

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Aritmatika	17 September 2024
[Nomor Soal] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variable</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>Luaran: a: 20 b: 3 a - b = 17</p> <ol style="list-style-type: none"> 1. Tambahkan baris <code>System.out.println("a + b = " + (a + b));</code>; Ubahlah operator (+) dengan tanda (-, *, /, %) 2. Analisa perhitungan matematika yang terjadi! <p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <ol style="list-style-type: none"> 1. Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMylw Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman 2. https://www.youtube.com/watch?v=PzCMZOexZM Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman https://www.youtube.com/watch?v=LcFgl0yrKEw 		
[Nomor Soal] Analisis dan Argumentasi		
<p>1) Uraikan rancangan solusi yang diusulkan.</p> <ol style="list-style-type: none"> 1. Buat kelas operator dan metode utama 2. Deklarasi variable bertipe int 3. Tampilkan nilai variabel gunakan <code>System.out.println()</code> untuk menampilkan nilai variabel a dan b. 4. Operasi aritmatika dilakukan operasi penjumlahan dengan variabel a dan b. 5. Kesalahan dalam Kode pada kode yang diberikan, terdapat kesalahan: program melakukan pengurangan (a - b) tetapi ditampilkan sebagai penjumlahan (a + b). 6. Perbaiki dan gantilah operator - dengan + jika yang dimaksud adalah operasi penjumlahan. <p>2) Analisis solusi, kaitkan dengan permasalahan.</p> <ol style="list-style-type: none"> 1. Masalah program dimaksudkan untuk menghitung penjumlahan antara a = 20 dan b = 3. 2. Kesalahan alih-alih menjumlahkan, program malah mengurangkan a - b, sehingga hasilnya salah. 		

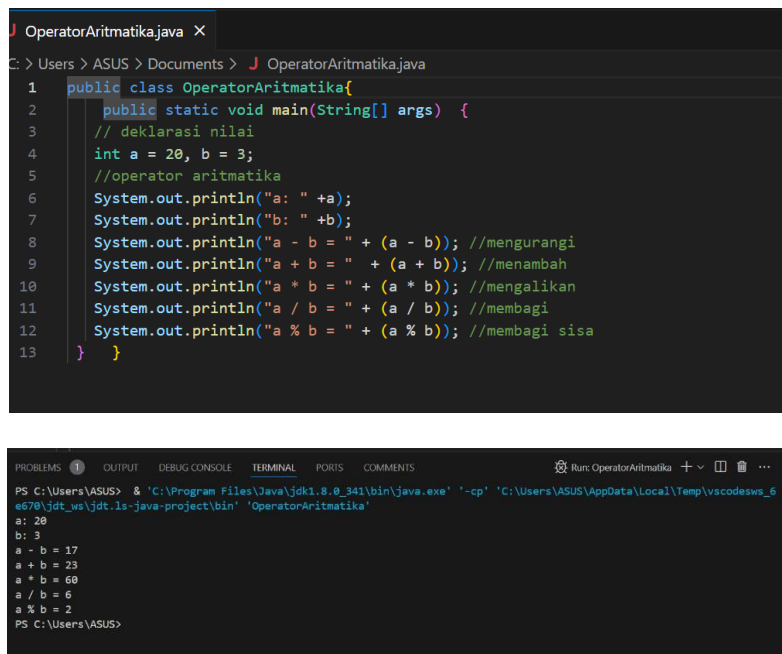
3. Solusi yang diberikan program sudah benar dalam deklarasi variabel dan menampilkan hasil, tapi operator yang digunakan adalah pengurangan (-), bukan penjumlahan (+).
4. Perbaikan ganti operator - dengan + untuk mendapatkan hasil penjumlahan yang benar.
5. Hasil akhir setelah perbaikan, program akan menampilkan hasil $a + b = 23$, sesuai dengan tujuan awal.

[Nomor Soal] Penyusunan Algoritma dan Kode Program

1) Rancang desain solusi atau algoritma

1. Salin potongan kode
2. Masuk ke VSCode dan tempel potongan kode
3. Tambahkan perintah sesuai yang diminta soal
4. Run kode dan pastikan tidak ada error
5. Selesai

2) Tuliskan kode program dan luaran



```
OperatorAritmatika.java X
C:\Users\ASUS\Documents > J OperatorAritmatika.java
1 public class OperatorAritmatika{
2     public static void main(String[] args) {
3         // deklarasi nilai
4         int a = 20, b = 3;
5         //operator aritmatika
6         System.out.println("a: " + a);
7         System.out.println("b: " + b);
8         System.out.println("a - b = " + (a - b)); //mengurangi
9         System.out.println("a + b = " + (a + b)); //menambah
10        System.out.println("a * b = " + (a * b)); //mengalikan
11        System.out.println("a / b = " + (a / b)); //membagi
12        System.out.println("a % b = " + (a % b)); //membagi sisa
13    }
}

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\ASUS> & "C:\Program Files\Java\jdk1.8.0_341\bin\java.exe" "-cp" "C:\Users\ASUS\AppData\Local\Temp\vscodesws_6
e670\jdt_ws\jdt.ls-java-project\bin" "OperatorAritmatika"
a: 20
b: 3
a - b = 17
a + b = 23
a * b = 60
a / b = 6
a % b = 2
PS C:\Users\ASUS>
```

