

**RENCANA PEMBELAJARAN SEMESTER  
PROGRAM STUDI SARJANA TERAPAN TEKNIK INFORMATIKA  
DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER**



|                      |  |                                       |                                |                                     |                              |                |       |
|----------------------|--|---------------------------------------|--------------------------------|-------------------------------------|------------------------------|----------------|-------|
| Kode                 | VI-043103  | Mata Kuliah                           | Pemrograman Berorientasi Obyek |                                     |                              |                |       |
| Bobot SKS            | 2  | Semester                              | 3                              |                                     |                              |                |       |
| Kelompok MK          | MK Umum/ MK Dasar Keahlian/ MK Keahlian  | Jam/minggu                            | 2                              |                                     |                              |                |       |
| Tim Pengampu MK      | Tita Karlita, Yuliana Setyowati  |                                       |                                | Noid: RF-DTEL-PSTE-4.05.Rev.01[031] |                              |                |       |
| Capaian Pembelajaran | Mahasiswa memahami konsep pemrograman berorientasi obyek yang meliputi enkapsulasi, inheritance, dan polymorphism, mengimplementasikan konsep pemrograman berorientasi obyek tersebut ke dalam bahasa pemrograman Java, serta memecahkan permasalahan menjadi sebuah desain konsep pemrograman berorientasi obyek dan kemudian mengimplementasikannya menjadi sebuah program komputer, baik secara individu maupun berkelompok/kerjasama tim.  |                                       |                                |                                     |                              |                |       |
| Pokok Bahasan        | <ol style="list-style-type: none"> <li>1. Konsep dasar pemrograman berorientasi obyek</li> <li>2. Perbedaan pembuatan program dengan pendekatan paradigma berorientasi obyek dan prosedural</li> <li>3. Mengidentifikasi class, obyek, variabel, dan method dari sebuah kasus atau permasalahan</li> <li>4. Mengenal lingkungan dan teori dasar pemrograman Java yang meliputi tipe data, sintaks dan struktur bahasa Java</li> <li>5. Konsep utama pemrograman berorientasi obyek yang meliputi enkapsulasi, inheritance, dan polymorphism</li> <li>6. Tata aturan membuat class, obyek, dan cara pengaksesan anggota class</li> <li>7. Konsep lanjut pemrograman berorientasi obyek yang meliputi abstract class, interface, inner class, dan generics</li> <li>8. Tata cara menangani kesalahan program dengan menggunakan Exception</li> <li>9. Karakteristik dan sifat beberapa macam tipe data koleksi yang terdapat dalam package java.util</li> <li>10. Mampu membuat dokumentasi kode program berorientasi obyek</li> </ol> |                                       |                                |                                     |                              |                |       |
| Referensi            | <ol style="list-style-type: none"> <li>1. Oracle Java Documentation, The Java™ Tutorials, <a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a>, Copyright © 1995, Oracle 2015.</li> <li>2. Tita Karlita, Yuliana Setrowati, Rizky Yuniar Hakkun, Pemrograman Berorientasi Obyek, PENS-2012</li> <li>3. Sun Java Programming, Sun Educational Services, Student Guide, Sun Microsystems, 2001.</li> <li>4. John R. Hubbard, Programming With Java, McGraw-Hill, ISBN: 0-07-142040-1, 2004.</li> <li>5. Patrick Niemeyer, Jonathan Knudsen, Learning Java, O’reilly, CA, ISBN: 1565927184, 2000.</li> <li>6. Philip Heller, Simon Roberts, Complete Java 2 Certification Study Guide, Third Edition, Sybex, San Francisco, London, ISBN: 0-7821-4419-5, 2002.</li> <li>7. Herbert Schildt, The Complete Reference, Java™ Seventh Edition, Mc Graw Hill, Osborne, ISBN: 978-0-07-163177-8, 2007</li> </ol>   |                                       |                                |                                     |                              |                |       |
| MK Prasyarat         | 1. Konsep Pemrograman 2. Logika dan Algoritma  |                                       |                                |                                     |                              |                |       |
| Media Pembelajaran   | Software: Text Editor (Notepad, MS Word dll), Net Beans<br>Hardware: PC/Laptop, LCD Projector  |                                       |                                |                                     |                              |                |       |
| Asesmen (%)          | UTS (30 %), UAS (40 %), Tugas (30 %)   |                                       |                                |                                     |                              |                |       |
| Mgg Ke-              | Sub Capaian Pembelajaran MK<br>(Kemampuan Akhir Yang Direncanakan)   | Bahan Kajian<br>(Materi Pembelajaran) | Bentuk Pembelajaran            | Waktu Belajar<br>(menit)            | Kriteria Asesmen (Indikator) | Bentuk Asesmen | Bobot |

