

### Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
Lina Rosalinda G1F024063	Operator Java	6 September 2024

#### [Nomor 1] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variable  
Salin dan tempel kode program berikut ke Eclipse atau JDoodle

```
public class OperatorAritmatika{  
    public static void main(String[] args) {  
        // deklarasi nilai  
        int a = 20, b = 3;  
        //operator aritmatika  
        System.out.println("a: " +a);  
        System.out.println("b: " +b);  
        System.out.println("a + b = " + (a + b));  
    }  
}
```

#### Luaran:

```
a: 20  
b: 3  
a + b = 23
```

#### Latihan 1

Susun kode Java untuk perhitungan dengan ekspresi  $(2*3 + 6 / 2 - 4)$ . Simpulkan urutan prioritas operator yang dijalankan ekspresi tersebut!

Menghitung hasil matematika dari  $(2*3 + 6 / 2 - 4)$  menggunakan operator aritmatika dalam Java. Variabel bisa dapat dihitung langsung dalam pernyataan `System.out.println`.

Urutan proiritas operator yaitu:

1.  $2*3$  Menghitung hasil perkalian terlebih dahulu
2.  $6/2$  Menghitung hasil pembagian setelah perkalian
3. Hasil dari  $2*3$  dan hasil  $6/2$  kemudian dijumlahkan lalu dikurangi 4.

#### [Nomor 1] Analisis dan Argumentasi

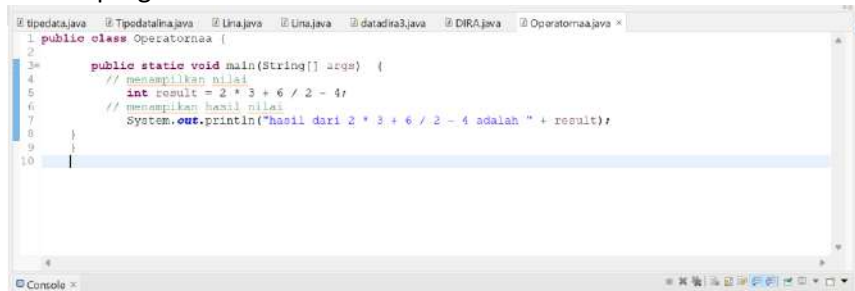
- 1) Uraikan rancangan solusi yang diusulkan.  
Saya mengusulkan permasalahan ini dapat diatasi dengan cara menghitung hasil dari ekspresi matematika dengan urutan yang diprioritaskan dan menampilkannya di Java. Alasan solusi ini karena menghitung dengan mengikuti prioritas operator memudahkan proses pemograman. Perbaikan kode program ini dengan cara menghitung hasil dari ekspresi matematika dengan urutan yang telah diprioritaskan.
- 2) Analisis solusi, kaitkan dengan permasalahan.  
Solusi yang saya usulkan yaitu menyelesaikan permasalahan dengan menghitung ekspresi matematika dan menampilkannya di Java. Karena solusi ini mengikuti prioritas operator maka solusi ini memenuhi tujuan perhitungan sesuai dengan yang diinginkan dengan akurat.

#### [Nomor 1] Penyusunan Algoritma dan Kode Program

1. Rancang desain solusi atau algoritma
  - a) Algoritma
  - b) Mulai
  - c) Buat kelas dengan nama 'Operatorna'
  - d) Buat metode 'main' dengan parameter 'public static void main (String[] args)'
  - e) Berikan komentar pada tampilan nilai yang ingin dihitung

- f) Input nilai yang ingin dihitung
- g) Berikan komentar pada tampilan hasil nilai yang ingin dihitung
- h) Tambahkan baris "System.out.println("") dan diisi dengan nilai yang ingin dihitung
- i) Akhiri dengan tanda kurung kurawal penutup
- j) Akhiri program

## 2. Kode program dan Luaran



```

1 public class Operator {
2
3     public static void main(String[] args) {
4         // menampilkan nilai
5         int result = 2 * 3 + 6 / 2 - 4;
6         // menampilkan hasil nilai
7         System.out.println("hasil dari 2 * 3 + 6 / 2 - 4 adalah " + result);
8     }
9 }
10

```

- a) Analisa luaran yang dihasilkan



```

<terminated> Operator [Java Application] C:\Users\user\p2\poo\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.12.v20240802-1510\jre\bin\javaw.exe (Se
hasil dari 2 * 3 + 6 / 2 - 4 adalah 5

```

Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

## [Nomor 1] Kesimpulan

### 1) Analisa

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!  
Program ini menghitung ekspresi operator sesuai dengan prioritas operator, dengan menggunakan Java maka mempermudah perhitungan dan menampilkan hasil sesuai yang diinginkan dengan akurat.
- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?  
Saya memilih mengikuti urutan operator yang benar karena solusi ini memastikan akurasi perhitungan yang sederhana dan langsung yang membuatnya mudah dipahami dan diterapkan. Dengan memilih mengikuti urutan operator maka tujuan perhitungan matematika yang diinginkan akan terpenuhi.