[Nomor Soal] Kesimpulan

1) Analisa

Kesimpulannya, program ini awalnya ditujukan untuk menghitung penjumlahan dua bilangan, namun terjadi kesalahan karena menggunakan operator pengurangan. Meskipun struktur program, deklarasi variabel, dan cara menampilkan hasil sudah benar, operator yang salah menyebabkan hasilnya tidak sesuai. Dengan memperbaiki operator dari - menjadi +, program akan berfungsi sesuai tujuan dan memberikan hasil penjumlahan yang benar, yaitu $a + b = 23$.

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Penugasan	17 September 2024
[Nomor Soal] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variable</p> <pre> public class OperatorPenugasan { public static void main(String[] args) { // deklarasi nilai int a = 20, b = 3; //operator penugasan b += a; System.out.println("Penambahan : " + b); // pengurangan b -= a; System.out.println("Pengurangan : " + b); // perkalian b *= a; System.out.println("Perkalian : " + b); // Pembagian b /= a; System.out.println("Pembagian : " + b); // Sisa bagi b %= a; // sekarang b=0 System.out.println("Sisa Bagi: " + b); } } </pre> <p>Luaran: Penambahan : 23 Pengurangan : 3 Perkalian : 60 Pembagian : 3 Sisa Bagi: 3</p> <p>Latihan 2. 2.1. Bandingkan hasil Contoh 1 dengan Contoh 2!</p> <p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <p>3. Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMylw Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman</p> <p>4. https://www.youtube.com/watch?v=PzCMZOboxZM Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman https://www.youtube.com/watch?v=LcFglOyrKEw</p>		
[Nomor Soal] Analisis dan Argumentasi		
<p>3) Uraikan rancangan solusi yang diusulkan.</p> <ol style="list-style-type: none"> 1. Buat kelas operator penugasan dan metode: 2. Deklarasi Variabel dengan mendefinisikan dua angka, a = 20 dan b = 3. 3. Penambahan tambahkan a ke b menggunakan b += a (hasil: b = 23). 		

4. Pengurangan kurangi a dari b menggunakan $b -= a$ (hasil: $b = 3$).
5. Perkalian dengan mengalikan b dengan a menggunakan $b *= a$ (hasil: $b = 60$).
6. Pembagian dengan bagi b dengan a menggunakan $b /= a$ (hasil: $b = 3$).
7. Sisa bagi dihitung sisa pembagian b dengan a menggunakan $b \% = a$ (hasil: $b = 3$).

4) Analisis solusi, kaitkan dengan permasalahan.

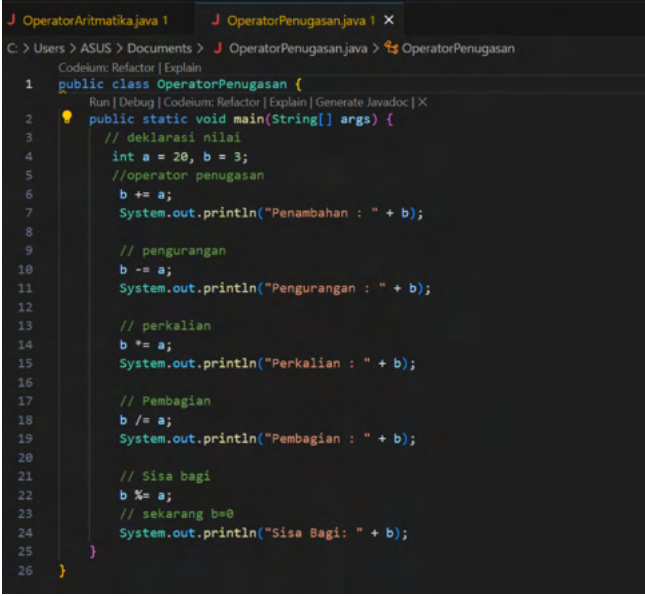
1. Deklarasi Variabel: Dua variabel (a dan b) digunakan sebagai input dalam operasi. Ini sesuai dengan permasalahan karena a dan b menjadi basis untuk operasi aritmatika.
2. Operator Penugasan:
 - Penambahan ($+=$): Nilai b ditambah dengan a, menghasilkan $b = 23$. Ini sesuai karena operasi penambahan dilakukan dengan benar.
 - Pengurangan ($-=$): Setelah penambahan, nilai b dikurangi dengan a, menghasilkan kembali $b = 3$. Pengurangan dilakukan sesuai dengan permasalahan.
 - Perkalian ($*=$): Setelah pengurangan, nilai b dikalikan dengan a, menghasilkan $b = 60$. Perkalian juga tepat sesuai operasi yang diharapkan.
 - Pembagian ($/=$): Setelah perkalian, nilai b dibagi dengan a, menghasilkan $b = 3$. Pembagian dilakukan dengan benar.
 - Sisa bagi ($\% =$): Setelah pembagian, sisa bagi dihitung antara b dan a, menghasilkan $b = 3$. Ini adalah langkah terakhir yang sesuai dengan permasalahan.