|     |   |  |   |  |  |   |       |
|-----|---|--|---|--|--|---|-------|
| (1) | <p>1) Mahasiswa mengenal konsep pemrograman berorientasi obyek dan manfaatnya</p> <p>2) Mahasiswa memahami tentang konsep pemrograman berorientasi obyek dan perbandingannya dengan pemrograman procedural</p> <p>3) Mahasiswa memahami konsep class, obyek, variabel, dan method</p> <p>4) Mahasiswa mengenal bahasa pemrograman Java dan OOP, sejarah Java, kelebihan Java, dan JVM.</p> <p>5) Mahasiswa mengetahui cara mengidentifikasi class, obyek, variabel, dan method dalam permasalahan pemrograman</p> | <ul style="list-style-type: none"> <li>○ Konsep dasar pemrograman berorientasi obyek</li> <li>○ Perbandingan pemrograman prosedural dan pemrograman berorientasi obyek</li> <li>○ Konsep class, obyek, variabel, dan method</li> <li>○ Java dan OOP, sejarah Java, kelebihan Java, dan JVM.</li> </ul> | Kuliah Pengantar & Brainstorming, Diskusi     | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan dan menjelaskan konsep utama pemrograman berorientasi obyek <b>dengan tepat dan benar</b></li> <li>○ Menjelaskan perbedaan antara pembuatan program menggunakan paradigma berorientasi obyek dan prosedural <b>dengan tepat dan benar</b></li> <li>○ Mengidentifikasi class, obyek, variabel, dan method dari suatu permasalahan pemrograman sederhana <b>dengan tepat dan benar</b></li> </ul>   | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (2) | Mahasiswa mengetahui dasar-dasar dan aturan pemrograman di Java sebagai bahasa implementasi pemrograman berorientasi obyek  | <ul style="list-style-type: none"> <li>○ Komentar</li> <li>○ Sintaks bahasa Java</li> <li>○ identifiers</li> <li>○ Kata-kata kunci</li> <li>○ Tipe data primitif</li> <li>○ Inisialisasi &amp; nilai default</li> <li>○ Casting &amp; promotion</li> </ul>   | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan berbagai jenis dan kegunaan komentar program <b>dengan tepat dan benar</b></li> <li>○ Menyebutkan tata cara penamaan identifiers <b>dengan tepat dan benar</b></li> <li>○ Menyajikan kata-kata kunci dan reserved word yang dipakai dalam Java <b>dengan tepat dan benar</b></li> <li>○ Menyebutkan sifat dan karakteristik tipe data primitif <b>dengan tepat dan benar</b></li> <li>○ Menyebutkan perbedaan tipe data primitif dan obyek <b>dengan tepat dan benar</b></li> <li>○ Menyebutkan aturan casting &amp; promotion pada tipe data primitif <b>dengan tepat dan benar</b></li> </ul> | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (3) | Mahasiswa mengetahui dan memahami operator-operator dasar yang ada di Java beserta contoh penggunaannya   | <ul style="list-style-type: none"> <li>○ - Bitwise</li> <li>○ AND (&amp;), OR ( ), XOR (^), NOT (!)</li> <li>○ - Short circuit</li> <li>○ AND (&amp;&amp;), OR (  )</li> <li>○ - Shift</li> </ul>  | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan berbagai macam jenis operator sesuai kelompoknya disertai dengan contoh penggunaan <b>dengan tepat dan benar</b></li> </ul>   | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |

|     |  |   |   |  |  |   |       |
|-----|--|---|---|--|--|---|-------|
|     |  | <ul style="list-style-type: none"> <li>○ Left shift (&lt;&lt;), right shift (&gt;&gt;), unsigned right shift (&gt;&gt;&gt;)</li> <li>○ - increment (++)</li> <li>○ - decrement (--)</li> <li>○ - Operator precedence</li> </ul>   |   |  |  |   |       |
| (4) | <ol style="list-style-type: none"> <li>1) Menerangkan berbagai macam cara melakukan percabangan dan keyword yang dapat digunakan disertai dengan contoh penggunaan</li> <li>2) Menerangkan berbagai macam cara melakukan perulangan dan keyword yang dapat digunakan disertai dengan contoh penggunaan</li> <li>3) Menerangkan tipe data array, sifat dan karakteristik, dan operasi pada array disertai dengan contoh penggunaan</li> </ol> | <ul style="list-style-type: none"> <li>○ Percabangan : if, if-else, else-if, switch</li> <li>○ Perulangan: for, while, do-while, break &amp; continue</li> <li>○ Declaring arrays</li> <li>○ Creating arrays</li> <li>○ Initializing arrays</li> <li>○ Multidimensional arrays</li> <li>○ Array bounds</li> <li>○ Array resizing</li> <li>○ Copying arrays</li> </ul> | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan berbagai macam cara melakukan percabangan dan keyword yang dapat digunakan disertai dengan contoh penggunaan <b>dengan tepat dan benar</b></li> <li>○ Menyebutkan berbagai macam cara melakukan perulangan dan keyword yang dapat digunakan disertai dengan contoh penggunaan <b>dengan tepat dan benar</b></li> <li>○ Menyebutkan tipe data array, sifat dan karakteristik, dan operasi pada array disertai dengan contoh penggunaan <b>dengan tepat dan benar</b></li> </ul>  | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (5) | <ol style="list-style-type: none"> <li>1) Mahasiswa memahami bagaimana cara membuat class dan cara pengaksesan anggota class</li> <li>2) Mahasiswa memahami cara membuat obyek dan step by step sebuah obyek dibuat</li> <li>3) Mahasiswa memahami konsep constructor</li> <li>4) Mahasiswa memahami konsep dan penggunaan String</li> </ol>   | <ul style="list-style-type: none"> <li>○ Declaring class, attributes, methods, dan obyek</li> <li>○ Accessing object members</li> <li>○ Default Constructor</li> <li>○ Overloading Constructor</li> <li>○ this keyword</li> <li>○ Literal string</li> <li>○ Membuat obyek String</li> <li>○ String vs StringBuffer</li> </ul>   | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan cara mendeklarasikan class, attributes, methods, dan obyek dalam sebuah file sumber Java <b>dengan benar</b></li> <li>○ Menyebutkan properti sebuah kelas dan bagaimana cara mengkasusnya. Termasuk juga menjelaskan tentang kata kunci static <b>dengan benar</b></li> <li>○ Menyebutkan ciri-ciri dan konsep default constructor <b>dengan benar</b></li> <li>○ Menyebutkan kegunaan dan manfaat melakukan overloading constructor <b>dengan benar</b></li> <li>○ Menyebutkan penggunaan kata kunci this <b>dengan benar</b></li> <li>○ Menyebutkan konsep literal</li> </ul> | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |

|     |  |  |  |   |   |  |              |
|-----|--|--|--|---|---|--|--------------|
|     |  |  |  |   | <p>string <b>dengan benar</b></p> <ul style="list-style-type: none"> <li>o Menyebutkan sifat dan karakteristik String <b>dengan benar</b></li> <li>o Menyebutkan perbedaan antara String dan StringBuffer <b>dengan benar</b></li> </ul>  |  |              |
| (6) | <p>1) Mahasiswa memahami konsep Enkapsulasi</p> <p>2) Mahasiswa memahami cara mengelola kelas dalam bentuk package dan import</p>  | <ul style="list-style-type: none"> <li>o Encapsulation</li> <li>o Information hiding</li> <li>o Access modifier : public dan private</li> <li>o package keyword</li> <li>o Importing classes</li> </ul>  | <p>Kuliah, Diskusi kelompok, maju di depan kelas</p> | <p>TM: 100 menit</p> <p>Tgs: 100 menit</p> <p>BM: 100 menit</p> | <ul style="list-style-type: none"> <li>o Menyebutkan konsep enkapsulasi dan tata cara implementasinya <b>dengan benar</b></li> <li>o Menyebutkan kegunaan mode akses public dan private <b>dengan benar</b></li> <li>o Menyebutkan konsep mengelola class dengan menggunakan package <b>dengan benar</b></li> <li>o Menyebutkan penggunaan kata kunci package dan import <b>dengan benar</b></li> <li>o Menyebutkan contoh kasus dan program yang menerapkan konsep enkapsulasi <b>dengan benar</b></li> <li>o Menyebutkan contoh kasus dan program yang membutuhkan kata kunci package dan import <b>dengan benar</b></li> </ul> | <p>Tugas, penyelesaian soal/studi kasus di kelas</p> | <p>6.25%</p> |
| (7) | <p>1) Mahasiswa memahami konsep inheritance</p> <p>2) Mahasiswa memahami konsep constructor tidak diwariskan</p> <p>3) Mahasiswa memahami cara mengakses properti parent class</p> <p>4) Mahasiswa memahami konsep single dan multilevel inheritance</p> | <ul style="list-style-type: none"> <li>o Basic Concept of Inheritance</li> <li>o Declaring inheritance in Java</li> <li>o Accessing parent's member</li> <li>o Accesibility criteria</li> <li>o super keyword</li> <li>o Constructor does not been inherited</li> <li>o Invoking parent's constructor</li> <li>o Mode akses : default</li> </ul> | <p>Kuliah, Diskusi kelompok, maju di depan kelas</p> | <p>TM: 100 menit</p> <p>Tgs: 100 menit</p> <p>BM: 100 menit</p> | <ul style="list-style-type: none"> <li>o Menyebutkan konsep inheritance dan tata cara implementasinya <b>dengan benar</b></li> <li>o Menyebutkan penggunaan kata kunci super <b>dengan benar</b></li> <li>o Menyebutkan konsep constructor tidak diwariskan <b>dengan benar</b></li> <li>o Menyebutkan konsep single dan multilevel inheritance <b>dengan benar</b></li> <li>o Menyebutkan contoh kasus dan</li> </ul>  | <p>Tugas, penyelesaian soal/studi kasus di kelas</p> | <p>6.25%</p> |

|      |   |  |   |  |   |   |       |
|------|---|--|---|--|---|---|-------|
|      |   | dan protected  |   |  | program yang menerapkan konsep inheritance <b>dengan benar</b><br><ul style="list-style-type: none"> <li>○ Menyebutkan contoh kasus dan program yang membutuhkan kata kunci super <b>dengan benar</b></li> <li>○ Menyebutkan kegunaan mode akses default dan protected <b>dengan benar</b></li> </ul>   |   |       |
| (8)  | <ol style="list-style-type: none"> <li>1) Mahasiswa memahami method overriding</li> <li>2) Mahasiswa memahami method overloading</li> <li>3) Mahasiswa memahami konsep polimorphisme</li> <li>4) Mahasiswa memahami penggunaan konsep polimorphisme dalam sebuah permasalahan</li> <li>5) Mahasiswa memahami kejadian virtual method invocation</li> <li>6) Memahami penggunaan keyword instanceof</li> </ol> | <ul style="list-style-type: none"> <li>○ Basic concept of polymorphism</li> <li>○ Overloading methods</li> <li>○ Overriding methods</li> <li>○ Rules about overridden methods</li> <li>○ Virtual method Invocation</li> <li>○ Polymorphic arguments</li> <li>○ instanceof statements</li> <li>○ Casting objects</li> </ul> | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan contoh kejadian dan alasan penerapan method overriding <b>dengan benar</b></li> <li>○ Menyebutkan aturan method overriding <b>dengan benar</b></li> <li>○ Menyebutkan contoh kejadian dan alasan penerapan method overloading <b>dengan benar</b></li> <li>○ Menyebutkan aturan method overloading <b>dengan benar</b></li> <li>○ Menyebutkan proses terjadinya polimorphisme <b>dengan benar</b></li> <li>○ Menyebutkan kejadian virtual method invocation <b>dengan benar</b></li> <li>○ Menyebutkan arti polymorphic arguments, instanceof dan casting objects <b>dengan benar</b></li> </ul> | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (9)  | <b>Ujian Tengah Semester (UTS)</b>  |  |   |  |   |   |       |
| (10) | <ol style="list-style-type: none"> <li>1) Mahasiswa memahami konsep abstract method, abstract class, dan interface</li> <li>2) Mahasiswa memahami aturan casting dan conversion pada obyek</li> </ol>   | <ul style="list-style-type: none"> <li>○ Abstract class</li> <li>○ Abstract methods</li> <li>○ Interface</li> <li>○ Aturan casting dan conversion pada obyek</li> </ul>  | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan definisi abstract method <b>dengan benar</b></li> <li>○ Menyebutkan definisi abstract class <b>dengan benar</b></li> <li>○ Menyebutkan karakteristik interface <b>dengan benar</b></li> <li>○ Menyebutkan perbedaan antara class dan interface <b>dengan benar</b></li> <li>○ Menyebutkan aturan casting dan conversion pada obyek yang melibatkan tipe data class dan interface <b>dengan benar</b></li> <li>○ Memberi contoh kejadian</li> </ul>   | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |

|      |   |  |   |  |   |   |       |
|------|---|--|---|--|---|---|-------|
|      |   |  |   |  | casting dan conversion pada obyek <b>dengan benar</b>   |   |       |
| (11) | <p>1) Mahasiswa memahami konsep dan berbagai macam bentuk inner class</p> <p>2) Mahasiswa mengetahui manfaat penggunaan inner class</p> <p>3) Mahasiswa mengetahui contoh penerapan inner class</p> | <ul style="list-style-type: none"> <li>○ Non static inner class</li> <li>○ Static inner class</li> <li>○ Local inner class</li> <li>○ Anonymous Non static inner class</li> </ul>  | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan aturan dan cara mendeklarasikan non static inner class <b>dengan benar</b></li> <li>○ Menyebutkan aturan dan cara mendeklarasikan static inner class <b>dengan benar</b></li> <li>○ Menyebutkan aturan dan cara mendeklarasikan local inner class <b>dengan benar</b></li> <li>○ Menyebutkan aturan dan cara mendeklarasikan anonymous Non static inner class <b>dengan benar</b></li> <li>○ Menyebutkan kegunaan inner class <b>dengan benar</b></li> </ul>   | Penilaian tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (12) | <p>1) Mahasiswa memahami konsep dan penggunaan wrapper class</p> <p>2) Mahasiswa memahami penggunaan Enum Type dalam sebuah</p>   | <ul style="list-style-type: none"> <li>○ Wrapper class</li> <li>○ Autoboxing</li> <li>○ Unboxing</li> <li>○ Fungsi-fungsi wrapper class</li> <li>○ Tata cara mendeklarasikan enum types</li> <li>○ Bentuk penggunaan enum types</li> <li>○ Fungsi-fungsi enum types</li> </ul> | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan definisi wrapper class dan hubungannya dengan tipe data primitif <b>dengan benar</b></li> <li>○ Menyebutkan konsep auto boxing dan auto unboxing untuk tipe data primitif dan wrapper class <b>dengan benar</b></li> <li>○ Menyebutkan fungsi-fungsi yang dimiliki wrapper class <b>dengan benar</b></li> <li>○ Menyebutkan tata cara mendeklarasikan enum types <b>dengan benar</b></li> <li>○ Menyebutkan konsep Enum types <b>dengan benar</b></li> <li>○ Menyebutkan bentuk-bentuk penggunaan Enum <b>dengan benar</b></li> <li>○ Menyebutkan dan menjelaskan fungsi-fungsi yang terdapat di Enum <b>dengan benar</b></li> </ul> | Tugas, penyelesaian soal/studi kasus di kelas           | 6.25% |
| (13) | Mahasiswa mengetahui berbagai jenis kesalahan dan sifatnya sehingga mampu melakukan proses penanganan kesalahan   | <ul style="list-style-type: none"> <li>○ Kategori Exception</li> <li>○ Jenis Exception yang sering terjadi</li> <li>○ Exception handling: try</li> </ul>   | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>○ Menyebutkan berbagai macam jenis Exception, kategori Exception, beserta hirarkinya</li> <li>○ Menyebutkan dan menjelaskan</li> </ul>   | Tugas, penyelesaian soal/studi kasus di kelas           | 6.25% |