## Template Lembar Kerja Individu dan Kelompok

<b>Nama &amp; NPM</b>	<b>Topik:</b>	<b>Tanggal:</b>
Lina Rosalinda G1F024063	Operator Java	6 September 2024
<b>[Nomor 2] Identifikasi Masalah:</b>		
<p>2) Uraikan permasalahan dan variable</p> <p>Salin dan tempel kode program berikut ke Eclipse atau JDoodle</p> <pre>public class OperatorAritmatika{     public static void main(String[] args) {         // deklarasi nilai         int a = 20, b = 3;         //operator aritmatika         System.out.println("a: " +a);         System.out.println("b: " +b);         System.out.println("a + b = " + (a + b));     } }</pre> <p><b>Luaran:</b></p> <pre>a: 20 b: 3 a + b = 23</pre> <p><b>Latihan 2</b></p> <p>Susun kode untuk menampilkan perhitungan dengan operator ( -=, *=, /=, %=)! Simpulkan hasilnya!</p> <p>Setelah operator java melakukan <b>penugasan berupa</b> ( -=, *=, /=, %=) maka akan keluar kesimpulan perhitungan pada penugasan yang diperintahkan yaitu:</p> <ol style="list-style-type: none"> <li>1. Nilai akhir b adalah 3</li> <li>2. Denagn operasi yang mencakup penambahan, pengurangan, perkalian, pembagian, dan sisa bagi. Setelah selesai melakukan operasi terebut maka nilai b akan tetap Kembali menjadi 3b.</li> </ol>		
<b>[Nomor Soal] Analisis dan Argumentasi</b>		
<ol style="list-style-type: none"> <li>1. Uraikan rancangan solusi yang diusulkan. Saya mengusulkan permasalahan ini dapat diatasi dengan cara menentukan nilai awal untuk variabel, lalu menerapkan operator penugasan satu per satu dan muai melihat perubahan-perubahan nilai dari sebelumnya, Alasan saya memilih solusi ini karena kita dapat memahami bagaimana cara kerja operator penugasan.</li> <li>2. Analisis solusi, kaitkan dengan permasalahan. Solusi yang saya usulkan yaitu menyelesaikan permasalahan dengan memberikan cara yang sederhana untuk memahami operator penugasan. Selain itu, Langkah-langkah nya yang jelas memungkinkan pemahaman yang lebih jelas tentang bagaimana setiap operator mempengaruhi nilai variabel.</li> </ol>		
<b>[Nomor 1] Penyusunan Algoritma dan Kode Program</b>		
<ol style="list-style-type: none"> <li>1. Rancang desain solusi atau algoritma <ol style="list-style-type: none"> <li>a) Algoritma</li> <li>b) Mulai</li> <li>c) Buat kelas dengan nama 'Operatorpenugasan'</li> <li>d) Buat metode 'main' dengan parameter 'public static void main (String[] args)'</li> <li>e) Berikan komentar pada tampilan nilai yang ingin dideklarasikan</li> <li>f) Input nilai yang ingim dihitung</li> <li>g) Berikan komentar pada tampilan operator penugasan</li> </ol> </li> </ol>		

- h) Tambahkan baris "System.out.println(\"\") lalu diisi dengan nilai yang ingin dihitung dan lakukan berulang sampai memenuhi jumlah operator penugasan yang ingin dihitung
- i) Akhiri dengan tanda kurung kurawal penutup
- j) Akhiri program

## 2. Kode program dan Luaran



```

1 public class OperatorPenugasan {
2     public static void main(String[] args) {
3         // deklarasi nilai
4         int a = 20, b = 3;
5         // operasi penjumlahan
6         System.out.println("Penjumlahan : " + a + b);
7         // pengurangan
8         System.out.println("Pengurangan : " + a - b);
9         // perkalian
10        System.out.println("Perkalian : " + a * b);
11        // pembagian
12        System.out.println("Pembagian : " + a / b);
13        // sisa hasil
14        System.out.println("Sisa Hasil : " + a % b);
15    }
16 }

```

### b) Analisa luaran yang dihasilkan



```

Penjumlahan : 23
Pengurangan : 17
Perkalian : 60
Pembagian : 6
Sisa Hasil : 1

```

Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

## [Nomor 2] Kesimpulan

### 1. Analisa

- a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program! Program ini meminta Saya untuk memahami dan menerapkan operator penugasan berupa (-=, \*=, /=, %=) untuk mengetahui hasil akhir dari operator penugasan tersebut dengan algoritma yang tersusun mulai dari bagaimana mendeklarasikan variabel, penerapan operator penugasan yang memastikan penerapannya dilakukan dengan benar.
- b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini? Saya memilih keputusan ini karena kode program yang jelas menunjukkan bagaimana nilai dipengaruhi oleh operator penugasan dengan kode yang jelas memudahkan pemahaman sesuai yang diharapkan.