[Nomor Soal] Penyusunan Algoritma dan Kode Program

3) Rancang desain solusi atau algoritma

1. Salin kode dan tempel pada VSCode
2. Lakukan operasi yang diperintahkan
3. Bandingkan operasi aritmatika dengan operasi penugasan pada luarannya
4. Selesai

4) Tuliskan kode program dan luaran



```

J OperatorAritmatika.java 1  J OperatorPenugasan.java 1 X
C:\Users\ASUS\Documents> J OperatorPenugasan.java > OperatorPenugasan
Codeium: Refactor | Explain
Run | Debug | Codeium: Refactor | Explain | Generate Javadoc | X
1 public class OperatorPenugasan {
2     public static void main(String[] args) {
3         // deklarasi nilai
4         int a = 20, b = 3;
5         //operator penugasan
6         b += a;
7         System.out.println("Penambahan : " + b);
8
9         // pengurangan
10        b -= a;
11        System.out.println("Pengurangan : " + b);
12
13        // perkalian
14        b *= a;
15        System.out.println("Perkalian : " + b);
16
17        // Pembagian
18        b /= a;
19        System.out.println("Pembagian : " + b);
20
21        // Sisa bagi
22        b %= a;
23        // sekarang b=0
24        System.out.println("Sisa Bagi: " + b);
25    }
26 }
  
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\ASUS> & "C:\Program Files\Java\jdk1.8.0_341\bin\java.exe" "-cp" "C:\Users\ASUS\AppData\Local\Temp\vscodesna_6ed7915dt_m\jdt.ls-java-project\bin" "OperatorPenugasan"
Penambahan : 23
Pengurangan : 3
Perkalian : 60
Pembagian : 3
Sisa Bagi: 3
PS C:\Users\ASUS>
```

[Nomor Soal] Kesimpulan

1) Analisa

Kesimpulannya, program ini dengan efektif menyelesaikan permasalahan penggunaan operator penugasan pada variabel dengan mengikuti algoritma yang tepat. Dimulai dengan mendeklarasikan dua variabel, a dan b, program menggunakan operator penugasan (+, -, *, /, %) untuk memodifikasi nilai b berdasarkan operasi aritmatika terhadap a. Setiap langkah, dari penambahan hingga sisa bagi, dilakukan sesuai dengan harapan dan hasilnya ditampilkan dengan benar. Dengan demikian, kode program berhasil menerapkan dan menunjukkan penggunaan operator penugasan secara tepat dan sesuai dengan tujuan yang ditetapkan.

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Relasional	17 September 2024

[Nomor Soal] Identifikasi Masalah:

1) Uraikan permasalahan dan variable

```
public class OperatorRealasional {
    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("Hasil A > B = "+ hasil);
        // apakah A lebih kecil dari B?
        hasil = nilaiA < nilaiB;
        System.out.println("Hasil A < B = "+ hasil);
        // apakah A lebih besar samadengan B?
        hasil = nilaiA >= nilaiB;
        System.out.println("Hasil A >= B = "+ hasil);
        // apakah A lebih kecil samadengan B?
        hasil = nilaiA <= nilaiB;
        System.out.println("Hasil A <= B = "+ hasil);
        // apakah nilai A sama dengan B?
        hasil = nilaiA == nilaiB;
        System.out.println("Hasil A == B = "+ hasil);
        // apakah nilai A tidak samadengan B?
        hasil = nilaiA != nilaiB;
        System.out.println("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

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

- Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia
<https://www.youtube.com/channel/UC8B9rghd3dBIS6OKonLMylw>
[Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman](#)
- <https://www.youtube.com/watch?v=PzCMZOboxZM>
[Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman](#) <https://www.youtube.com/watch?v=LcFgl0yrKEw>

[Nomor Soal] Analisis dan Argumentasi

- 1) Uraikan rancangan solusi yang diusulkan.
 1. Buat kelas operator relasional dan metode main
 2. Deklarasikan variabel
 3. Tampilkan nilai A dan nilai B
 4. Lakukan perbandingan
 5. Tampilkan hasil
- 2) Analisis solusi, kaitkan dengan permasalahan.
 1. Deklarasi Variabel: Program sudah mendeklarasikan dua variabel int (nilaiA dan nilaiB) serta variabel boolean untuk hasil perbandingan.
 2. Tampilkan Nilai: Program menampilkan nilai awal dari nilaiA dan nilaiB, memberikan konteks sebelum perbandingan dilakukan.
 3. Perbandingan Relasional: Program melakukan perbandingan menggunakan berbagai operator (>, <, >=, <=, ==, !=), menyimpan hasil perbandingan dalam variabel boolean, dan menampilkan hasilnya.
 4. Output Hasil: Hasil dari setiap perbandingan ditampilkan dengan benar, sesuai dengan operator yang digunakan.

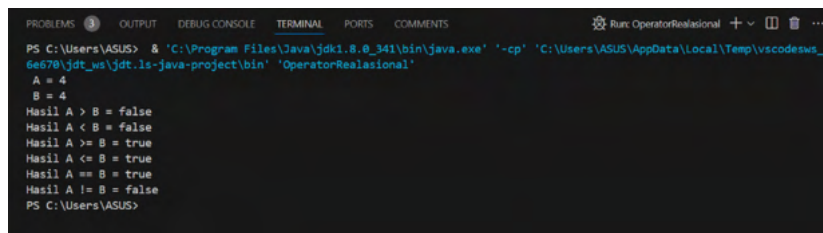
[Nomor Soal] Penyusunan Algoritma dan Kode Program

- 1) Rancang desain solusi atau algoritma
 1. Salin potongan kode
 2. Ubahlah nilai pada potongan kode sesuai yang diminta
 3. Lihat dan simpulkan perbandingan nilai A dan nilai B yang dihasilkan pada luaran
 4. Selesai
- 2) Tuliskan kode program dan luaran

