam 9 Malam Film Drakor
 Suka Baca Komik One Piece

- Latihan 4:**
- 4.1. Evaluasi method yang dimiliki kelas Anak extends Ortu dengan method di kelas Ortu!
Apakah penulisan method ini sudah efisien?
 - 4.2. Setelah di running di JDoodle, catat waktu eksekusinya.
Rekomendasikan perbaikan penulisan kode method untuk dapat mengoptimalkan waktu eksekusi!

2) Rincikan sumber informasi yang relevan (buku / webpage)

Video Materi 2 tentang – <https://www.youtube.com/watch?v=6qULMlcV-eg>

[Nomor Soal 4] Analisis dan Argumentasi

1) Uraikan rancangan solusi yang diusulkan.

➤ 4.1 Method di Kelas Ortu:

- Kelas **Ortu** memiliki dua metode:
 - sukaMenonton(String a) yang mencetak aktivitas menonton.
 - sukaMembaca(String a) yang mencetak aktivitas membaca.

- Method di Kelas Anak:
 - Kelas **Anak** yang meng-**extends** kelas **Ortu** memiliki dua metode:
 - `sukaMenonton(int a, String b)` untuk menambahkan parameter tambahan (int a) yang spesifik untuk menonton di jam tertentu.
 - `sukaMenonton(String a)` menimpa (overrides) method dari induk, tapi sebenarnya hanya mengulang kode yang sudah ada di kelas **Ortu**.
 - `sukaMembaca(String a)` juga menimpa method dari induk tetapi hanya mencetak informasi yang berbeda (bukan fungsi baru).
- 4.2 Hilangkan metode yang tidak perlu di kelas Anak. Jika metode di kelas induk (Ortu) tidak perlu diubah, tidak perlu dilakukan overriding di kelas Anak. Ini akan menyingkat kode dan meningkatkan efisiensi.
- 2) Analisis solusi, kaitkan dengan permasalahan.
- Solusi yang diberikan menyelesaikan permasalahan inheritance dengan lebih efisien. Kelas **Anak** mewarisi metode dari kelas **Ortu** secara langsung, kecuali pada kasus di mana perubahan diperlukan (seperti pada metode `sukaMenonton` yang memiliki parameter tambahan). Hal ini menghindari pengulangan kode yang tidak perlu dan membuat program lebih mudah untuk dikembangkan serta dijalankan dengan lebih cepat.

[Nomor Soal 4] Penyusunan Algoritma dan Kode Program

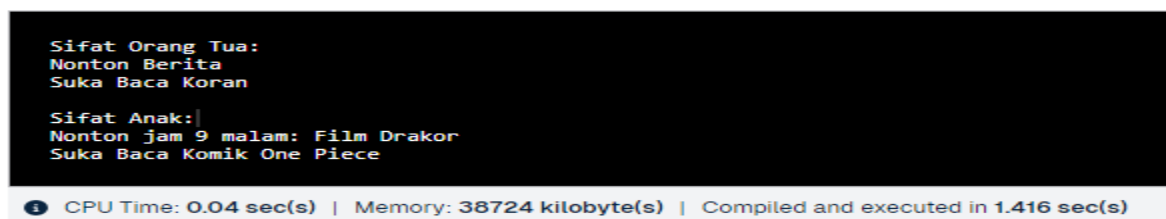
- 1) Rancang desain solusi atau algoritma
 - Membuat kelas
 - Mendeklarasi konstruktor
 - Mendeklarasi method utama

2) Tuliskan kode program dan luaran

(Codingan 4.2)

```
1 public class Ortu { // Membuat kelas induk
2     void sukaMenonton(String a) { // Method induk spesifik
3         System.out.println("Nonton " + a);
4     }
5
6     void sukaMembaca(String a) { // Method induk umum bisa diubah anak
7         System.out.println("Suka Baca " + a);
8     }
9
10    public static void main(String[] args) {
11        System.out.println("Sifat Orang Tua:");
12        Ortu objekO = new Ortu(); // Memanggil objek induk
13        objekO.sukaMenonton("Berita"); // Memanggil sifat spesifik induk
14        objekO.sukaMembaca("Koran"); // Memanggil method dengan variabel dapat diubah
15
16        System.out.println("\nSifat Anak:");
17        Anak objekA = new Anak(); // Memanggil objek anak
18        objekA.sukaMenonton(9, "Film Drakor"); // Memanggil sifat spesifik anak yang diturunkan induk
19        objekA.sukaMembaca("Komik One Piece"); // Memanggil method yang diubah oleh anak
20    }
21 }
22
23 class Anak extends Ortu { // Kelas Anak mewarisi dari Ortu
24     // Method overload dengan parameter tambahan (jam)
25     void sukaMenonton(int a, String b) {
26         System.out.println("Nonton jam " + a + " malam: " + b);
27     }
28
29     // Override method sukaMenonton dari kelas induk
30     @Override
31     void sukaMenonton(String a) {
32         System.out.println("Nonton " + a);
33     }
34
35     // Override method sukaMembaca dari kelas induk
36     @Override
37     void sukaMembaca(String a) {
38         System.out.println("Suka Baca " + a);
39     }
40 }
```

(Luaran codingan 4.2)



```
Sifat Orang Tua:
Nonton Berita
Suka Baca Koran

Sifat Anak:
Nonton jam 9 malam: Film Drakor
Suka Baca Komik One Piece
```

CPU Time: 0.04 sec(s) | Memory: 38724 kilobyte(s) | Compiled and executed in 1.416 sec(s)

- Luaran yang dihasilkan sudah sesuai dengan perintah

[Nomor Soal 4] Kesimpulan

- 1) Analisa
 - a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
 - Pewarisan memungkinkan pengelompokan logika dan fungsionalitas di satu tempat (kelas induk), kemudian kelas anak bisa memodifikasinya sesuai kebutuhan.
 - Dengan menggunakan overriding, program tetap mengikuti prinsip OOP (Object-Oriented Programming) di mana kelas anak dapat memperluas atau mengubah perilaku metode induk tanpa merusak kode aslinya.
 - Organisasi kode yang lebih baik (tanpa duplikasi) membantu menjaga kebersihan dan mempermudah pemeliharaan program di masa mendatang.