**Template Lembar Kerja Individu dan Kelompok**

<b>Nama &amp; NPM</b>	<b>Topik:</b>	<b>Tanggal:</b>
<b>Lina Rosalinda G1F024063</b>	<b>Operator Java</b>	<b>6 September 2024</b>

**[Nomor 3] Identifikasi Masalah:**

3) Uraikan permasalahan dan variable

Salin dan tempel kode program berikut ke Eclipse atau JDoodle

```
public class OperatorRelasional {  
    public static void main(String[] args) {  
        int nilaiA = 12;  
        int nilaiB = 4;  
        boolean hasil;  
  
        System.out.println(" A = " + nilaiA + "\n B = " + nilaiB);  
        // apakah A lebih besar dari B?  
        hasil = nilaiA > nilaiB;  
        System.out.println("\n Hasil A > B = "+ hasil);  
  
        // apakah A lebih kecil dari B?  
        hasil = nilaiA < nilaiB;  
        System.out.println("\n Hasil A < B = "+ hasil);  
  
        // apakah A lebih besar samadengan B?  
        hasil = nilaiA >= nilaiB;  
        System.out.println("\n Hasil A >= B = "+ hasil);  
  
        // apakah A lebih kecil samadengan B?  
        hasil = nilaiA <= nilaiB;  
        System.out.println("\n Hasil A <= B = "+ hasil);  
  
        // apakah nilai A sama dengan B?  
        hasil = nilaiA == nilaiB;  
        System.out.println("\n Hasil A == B = "+ hasil);  
  
        // apakah nilai A tidak samadengan B?  
        hasil = nilaiA != nilaiB;  
        System.out.println("\n Hasil A != B = "+ hasil);  
    }  
}
```

**Luaran:**

A = 12  
B = 4

Hasil A > B = true  
Hasil A < B = false  
Hasil A >= B = true  
Hasil A <= B = false  
Hasil A == B = false  
Hasil A != B = true

**Latihan 3**

Susunlah perintah kode dengan operator relasional (<, >, <=, >=, =, ==, !=) untuk nilai a dan b yang menghasilkan luaran TRUE!

**[Nomor Soal] Analisis dan Argumentasi**

1. Uraikan rancangan solusi yang diusulkan.

Saya mengusulkan permasalahan ini dapat diatasi dengan cara mendeklarasikan variabel tipe integer misalnya a dan b dengan nilai yang sesuai untuk menghasilkan true, lalu menginisialisasikan variabel tersebut dengan nilai yang ingin ditentukan

sesuai yang diinginkan. Kemudian menerapkan berbagai operator rasional yang telah dideklarasikan lalu mengujinya. Alasan saya memilih solusi ini karena solusi ini memastikan bahwa operator ini di uji dengan nilai variabel yang memungkinkan semua pernyataan menghasilkan true .

### 3. Analisis solusi, kaitkan dengan permasalahan.

Solusi yang saya usulkan yaitu menyelesaikan permasalahan dengan memberikan cara yang sederhana untuk memahami operator penugasan. Selain itu, Langkah-langkah nya yang jelas memungkinkan pemahaman yang lebih jelas tentang bagaimana setiap operator mempengaruhi nilai variabel.

## [Nomor 3] Penyusunan Algoritma dan Kode Program

### 1. Rancang desain solusi atau algoritma

- k) Algoritma
- l) Mulai
- m) Buat kelas dengan nama 'Operatorpenugasan'
- n) Buat metode 'main' dengan parameter 'public static void main (String[] args)'
- o) Berikan komentar pada tampilan nilai yang ingin dideklarasikan
- p) Input nilai yang ingin dihitung
- q) Berikan komentar pada tampilan operator penugasan
- r) Tambahkan baris "System.out.println("") lalu diisi dengan nilai yang ingin dihitung dan lakukan berulang sampai memenuhi jumlah operator penugasan yang ingin dihitung
- s) Akhiri dengan tanda kurung kurawal penutup
- t) Akhiri program