```

1 public class OperatorRelasional {
2     public static void main(String[] args) {
3         int nilaiA = 4;
4         int nilaiB = 4;
5         boolean hasil;
6
7         System.out.println(" A = " + nilaiA + "\n B = " + nilaiB);
8         // apakah A lebih besar dari B?
9         hasil = nilaiA > nilaiB;
10        System.out.println("Hasil A > B = " + hasil);
11
12        // apakah A lebih kecil dari B?
13        hasil = nilaiA < nilaiB;
14        System.out.println("Hasil A < B = " + hasil);
15
16        // apakah A lebih besar samadengan B?
17        hasil = nilaiA >= nilaiB;
18        System.out.println("Hasil A >= B = " + hasil);
19
20        // apakah A lebih kecil samadengan B?
21        hasil = nilaiA <= nilaiB;
22        System.out.println("Hasil A <= B = " + hasil);
23
24        // apakah nilai A sama dengan B?
25        hasil = nilaiA == nilaiB;
26        System.out.println("Hasil A == B = " + hasil);
27
28        // apakah nilai A tidak samadengan B?
29        hasil = nilaiA != nilaiB;
30        System.out.println("Hasil A != B = " + hasil);
31    }
32 }
33

```



```

PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\ASUS> & "C:\Program Files\Java\jdk1.8.0_341\bin\java.exe" "-cp" "C:\Users\ASUS\AppData\Local\Temp\vscodesws_6e678\jdt_ws\jdt.ls-java-project\bin" "OperatorRelasional"
A = 4
B = 4
Hasil A > B = false
Hasil A < B = false
Hasil A >= B = true
Hasil A <= B = true
Hasil A == B = true
Hasil A != B = false
PS C:\Users\ASUS>

```

[Nomor Soal] Kesimpulan

1) Analisa

Kesimpulannya, program ini menyelesaikan permasalahan dengan membandingkan dua nilai (nilaiA dan nilaiB) menggunakan berbagai operator relasional. Algoritma yang digunakan melibatkan mendeklarasikan variabel, melakukan perbandingan dengan operator (>, <, >=, <=, ==, !=), dan menampilkan hasil setiap perbandingan. Kode program menjalankan setiap langkah dengan benar, memberikan hasil perbandingan yang tepat dan jelas, sehingga sesuai dengan tujuan untuk menunjukkan bagaimana operator relasional bekerja dalam Java.

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Increment dan Decrement	17 September 2024

[Nomor Soal] Identifikasi Masalah:

1) Uraikan permasalahan dan variable

```
public class operator {
    public static void main(String[] args) {
        int a = 10;
        System.out.println("# Post Increment #");
        System.out.println("=====");
        System.out.println("Isi variabel a: " + a);
        System.out.println("Isi variabel a: " + a++);
        System.out.println("Isi variabel a: " + a);

        System.out.println();

        int b = 10;
        System.out.println("# Pre Increment #");
        System.out.println("=====");
        System.out.println("Isi variabel b: " + b);
        System.out.println("Isi variabel b: " + ++b);
        System.out.println("Isi variabel b: " + b);

        System.out.println();

        int c = 10;
        System.out.println("# Post Decrement #");
        System.out.println("=====");
        System.out.println("Isi variabel c: " + c);
        System.out.println("Isi variabel c: " + c--);
        System.out.println("Isi variabel c: " + c);

        System.out.println();

        int d = 10;
        System.out.println("# Pre Decrement #");
        System.out.println("=====");
        System.out.println("Isi variabel d: " + d);
        System.out.println("Isi variabel d: " + --d);
        System.out.println("Isi variabel d: " + d);
    }
}
```

Luaran:

```
# Post Increment #
=====
Isi variabel a: 10
Isi variabel a: 10
Isi variabel a: 11
# Pre Increment #
=====
```

Isi variabel b: 10

Isi variabel b: 11

Isi variabel b: 11

Post Decrement

=====

Isi variabel c: 10

Isi variabel c: 10

Isi variabel c: 9

Pre Decrement

=====

Isi variabel d: 10

Isi variabel d: 9

Isi variabel d: 9

Latihan 4.

4.1. Berdasarkan luaran program Contoh 4, bandingkan hasil Post dan Pre untuk Increment dan Decrement!

3.

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

1. Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia
<https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMylw>
[Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman](#)
2. <https://www.youtube.com/watch?v=PzCMZOexZM>
[Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman https://www.youtube.com/watch?v=LcFgl0yrKEw](#)

[Nomor Soal] Analisis dan Argumentasi

3) Uraikan rancangan solusi yang diusulkan.

1. Buat Kelas dan Metode:

- Buat kelas operator dan metode main.

2. Post Increment:

- Deklarasikan variabel a dengan nilai 10.
- Tampilkan nilai awal a.

3. Pre Increment:

- Deklarasikan variabel b dengan nilai 10.
- Tampilkan nilai awal b.

4. Post Decrement:

- Deklarasikan variabel c dengan nilai 10.
- Tampilkan nilai awal c.

5. Pre Decrement:

- Deklarasikan variabel d dengan nilai 10.
- Tampilkan nilai awal d.

4) Analisis solusi, kaitkan dengan permasalahan.

1. Post Increment (a++): Menampilkan nilai variabel sebelum ditambah, lalu menampilkan nilai setelah ditambah. Ini membuktikan bahwa nilai lama ditampilkan sebelum perubahan.
2. Pre Increment (++b): Menampilkan nilai variabel setelah ditambah. Ini membuktikan bahwa nilai diubah sebelum ditampilkan.

3. Post Decrement (c--): Menampilkan nilai variabel sebelum dikurangi, lalu menampilkan nilai setelah dikurangi. Ini menunjukkan bahwa nilai lama ditampilkan sebelum perubahan.
4. Pre Decrement (--d): Menampilkan nilai variabel setelah dikurangi. Ini menunjukkan bahwa nilai diubah sebelum ditampilkan.

[Nomor Soal] Penyusunan Algoritma dan Kode Program

- 1) Rancang desain solusi atau algoritma
 1. Salin potongan kode operator pada VSCode
 2. Run pemrograman post dan pre increment dan decrement
 3. Lihat dan bandingkan luaran yang dihasilkan
 4. Selesai
- 2) Tuliskan kode program dan luaran

```
1 public class IncreDecre {
2     public static void main(String[] args) {
3         int a = 10;
4         System.out.println("# Post Increment #");
5         System.out.println("=====");
6         System.out.println("Isi variabel a: " + a);
7         System.out.println("Isi variabel a: " + a++);
8         System.out.println("Isi variabel a: " + a);
9
10        System.out.println();
11
12        int b = 10;
13        System.out.println("# Pre Increment #");
14        System.out.println("=====");
15        System.out.println("Isi variabel b: " + b);
16        System.out.println("Isi variabel b: " + ++b);
17        System.out.println("Isi variabel b: " + b);
18
19        System.out.println();
20
21        int c = 10;
22        System.out.println("# Post Decrement #");
23        System.out.println("=====");
24        System.out.println("Isi variabel c: " + c);
25        System.out.println("Isi variabel c: " + c--);
26        System.out.println("Isi variabel c: " + c);
27
28        System.out.println();
29
30        int d = 10;
31        System.out.println("# Pre Decrement #");
32        System.out.println("=====");
33        System.out.println("Isi variabel d: " + d);
34        System.out.println("Isi variabel d: " + --d);
35        System.out.println("Isi variabel d: " + d);
36    }
37 }
```

```
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
Isi variabel b: 10
Isi variabel b: 11
Isi variabel b: 11