|      |   |   |   |  |   |   |       |
|------|---|---|---|--|---|---|-------|
|      |   | <ul style="list-style-type: none"> <li>- catch - finally</li> <li>o Aturan mendeklarasikan Exception</li> <li>o Method overriding and exception</li> <li>o Creating your own exception</li> </ul> |   |  | <p>perbedaan antara Error dan Exception <b>dengan benar</b></p> <ul style="list-style-type: none"> <li>o Menyebutkan dan menjelaskan cara menangani Exception <b>dengan benar</b></li> <li>o Menyebutkan dan menjelaskan penggunaan keyword try, catch, finally, throw, dan throws <b>dengan benar</b></li> <li>o Menyebutkan dan menjelaskan aturan overriding method sehubungan dengan Exception <b>dengan benar</b></li> <li>o Menyebutkan dan menjelaskan aturan dan cara mendeklarasikan Eksepsi baru <b>dengan benar</b></li> </ul>       |   |       |
| (14) | Mahasiswa memahami penggunaan Java Collections Framework dan java.util package untuk menyelesaikan suatu kasus penyimpanan koleksi data | <ul style="list-style-type: none"> <li>o Java Collection Framework API</li> <li>o Collection</li> <li>o Set</li> <li>o Map</li> <li>o List</li> <li>o Iterator</li> </ul>                         | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>o Menjelaskan konsep Collection framework <b>dengan benar</b></li> <li>o Menjelaskan Collection interface <b>dengan benar</b></li> <li>o Menjelaskan Set, class Set dan contoh penggunaannya <b>dengan benar</b></li> <li>o Menjelaskan List, class List dan contoh penggunaannya <b>dengan benar</b></li> <li>o Menjelaskan Map, class Map dan contoh penggunaannya <b>dengan benar</b></li> <li>o Menjelaskan Iterator dan Enumeration untuk retrieve elemen Collection <b>dengan benar</b></li> </ul> | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (15) | Memahami cara membandingkan obyek dengan menggunakan interface Comparable dan Comparator  | <ul style="list-style-type: none"> <li>o Arrays</li> <li>o Arrays.sort()</li> <li>o Comparable</li> <li>o Comparator</li> </ul>   | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>o Memahami konsep membandingkan obyek <b>dengan benar</b></li> <li>o Memahami penggunaan Arrays class untuk membandingkan obyek <b>dengan benar</b></li> <li>o Memahami penggunaan Comparable class untuk</li> </ul>   | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |

|      |  |  |   |  |  |   |       |
|------|--|--|---|--|--|---|-------|
|      |  |  |   |  | <p>membandingkan obyek <b>dengan benar</b></p> <ul style="list-style-type: none"> <li>o Memahami penggunaan Comparator class untuk membandingkan obyek <b>dengan benar</b></li> <li>o Memahami perbedaan konsep membandingkan obyek menggunakan Comparable dan Comparator <b>dengan benar</b></li> </ul>   |   |       |
| (16) | Memahami konsep dan penggunaan Generics        | <ul style="list-style-type: none"> <li>o Generics types</li> <li>o Raw types</li> <li>o Generics methods</li> <li>o Bounded type parameters</li> <li>o Generics, inheritance, dan subtypes</li> <li>o Wildcards</li> </ul> | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>o Menerangkan tipe generic dan non generic dalam sebuah class <b>dengan benar</b></li> <li>o Menerangkan contoh raw types sebuah class generics <b>dengan benar</b></li> <li>o Menerangkan penggunaan generic dalam method <b>dengan benar</b></li> <li>o Menerangkan konsep bounded (extends dan super) type parameters dalam class dan method <b>dengan benar</b></li> <li>o Menerangkan konsep generics sehubungan dengan konsep inheritance dan subtypes <b>dengan benar</b></li> <li>o Menerangkan konsep wildcards <b>dengan benar</b></li> </ul> | Tugas, penyelesaian soal/studi kasus di kelas | 6.25% |
| (17) | Memahami cara membuat dokumentasi kode program | <ul style="list-style-type: none"> <li>o Command untuk mengenerate dokumentasi kode program</li> <li>o Doclet API</li> <li>o Taglet API</li> </ul>   | Kuliah, Diskusi kelompok, maju di depan kelas | TM: 100 menit<br>Tgs: 100 menit<br>BM: 100 menit | <ul style="list-style-type: none"> <li>o Menyebutkan command untuk mengenerate dokumentasi kode program <b>dengan benar</b></li> <li>o Menyebutkan dan menjelaskan berbagai macam Doclet API <b>dengan benar</b></li> <li>o Menyebutkan dan menjelaskan berbagai macam Taglet API <b>dengan benar</b></li> </ul>   | Tugas Akhir Semester                          | 6.25% |
| (18) | <b>Ujian Akhir Semester (UAS)</b>              |  |   |  |  |   |       |

Keterangan:  
TM : Tatap Muka



Tgs : Tugas

BM : Belajar Mandiri