### 2.Kode program dan Luaran

### c) Analisa luaran yang dihasilkan

Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

<b>[Nomor 3] Kesimpulan</b>
<p>2. Analisa</p> <p>c) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program! Program ini meminta Saya untuk memahami dan menerapkan operator penugasan berupa (-=, *=, /=, %=) untuk mengetahui hasil akhir dari operator penugasan tersebut dengan algoritma yang tersusun mulai dari bagaimana mendeklarasikan variabel, penerapan operator penugasan yang memastikan penerapannya dilakukan dengan benar.</p> <p>d) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini? Saya memilih keputusan ini karena kode program yang jelas menunjukkan bagaimana nilai dipengaruhi oleh operator penugasan dengan kode yang jelas memudahkan pemahaman sesuai yang diharapkan.</p>

**Lembar Kerja Individu dan Kelompok**

Nama & NPM	Topik:	Tanggal:
Lina Rosalinda G1F024063	Operator Java	6 September 2024

**[Nomor 4] Identifikasi Masalah:**

Kedua operator ini digunakan pada operand bertipe bilangan bulat. Operator penaikan digunakan untuk menaikan nilai variabel sebesar satu, sedangkan operator penurunan dipakai untuk menurunkan nilai variabel sebesar satu.

**Contoh 4:** Salin dan tempel kode program berikut ke Eclipse atau JDoodle

```
public class operator {  
    public static void main(String[] args) {  
        // deklarasi nilai  
        int a = 5;  
  
        System.out.println("a: " + a);  
        System.out.println("b: " + (a++));  
    }  
}
```

**Luaran:**

a: 5  
b: 5

**Latihan 4.**

- 4.1. Susunlah kode program untuk menghasilkan luaran nilai a = 5 dan b = 6 dengan pre/post increment dan pre/post decrement.
- 4.2. Simpulkan hasil perbandingan Anda (pre/post increment, pre/post decrement)!

Jawab:

- 4.1 untuk kode program akan ditampilkan pada analisis dan luaran program
- 4.2 kesimpulannya dibagi menjadi beberapa yaitu:

Post-Increment (a++) yaitu Nilai b adalah nilai awal a, kemudian a bertambah.

Pre-Decrement (--a) yaitu a dikurangi terlebih dahulu, kemudian nilai a digunakan untuk b.

Pre-Increment (++a) yaitu a ditambah terlebih dahulu, kemudian nilai a digunakan untuk b.

Post-Decrement (a--) yaitu Nilai b adalah nilai awal a, kemudian a berkurang.

**[Nomor 4] Analisis dan Argumentasi**

- 1) Uraikan rancangan solusi yang diusulkan.

Saya mengusulkan permasalahan ini dapat diatasi dengan cara mendeklarasikan a dengan nilai 5 dan b untuk menyimpan hasil. Alasan solusi ini karena dengan dapat membedakan perbedaan dari pre dan post increment /decrement. .

- 2.) Analisis solusi, kaitkan dengan permasalahan.

Solusi yang saya usulkan yaitu melakukan pengamatan langsung terhadap bagaimana operator mempengaruhi nilai variabel karena dengan solusi ini memberikan pengalaman langsung terhadap pre dan post increment/decrement sesuai yang diinginkan.

**[Nomor 1] Penyusunan Algoritma dan Kode Program**

1. Rancang desain solusi atau algoritma
  1. Algoritma
  - 4) Mulai



- 5) Buat kelas dengan nama 'OperatorPenugasan'
- 6) Buat metode 'main' dengan parameter 'public static void main(String[] args)'
- 7) Berikan komentar pada tampilan nilai yang ingin dideklarasikan
- 8) Komentar: Deklarasikan nilai awal variabel untuk perhitungan.
- 9) Input nilai yang ingin dihitung
- 10) Deklarasikan variabel dan inisialisasi dengan nilai awal.
- 11) Berikan komentar pada tampilan operator penugasan
- 12) Komentar: Jelaskan operasi increment dan decrement yang akan diterapkan.
- 13) Tambahkan baris "System.out.println("")" lalu diisi dengan nilai yang ingin dihitung dan lakukan berulang sampai memenuhi jumlah operator penugasan yang ingin dihitung
- 14) Implementasikan operasi post-increment, pre-increment, post-decrement, dan pre-decrement satu per satu.
- 15) Tampilkan hasil setiap operasi dengan System.out.println().
- 16) Akhiri dengan tanda kurung kurawal penutup
- 17) Akhiri program