# Post Decrement #
=====
Isi variabel c: 10
Isi variabel c: 10
Isi variabel c: 9

# Pre Decrement #
=====
Isi variabel d: 10
Isi variabel d: 9
Isi variabel d: 9
PS C:\Users\ASUS>
```

[Nomor Soal] Kesimpulan

1) Analisa

Program ini dengan jelas menunjukkan perbedaan antara Post dan Pre Increment serta Post dan Pre Decrement. Dengan menggunakan algoritma yang tepat, program mendeklarasikan variabel, menerapkan operasi increment dan decrement, dan menampilkan hasil sebelum dan setelah perubahan nilai. Kode program berhasil membedakan efek dari Post dan Pre operator, dengan Post Increment dan Post Decrement menampilkan nilai lama sebelum perubahan, sedangkan Pre Increment dan Pre Decrement menampilkan nilai yang telah diubah. Program ini efektif dalam menjelaskan perbedaan mendasar antara jenis-jenis operator ini.

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Logika	17 September 2024
[Nomor Soal] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variable</p> <pre>public class OperatorLogika { public static void main (String [] args) { boolean a = true; boolean b = false; boolean c; c = a && b; System.out.println("true && false = " +c); } }</pre> <p>Luaran: true && false = false</p> <p>Latihan 5</p> <p>5.1. Tambahkan baris kode untuk memeriksa a b. 5.2. Ubahlah nilai a = false dan b = false. Analisa perubahan dan perbedaan boolean yang terjadi! 5.2. Apabila diketahui pernyataan a b && a !b. Uraikan urutan logika yang akan dikerjakan! Analisa luaran true atau false dari pernyataan tersebut!</p> <p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <ol style="list-style-type: none"> 1. Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMylw Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman 2. https://www.youtube.com/watch?v=PzCMZObexZM Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman https://www.youtube.com/watch?v=LcFglOyrKEw 		
[Nomor Soal] Analisis dan Argumentasi		
<p>1) Uraikan rancangan solusi yang diusulkan.</p> <ol style="list-style-type: none"> 1) Buat kelas operator logika 2) Deklarasikan dua variabel Boolean dengan nilai true dan false 3) Operasi logika menggunakan operator logika AND (&&) untuk membandingkan a dan b. 4) Tampilkan dan cetak hasil dari operasi a && b dengan format "true && false = ", diikuti oleh nilai hasil c. <p>2) Analisis solusi, kaitkan dengan permasalahan.</p> <ol style="list-style-type: none"> 1) Deklarasi Variabel: Program mendeklarasikan dua variabel boolean (a dan b), dengan nilai true dan false masing-masing. 2) Operasi Logika: Program melakukan operasi logika AND (&&) antara a dan b, dan menyimpan hasilnya dalam variabel c. 3) Hasil Output: Program menampilkan hasil dari operasi tersebut, yang menunjukkan bahwa true && false menghasilkan false. 		

[Nomor Soal] Penyusunan Algoritma dan Kode Program

1) Rancang desain solusi atau algoritma

1. Salin potongan kode dari operator logika ke VSCode
2. Tambahkan baris kode untuk memeriksa `a || b`
3. Mengubah nilai `a = false` dan `b = false` lalu analisa perubahan dan perbedaan boolean yang terjadi
4. Menguraikan urutan logika yang akan dikerjakan lalu analisa luaran `true` atau `false` dari pemrograman
5. Selesai

2) Tuliskan kode program dan luaran

```
C:\> Users > ASUS > Documents > J TrueFalse.java > TrueFalse > main(String[])
Codeium: Refactor | Explain
1 class TrueFalse {
2   public static void main (String [] args) {
3     boolean a = true; // variabel a diatur ke true
4     boolean b = false; // variabel b diatur ke false
5     boolean c; // variabel c untuk menyimpan hasil operasi logika
6     c = a && b;
7     System.out.println("true && false = " + c); //Menampilkan hasil dari operasi logika AND (true && false)
8   }
9 }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\ASUS> & "C:\Program Files\Java\jdk1.8.0_341\bin\java.exe" "-cp" "C:\Users\ASUS\AppData\Local\Temp\vscodesws_6e670\jdt_ws\jdt.ls-java-project\bin" "TrueFalse"
true && false = false
PS C:\Users\ASUS>
```

[Nomor Soal] Kesimpulan

1) Analisa

Program ini menunjukkan cara kerja operator logika AND (&&) dengan menggunakan dua variabel boolean. Permasalahan yang dihadapi adalah untuk membuktikan hasil dari operasi logika AND antara `true` dan `false`. Algoritma yang digunakan melibatkan mendeklarasikan dua variabel boolean, melakukan operasi logika AND pada variabel tersebut, dan menyimpan hasilnya dalam variabel ketiga. Kode program menjalankan langkah-langkah ini dengan benar dan menampilkan hasil `false`, sesuai dengan prinsip bahwa operator AND hanya menghasilkan `true` jika kedua operand bernilai `true`. Program ini efektif dalam menjelaskan bagaimana operator logika AND berfungsi.

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Kondisional (Ternary)	17 September 2024
[Nomor Soal] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variable</p> <pre>public class OperatorKondisi{ public static void main(String[] args){ String status = ""; int nilai = 80; status = (nilai > 60)?"Lulus":"Gagal"; System.out.println(status); } }</pre> <p>Luaran: Lulus</p> <p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <ol style="list-style-type: none"> Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMylw Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman https://www.youtube.com/watch?v=PzCMZOboxZM Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman https://www.youtube.com/watch?v=LcFgl0yrKEw 		
[Nomor Soal] Analisis dan Argumentasi		
<p>1) Uraikan rancangan solusi yang diusulkan.</p> <ol style="list-style-type: none"> Buat kelas operator kondisi Deklarasikan variabel String bernama status dengan nilai awal kosong dan deklarasikan variabel int bernama nilai dan set nilai awalnya ke 80. Gunakan operator ternary dan diterapkan untuk mengevaluasi kondisi nilai > 60. Jika kondisi benar (true), set status ke "Lulus" dan jika kondisi salah (false), set status ke "Gagal". Tampilkan hasil dengan cetak nilai status menggunakan System.out.println(). <p>2) Analisis solusi, kaitkan dengan permasalahan.</p> <ol style="list-style-type: none"> Deklarasi variabel: mendeklarasikan variabel nilai yang menyimpan angka nilai siswa (80 dalam kasus ini) dan variabel status untuk menyimpan hasil evaluasi. Operator ternary: program menggunakan operator ternary untuk mengevaluasi kondisi nilai > 60. Operator ternary ini memungkinkan penentuan nilai status dengan sintaks yang ringkas: jika kondisi benar, status diatur ke "Lulus"; jika salah, status diatur ke "Gagal". Output: program menampilkan hasil evaluasi dengan mencetak nilai status. Karena nilai 80 lebih besar dari 60, hasil yang ditampilkan adalah "Lulus". 		
[Nomor Soal] Penyusunan Algoritma dan Kode Program		
<p>1) Rancang desain solusi atau algoritma</p> <ol style="list-style-type: none"> Salin potongan kode pemrograman ke VSCode Ubah nilai yang pada kode 80 menjadi 60 sesuai perintah 		

3. Run kode pemrograman dan lihat outputnya
4. Selesai.

2) Tuliskan kode program dan luaran