## 2. Kode program dan Luaran

```

1 package classoperator;
2 public class OperatorPenugasan {
3     public static void main(String[] args) {
4         // Deklarasi
5         int a = 10;
6         System.out.println("Post Increment");
7         System.out.println("Ini variabel a: " + a);
8         System.out.println("Ini variabel a: " + ++a);
9         System.out.println("Ini variabel a: " + a);
10        System.out.println();
11
12        // Pre Increment
13        int b = 10;
14        System.out.println("Pre Increment");
15        System.out.println("Ini variabel b: " + ++b);
16        System.out.println("Ini variabel b: " + b);
17        System.out.println("Ini variabel b: " + ++b);
18        System.out.println();
19
20        // Post Decrement
21        int c = 10;
22        System.out.println("Post Decrement");
23        System.out.println("Ini variabel c: " + c);
24        System.out.println("Ini variabel c: " + --c);
25        System.out.println("Ini variabel c: " + c);
26        System.out.println();
27
28        // Pre Decrement
29        int d = 10;
30        System.out.println("Pre Decrement");
31        System.out.println("Ini variabel d: " + --d);
32        System.out.println("Ini variabel d: " + d);
33        System.out.println("Ini variabel d: " + --d);
34        System.out.println();
35    }
36 }

```

### d) Analisa luaran yang dihasilkan

```

C:\Users\user> java OperatorPenugasan
Post Increment
Ini variabel a: 10
Ini variabel a: 11
Ini variabel a: 11

Pre Increment
Ini variabel b: 11
Ini variabel b: 12
Ini variabel b: 13

Post Decrement
Ini variabel c: 10
Ini variabel c: 9
Ini variabel c: 9

Pre Decrement
Ini variabel d: 9
Ini variabel d: 9
Ini variabel d: 8

```

Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan.

Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

<b>[Nomor 4] Kesimpulan</b>
-----------------------------

2) Analisa
------------

- |   |
|---|
| <ul style="list-style-type: none"><li>c) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!<br/>Program ini menugaskan bahwa dengan menerapkan operator secara langsung dalam kode maka kita bisa melihat secara nyata. Algoritma yang disusun pun mengarahkan untuk mendeklarasikan variabel dengan berbagai jenis operator. Kode yang disediakan pun sesuai dengan ekspektasi dan luaran yang sesuai yang diinginkan yang diinginkan dengan akurat.</li><li>d) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?<br/>Saya memilih menggunakan contoh kode langsung agar bisa membedakan berbagai jenis variabel tentang bagaimana operator mempengaruhi nilai-nilai variabel. Dengan menerapkan operator secara langsung dalam kode membuat kita bisa melihat hasil kerja secara nyata.</li></ul> |
|---|

## Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
Lina Rosalinda G1F024063	Operator Java	6 September 2024

### [Nomor 5] Identifikasi Masalah:

Salin dan tempel kode program berikut ke Eclipse atau JDoodle

```
public class operator {  
    public static void main(String[] args) {  
        // deklarasi nilai  
        boolean a = true;  
        boolean b = false;  
  
        System.out.println("Hasil logika (a && b) : " + (a && b));  
    }  
}
```

### Luaran:

Hasil logika (a && b) : false

### Latihan 5

Susun kode program dengan mengubah nilai **a** dan **b** untuk menghasilkan luaran **true** dengan operator **&&** dan operator **||**. Beri kesimpulan!

Jawab:

Untuk kode program saya akan menampilkan di abalisa dan luaran kode program

Keismpulannya adalah dengan menggunakan operator logika **&&** dan **||**, kita dapat mengontrol hasil logika boolean. Operator **&&** memerlukan kedua operand **true** untuk menghasilkan **true**, sedangkan operator **||** memerlukan minimal satu operand **true**.

### [Nomor 5] Analisis dan Argumentasi

1. Uraikan rancangan solusi yang diusulkan.

Saya mengusulkan permasalahan ini dapat diatasi dengan cara mendeklarasikan dua variabel Boolean **a** dan **b** lalu menginisialisasikan **a** dan **b** dengan nilai yang sesuai untuk menghasilkan **true**. Alasan solusi ini karena dengan mengubah nilai variabel **a** dan **b** dalam kode program untuk menghasilkan luaran **true** menggunakan operator logika **&&** (AND) dan **||** (OR) akan menghasilkan luaran sesuai yang diinginkan.

2. Analisis solusi, kaitkan dengan permasalahan.

Solusi yang saya usulkan yaitu menyelesaikan permasalahan dengan menghasilkan hasil **true** menggunakan operator logika **&&** dan **||** dengan nilai variabel Boolean. Operator **&&**: Memerlukan kedua operand **true** untuk hasil **true**. Operator **||**: Memerlukan minimal satu operand **true** untuk hasil **true**.: Solusi ini menunjukkan cara yang efektif untuk menghasilkan **true** dengan kedua operator logika, sesuai dengan kebutuhan dan aturan operator boolean.

### [Nomor 5] Penyusunan Algoritma dan Kode Program

1. Rancang desain solusi atau algoritma

- 1) Mulai
- 2) Buat kelas dengan nama **operatorlogika**
- 3) Menyusun kode dalam kelas ini untuk menggunakan operator logika.
- 4) Buat metode **main** dengan parameter **public static void main(String[] args)**
- 5) Deklarasikan variabel boolean **a** dan **b** misalnya, **boolean a = true;** dan **boolean b = false;**
- 6) Tampilkan nilai awal dari variabel **a** dan **b**

- 7) Gunakan System.out.println untuk menampilkan nilai-nilai ini.
- 8) Berikan komentar pada setiap tampilan hasil operasi logika. misalnya, komentar tentang operasi && dan || untuk memudahkan pemahaman.
- 9) Tambahkan baris System.out.println untuk hasil logika dengan operator && dan || misalnya:
- 10) System.out.println("Hasil logika (a && b) : " + (a && b));
- 11) System.out.println("Hasil logika (a || b) : " + (a || b));
- 12) Akhiri dengan tanda kurung kurawal penutup
- 13) Untuk menutup metode main.
- 14) Akhiri program

## 2. Kode program dan Luaran

```

package operatorPrecedence;

public class operatorPrecedence {

    public static void main(String[] args) {
        // Mendefinisikan dan menginisialisasi variabel
        int a = 12;
        int b = 4;

        // Tampilkan hasil awal
        System.out.println("A = " + a + " dan B = " + b);

        // Operator eksponensial yang menghasilkan 2048
        hasil = a > b; // A lebih besar dari B
        System.out.println("Hasil A > B = " + hasil); // true

        hasil = a < b; // A lebih besar atau sama dengan B
        System.out.println("Hasil A < B = " + hasil); // false

        hasil = a <= b; // A lebih besar atau sama dengan B
        System.out.println("Hasil A <= B = " + hasil); // true

        hasil = a >= b; // A lebih besar atau sama dengan B
        System.out.println("Hasil A >= B = " + hasil); // true

        hasil = a & b; // A lebih besar atau sama dengan B
        System.out.println("Hasil A & B = " + hasil); // true

        hasil = a | b; // A lebih besar atau sama dengan B
        System.out.println("Hasil A | B = " + hasil); // true

        hasil = a ^ b; // A lebih besar atau sama dengan B
        System.out.println("Hasil A ^ B = " + hasil); // true

        hasil = ~a; // A lebih besar atau sama dengan B
        System.out.println("Hasil ~A = " + hasil); // false
    }
}

```

### e) Analisa luaran yang dihasilkan

```

A = 12
B = 4
Hasil A > B = true
Hasil A < B = false
Hasil A <= B = true
Hasil A >= B = true
Hasil A & B = true
Hasil A || B = true
Hasil A & B = true
Hasil A | B = true
Hasil A ^ B = true
Hasil ~A = false

```

Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

## [Nomor 5] Kesimpulan

### 3) Analisa

- e) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program! Untuk menghasilkan true dengan operator logika: && memerlukan kedua operand true. Misalnya, a = true dan b = true menghasilkan true. || memerlukan minimal satu operand true. Misalnya, a = false dan b = true menghasilkan true.
- f) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini? Saya memilih solusi ini memastikan kita memahami cara kerja operator logika dengan menguji nilai variabel dan menghasilkan hasil yang diinginkan, yaitu true untuk kombinasi yang sesuai