```
C: > Users > ASUS > Documents > J OperatorKondisi.java > OperatorKondisi > main(String[])
Codeium: Refactor | Explain
1 public class OperatorKondisi{
  Run | Debug | Codeium: Refactor | Explain | Generate Javadoc | X
2 public static void main( String[] args ){
3     String status = "";
4     int nilai = 60; // Nilai = 60
5     status = (nilai > 60)? "Lulus": "Gagal"; // Nilai > 60 ? "Lulus" : "Gagal"
6     System.out.println( status );
7 }
```

```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
Run: OperatorKondisi + - [ ] [ ] ... ^
PS C:\Users\ASUS> & 'C:\Program Files\Java\jdk1.8.0_341\bin\java.exe' '-cp' 'C:\Users\ASUS\AppData\Local\Temp\vscodesws_
6e670\jdt_ws\jdt.ls-java-project\bin' 'OperatorKondisi'
Gagal
PS C:\Users\ASUS>
```

[Nomor Soal] Kesimpulan

1) Analisa

Program ini dengan efektif menggunakan operator ternary untuk menentukan apakah seorang siswa lulus atau gagal berdasarkan nilai yang diberikan. Permasalahan yang dihadapi adalah menentukan status kelulusan, dan solusi ini melibatkan penggunaan operator ternary untuk mengevaluasi apakah nilai siswa lebih besar dari 60. Dengan nilai 80, program menetapkan status sebagai "Lulus" dan menampilkannya. Program ini berhasil menunjukkan bagaimana operator ternary dapat digunakan untuk membuat keputusan sederhana dalam kode dengan cara yang ringkas dan jelas.

Nama & NPM	Topik:	Tanggal:
Nadya Putri Marsela G1F024036	Operator Bitwise	17 September 2024
[Nomor Soal] Identifikasi Masalah:		
<p>1) Uraikan permasalahan dan variable</p> <pre> public class operator { 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); hasil = ~a; System.out.println("Hasil dari ~a : " + hasil); hasil = a >> 1; System.out.println("Hasil dari a >> 1 : " + hasil); hasil = b << 2; System.out.println("Hasil dari b << 2 : " + hasil); } } </pre> <p>Luaran: Hasil dari a & b : 2 Hasil dari a b : 15 Hasil dari a ^ b : 13 Hasil dari ~a : -11 Hasil dari a >> 1 : 5 Hasil dari b << 2 : 28</p>		
<p>2) Rincikan sumber informasi yang relevan (buku / webpage)</p> <ol style="list-style-type: none"> Video penjelasan pembelajaran dapat diakses pada Chanel Youtube Rumah Ilmu Raflesia https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMylw Video Materi 1 – Definisi operator, unary, binary, ternary, operasi aritmatika, penugasan, relasional, increment/decrement pada laman https://www.youtube.com/watch?v=PzCMZOboxZM Video Materi 2 – Operator Logika, Kondisional, Bitwise, dan contoh pembahasan soal pada laman https://www.youtube.com/watch?v=LcFgl0yrKEw 		
[Nomor Soal] Analisis dan Argumentasi		

- 1) Uraikan rancangan solusi yang diusulkan.
1. Buat kelas operator bitwise
2. Deklarasikan variabel a dengan nilai 10 dan b dengan nilai 7 dan deklarasikan variabel int bernama hasil untuk menyimpan hasil operasi bitwise.
3. Operasi Bitwise dan Shift:
4. AND (&):
 - a. Lakukan operasi bitwise AND antara a dan b, dan simpan hasilnya di hasil.
 - b. Cetak hasil operasi a & b.
5. OR (|):
 - a. Lakukan operasi bitwise OR antara a dan b, dan simpan hasilnya di hasil.
 - b. Cetak hasil operasi a | b.
6. XOR (^):
 - a. Lakukan operasi bitwise XOR antara a dan b, dan simpan hasilnya di hasil.
 - b. Cetak hasil operasi a ^ b.
7. NOT (~):
 - a. Lakukan operasi bitwise NOT pada a, dan simpan hasilnya di hasil.
 - b. Cetak hasil operasi ~a.
8. Right Shift (>>):