**Lembar Kerja Individu dan Kelompok**

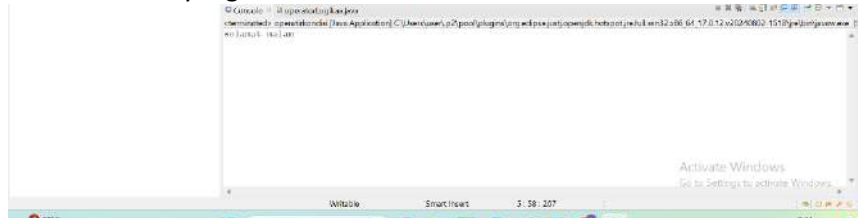
<b>Nama &amp; NPM</b>	<b>Topik:</b>	<b>Tanggal:</b>
Lina Rosalinda G1F024063	Operator Java	6 September 2024
<b>[Nomor 6] Identifikasi Masalah:</b>		
<b>Bentuk umum: (ekspresi) ? (jika benar) : (jika salah);</b>  Contoh 6: <pre>public class OperatorKondisi{     public static void main( String[] args ){         String status = "";         int nilai = 80;         status = (nilai &gt; 60)?"Lulus":"Gagal";         System.out.println( status );     } }</pre> <b>Luaran:</b> Lulus  <b>Latihan 6</b> Susun kode program! Dengan informasi berikut: Diketahui nama variabel Jam = 12 Apabila jam < 12 maka tampil "Selamat Pagi", apabila jam > 12 maka tampil "Selamat Malam".		
<b>[Nomor 6] Analisis dan Argumentasi</b>		
<p>1). Uraikan rancangan solusi yang diusulkan. Saya mengusulkan permasalahan ini dapat diatasi dengan cara mendeklarasikan variabel jam dan menggunakan operator kondisi untuk menentukan pesan yang sesuai menghitung hasil dari ekspresi matematika dengan urutan yang diprioritaskan dan menampilkannya di Java. Alasan solusi ini karena operator kondisi memudahkan penulisan kode dengan ringkas untuk mengambil keputusan yang berbasis kondisi sehingga mudah dibaca</p> <p>2). Analisis solusi, kaitkan dengan permasalahan. Solusi yang saya usulkan yaitu menggunakan operator kondisi memudahkan pengambilan keputusan yang membuatnya menjadi lebih simple, karena secara efektif menetapkan hasil berdasarkan variabel.</p>		
<b>[Nomor 6] Penyusunan Algoritma dan Kode Program</b>		
<p>1. Rancang desain solusi atau algoritma</p> <p>1. Algoritma</p> <ol style="list-style-type: none"><li>1) Mulai</li><li>2) Buat kelas dengan nama operatorKondisi</li><li>3) buat metode main dengan parameter public static void main (String [ ] args)</li><li>4) Deklarasikan variabel jam dengan nilai 12</li><li>5) Gunakan operator kondisi untuk menentukan pesan berdasarkan jam</li><li>6) Tampilkan hasil menggunakan System.out.println</li><li>7) Akhiri dengan kurung kurawal</li><li>8) Akhiri program</li></ol> <p>2. Kode program dan Luaran</p>		

```

1 public class operasiKondisi {
2     public static void main( String[] args ) {
3         String status = "";
4         int jam = 12;
5         status = (jam > 12) ? "malam pagi" : "petang malam";
6         System.out.println( status );
7     }
8 }

```

f) Analisa luaran yang dihasilkan



Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

**[Nomor 6] Kesimpulan**

4) Analisa

1.Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!

Program ini menentukan pesan berdasarkan nilai variabel jam. Ini memenuhi tujuan dengan cara yang efisien dan sesuai dngan apa yag ingin diinginkan.

2.Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Saya memilih mnentukan pesan berdasarkan nilai variabel jam karena langsung memberikan hasil berdasarkan kondisi serta mempermudah pembacaan dan pemahaman kode.

## Lembar Kerja Individu dan Kelompok

Nama & NPM	Topik:	Tanggal:
Lina Rosalinda G1F024063	Operator Java	6 September 2024

### [Nomor 7] Identifikasi Masalah:

Salin dan tempel kode program berikut ke Eclipse atau JDoodle

```
public class operatorBitwise {  
    public static void main(String[] args) {  
        int a = 10;  
        int b = 7;  
        int hasil;  
  
        hasil = a & b;  
        System.out.println("Hasil dari a & b : " + hasil );  
  
        hasil = a | b;  
        System.out.println("Hasil dari a | b : " + hasil );  
  
        hasil = a ^ b;  
        System.out.println("Hasil dari a ^ b : " + hasil );  
    }  
}
```

#### Luaran:

```
Hasil dari a & b : 6  
Hasil dari a | b : 7  
Hasil dari a ^ b : 1
```

#### Latihan 7

Susun kode tambahan dari Contoh 7 untuk melakukan perhitungan dengan operator ( $\gg$ ,  $\ll$ ).  
Hubungkan hasil luaran dengan perhitungan manual bilangan biner!

Jawab:

Untuk menambahkan perhitungan dengan operator bitwise pergeseran ( $\gg$  dan  $\ll$ ) pada kode program yang sudah ada, ikuti langkah-langkah berikut: Operator  $\ll$  (left shift): Menggeser bit ke kiri. Setiap pergeseran ke kiri setara dengan perkalian dengan 2. Operator  $\gg$  (right shift): Menggeser bit ke kanan. Setiap pergeseran ke kanan setara dengan pembagian dengan 2.

### [Nomor 7] Analisis dan Argumentasi

1. Uraikan rancangan solusi yang diusulkan.

Saya mengusulkan permasalahan ini dapat diatasi dengan cara menambahkan operator bitwise pergeseran ( $\gg$  dan  $\ll$ ) ke dalam kode bitwise yang sudah ada untuk memverifikasi hasilnya dengan perhitungan manual bilangan biner.

2. Analisis solusi, kaitkan dengan permasalahan.

Solusi yang saya usulkan yaitu Menambahkan operasi  $\ll$  (left shift) dan  $\gg$  (right shift) ke kode bitwise yang ada.