 - a. Lakukan operasi right shift pada a dengan 1 bit, dan simpan hasilnya di hasil.
 - b. Cetak hasil operasi a >> 1.
9. Left Shift (<<):
 - a. Lakukan operasi left shift pada b dengan 2 bit, dan simpan hasilnya di hasil.
 - b. Cetak hasil operasi b << 2.

- 2) Analisis solusi, kaitkan dengan permasalahan.

1. Deklarasikan dua variabel integer (a dan b) dan satu variabel hasil untuk menyimpan hasil operasi.
2. Operasi bitwise: AND (&): Program menghitung a & b, yang melakukan operasi bitwise AND dan menyimpan hasilnya. Hasil ini menunjukkan bit yang setara di kedua operand.
3. OR (|): Program menghitung a | b, yang melakukan operasi bitwise OR dan menyimpan hasilnya. Hasil ini menunjukkan bit yang setara di salah satu operand.
4. XOR (^): Program menghitung a ^ b, yang melakukan operasi bitwise XOR dan menyimpan hasilnya. Hasil ini menunjukkan bit yang berbeda di operand.
5. NOT (~): Program menghitung ~a, yang melakukan operasi bitwise NOT pada a dan menyimpan hasilnya. Ini membalikkan semua bit dalam a.
6. Operasi Shift:
 - Right Shift (>>): Program menghitung a >> 1, yang melakukan operasi right shift pada a dengan 1 bit, memindahkan bit-bit ke kanan dan mengisi bit kosong dengan bit tanda.
 - Left Shift (<<): Program menghitung b << 2, yang melakukan operasi left shift pada b dengan 2 bit, memindahkan bit-bit ke kiri dan mengisi bit kosong dengan nol.

[Nomor Soal] Penyusunan Algoritma dan Kode Program

- 1) Rancang desain solusi atau algoritma
1. Salin potongan kode pemrograman ke VSCode
2. Run kode pemrograman dan lihat luarannya
3. Ubah nilai 2 menjadi bilangan biner 0010
4. Ubah nilai 15 menjadi bilangan biner 1101
5. Ubah nilai 13 menjadi bilangan biner 1011
6. Selesai.

2) Tuliskan kode program dan luaran

```
C: > Users > ASUS > Documents > J Bitwise.java > Bitwise
Codeium: Refactor | Explain
1 public class Bitwise {
2     public static void main(String[] args) {
3         int a = 10;
4         int b = 7;
5         int hasil;
6
7         hasil = a & b;
8         System.out.println("Hasil dari a & b : " + hasil );
9
10        hasil = a | b;
11        System.out.println("Hasil dari a | b : " + hasil );
12
13        hasil = a ^ b;
14        System.out.println("Hasil dari a ^ b : " + hasil );
15
16        hasil = ~a;
17        System.out.println("Hasil dari ~a : " + hasil );
18
19        hasil = a >> 1;
20        System.out.println("Hasil dari a >> 1 : " + hasil );
21
22        hasil = b << 2;
23        System.out.println("Hasil dari b << 2 : " + hasil );
24    } }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\ASUS> & 'C:\Program Files\Java\jdk1.8.0_341\bin\java.exe' '-cp' 'C:\Users\ASUS\AppData\Local\Temp\vscodesw_6e670\jdt_ws\jdt.ls-java-project\bin' 'Bitwise'
Hasil dari a & b : 2
Hasil dari a | b : 15
Hasil dari a ^ b : 13
Hasil dari ~a : -11
Hasil dari a >> 1 : 5
Hasil dari b << 2 : 28
PS C:\Users\ASUS>
```

[Nomor Soal] Kesimpulan

2) Analisa

Program ini dengan jelas menunjukkan bagaimana berbagai operasi bitwise dan shift bekerja pada dua variabel integer. Melalui operasi AND, OR, XOR, dan NOT, serta shift kanan dan kiri, program mengilustrasikan perubahan nilai bitwise dan pergeseran bit pada angka 10 dan 7. Hasil yang ditampilkan mencerminkan bagaimana setiap operasi memanipulasi bit-bit angka, membantu memahami cara kerja operasi bitwise dan shift dalam pemrograman dengan cara yang praktis dan terukur.