$a \ll 1$  menggeser bit ke kiri, setara dengan perkalian dengan 2.

$a \gg 1$  menggeser bit ke kanan, setara dengan pembagian dengan 2.

Kaitannya:

Operator Bitwise Dasar (&, |, ^) menunjukkan operasi logika bitwise.

Operator Pergeseran (<<, >>) menunjukkan manipulasi posisi bit yang memperluas pemahaman tentang perubahan nilai biner.

### [Nomor 7] Penyusunan Algoritma dan Kode Program

### 1. Rancang desain solusi atau algoritma

- 1.Algoritma
- 1.Mulai
- 2.Buat kelas dengan nama 'operatorbitwise'
- 3.Buat metode 'main' dengan parameter 'public static void main (String[] args)'
- 4.Berikan komentar pada setiap bitwise yang ingin ditampilkan
- 5.Input nilai yang ingim dihitung
- 6.Berikan komentar pada tampilan hasil nilai operasi bitwise yang ingin dihitung
- 7.Tambahkan baris "System.out.println("") dan diisi dengan nilai yang ingin dihitung
- 8.Akhiri dengan tanda kurung kurawal penutup
- 9.Akhiri program

## 2.Kode program dan Luaran

The screenshot shows an IDE with two Java files open. The first file, `Operasional.java`, contains the following code:

```

package operasional;

public class Operasional {
    public static void main(String[] args) {
        int a = 10;
        int b = 1;
        int hasil;

        hasil = a + b;
        System.out.println("Hasil dari a + b = " + hasil);

        hasil = a * b;
        System.out.println("Hasil dari a * b = " + hasil);

        hasil = a - b;
        System.out.println("Hasil dari a - b = " + hasil);

        hasil = a / b;
        System.out.println("Hasil dari a / b = " + hasil);
    }
}

```

The second file, `OperasionalKondisi.java`, contains the following code:

```

package operasional;

public class OperasionalKondisi {
    public static void main(String[] args) {
        int a = 10;
        int b = 1;
        int hasil;

        hasil = a + b;
        System.out.println("Hasil dari a + b = " + hasil);

        hasil = a * b;
        System.out.println("Hasil dari a * b = " + hasil);

        hasil = a - b;
        System.out.println("Hasil dari a - b = " + hasil);

        hasil = a / b;
        System.out.println("Hasil dari a / b = " + hasil);

        hasil = a * b <= 1 ? * : * + hasil;
        System.out.println("Hasil dari a * b <= 1 : * + hasil");
    }
}

```

g) Analisa luaran yang dihasilkan

[illegible]

Luaran sudah sesuai dengan program yang disusun karena hasil yang dihasilkan oleh program telah sesuai dengan apa yang diharapkan. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data sesuai dengan spesifikasi yang diminta.

### [Nomor 7] Kesimpulan

## 5) Analisa

- g) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!



Pada program ini masalah yang dihadapi adalah menambahkan perhitungan dengan operator bitwise pergeseran (>> dan <<) ke dalam kode yang sudah ada, dan memverifikasi hasilnya dengan perhitungan manual bilangan biner. Operator pergeseran bitwise ini memanipulasi posisi bit dalam bilangan biner, yang dapat memberikan pemahaman lebih mendalam tentang operasi bitwise. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Saya memilih menambahkan perhitungan dengan operator bitwise pergeseran (>> dan <<) ke dalam kode yang sudah ada karena alasan utama adalah menambahkan operasi bitwise pergeseran (>> dan <<) ke dalam kode program yang sudah ada dan memverifikasi hasilnya dengan perhitungan manual bilangan biner. mengikuti urutan operator yang benar karena solusi ini memastikan akurasi perhitungan yang sederhana dan langsung yang membuatnya mudah dipahami dan diterapkan.

## **Refleksi**

Pengalaman saya selama belajar yaitu mendapatkan pengetahuan baru tentang berbagai jenis operator yaitu operator relasional, operator aritmatika, operator penugasan (pre/post increment, pre/post decrement), logika boolean, operator kondisi dan operator bitwise.

Tantangan yang saya hadapi ketika mengerjakan ini yaitu mulai dari kesulitan memahami materi karena belum terlalu memahami tentang materi ini tapi dengan meminta bantuan kepada teman-teman maka membantu saya selama minggu ini.

Pada materi ini saya mempelajari tentang bagaimana menghitung aritmatika dengan ekspresi, mengerjakan penugasan aritmatika, mendeklarasikan variabel dengan nilai yang sesuai untuk menghasilkan true/false, mencari nilai dengan pre/post increment dan pre/post decrement, menggunakan operator logika && dan | |, menentukan operator kondisi dan operator bitwise