|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | **POLITEKNIK NEGERI MALANG**  **JURUSAN TEKNOLOGI INFORMASI**  **PROGRAM STUDI : D4TEKNIK INFORMATIKA** | | | | | | | | | | | | |
| **RENCANA PEMBELAJARAN SEMESTER (RPS)** | | | | | | | | | | | | | | | |
| **MATA KULIAH** | | | **KODE** | **RUMPUN MATA KULIAH** | | | | | **BOBOT (sks)/jam** | | **SEMESTER** | | **TGL. PENYUSUNAN** | | |
| **PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK** | | | **RIF 4** | **INFORMATIKA DASAR** | | | | | **2 sks / 3 jam** | | **2** | | **21 Februari 2017** | | |
| **OTORISASI** | | | **Dosen Pengembang RPS** | | | | | | **Koordinator RMK** | | **Ka PRODI** | | | | |
| **Ridwan Rismanto, S.ST., M.Kom.** | | | | | | **Mungki Astiningrum, ST., M.Kom.** | | **Dr. Eng. Rosa Andrie A., ST., MT.** | | | | |
| **Capaian Pembelajaran (CP)** | | | **Capaian Pembelajaran Lulusan Program Studi(CPL-Prodi)** | | | | | | | |  | | | | |
| * Mampu merumuskan masalah, menganalisa kebutuhan sistem, merancang, dan membangun sistem berdasarkan hasil   perancangan, menguji coba, serta mengintegrasikan sistem sesuai dengan tahapan dalam membangun sistem dibidang  pengolahan data dan informasi, multimedia dan rekayasa perangkat lunak baik desktop, web, mobile, dan jaringan komputer  dengan memperhatikan aspek kinerja dan kehandalan, keamanan, kemudahan penerapan dan keberlanjutan   * Mampu menggunakan teknologi informasi modern dalam melaksanakan pekerjaan | | | | | | | | | | | | |
| **Capaian Pembelajaran Lulusan yang dibebankan pada mata kuliah (CPL-MK)** | | | | | | | | | | |  | |
| Mampu membuat program dengan menggunakan prinsip-prinsip OOP menggunakan bahasa pemrograman Java | | | | | | | | | | | | |
| **Diskripsi Singkat Mata Kuliah** | | | Mata kuliah Pemrograman Berbasis Objek (PBO) adalah matakuliah yang mengajarkan kepada mahasiswa bagaimana konsep pengembangan program dengan paradigma berorientasi objek | | | | | | | | | | | | |
| **Materi Pembelajaran / Pokok Bahasan** | | | * Konsep OOP * Class dan object * Enkpasulasi * Relasi class * Inheritance * Polimorfisme * Abstract Class * Interface * Java Basic Programming * Pengenalan GUI * Pengenalan Java API | | | | | | | | | | | | |
| **Pustaka** | | | **Utama :** | |  | | | | | | | | | | |
| Horstmann, C. S., & Cornell, G. (2007). *Core Java Volume I–Fundamentals, Eighth Edition.* Network Circle, Santa Clara: Prentice Hall.  Horstmann, C. S., & Cornell, G. (2008). *Core Java Volume II–Advanced Features, Eighth Edition.* Network Circle, Santa Clara: Prentice Hall. | | | | | | | | | | | | |
| **Pendukung :** | |  | | | | | | | | | | |
| Rickyanto, I. (2005). *Dasar Pemrograman Berorientasi Objek dengan Java 2.* Yogyakarta: Andi Offset | | | | | | | | | | | | |
| **Media Pembelajaran** | | | **Software :** | | | | **Hardware :** | | | | | | | | |
| JDK 1.8  Netbeans 8 | | | | Komputer/Laptop dengan minilmal RAM 512 MB | | | | | | | | |
| **Nama Dosen Pengampu** | | | Ridwan Rismanto, S.ST., M.Kom. | | | | | | | | | | | | |
| **Matakuliah Syarat** | | | Algoritma dan Pemrograman | | | | | | | | | | | | |
| **Minggu Ke** | **Kemampuan Akhir Yang Direncanakan**  **(Sub-CP-MK)** | **Bahan kajian**  **(Materi Pembelajaran)** | | | | **Bentuk dan Metode Pembelajaran** | | **Estimasi Waktu** | | **Pengalaman Belajar Mahasiswa** | | **Kriteria & Bentuk Penilaian** | | **Indikator Penilaian** | **Bobot (%)** |
| **(1)** | **(2)** | **(3)** | | | | **(4)** | | **(5)** | | **(6)** | | **(7)** | | **(8)** | **(9)** |
| 1 | Pengantar Konsep Dasar OOP | 1. Pembahasan silabus dan kontrak kuliah 2. Perbedaan paradigma berorientasi objek dengan paradigma struktural 3. Konsep dasar PBO 4. Class 5. Object 6. Enkapsulasi 7. Inheritance 8. Polimorfisme 9. Pengenalan pemodelan UML Class Diagram | | | | 1. Ceramah 2. Diskusi 3. Praktikum | | 4x50’ | | * Mendengarkan ceramah materi dari dosen dan berdiskusi * Praktek menerapkankonsep dasar PBO dipandu dengan jobsheet * Praktek Menggambar UML Class Diagram | | Kriteria:Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitasPraktek | | * Ketepatan dalam menjelaskan perbedaan antara paradigma OOP dan Struktural * Ketepatan dalam mempraktekkan konsep Class, object, enkapsulasi, inheritance dan polimorfisme * Ketepatan dalam memberikan contoh dari masing-masing konsep dasar OOP |  |
| 2 | Class dan Object | 1. Class 2. Object 3. Atribut 4. Method 5. Instansiasi 6. Pengaksesan Atribut dan Method 7. Try-Catch 8. UML:Diagram Class | | | | 1. Praktikum 2. Diskusi | | 4x50’ | | * Mendengarkan instruksi praktikum * Menuliskan dasar pembuatan class dan object ke kode programberdasarkan class diagram * Menerapkan penggunaan try catch | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mengidentifikasi objek * Ketepatan dalam menentukan atribut dan method dari tiap objek yang telah diidentifikasi * Ketepatan dalam praktek penggambarannya ke dalam diagram class |  |
| 3 | Enkapsulasi | 1. Konstruktor 2. Access modifier 3. Atribut/Method Class 4. Atribut/Method Instansiasi 5. Setter dan getter 6. UML:Notasi access modifier dan notasi static | | | | 1. Praktikum 2. Diskusi | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan konstruktor, access modifier, atribut, setter getter dipandu dengan jobsheet * Berdiskusi tentang perbedaan antara atribut instansiasi dan atribut class, antara method instansiasi dan method class * Berdiskusi tentang kegunaan setter dan getter | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mempraktekkan menentukan konstruktor dan ketepatan dalam melakukan instansiasi yang menggunakan konstruktor tersebut * Ketepatan dalam menentukan modifier access dari suatu atribut dan method * Ketepatan dalam menjelaskan perbedaan atribut/method instansiasi dan atribut/method class * Ketepatan dalam membuat setter dan getter * Ketepatan dalam penulisan notasi static, dan modifier access * Ketepatan dalam mempraktekkan instruksi pada jobsheet |  |
| 4 | Relasi Class | 1. Relasi Has-A (Studi kasus 1 class memiliki hubungan has-a dengan 1 objek dari klass lain) 2. Penggambaran relasi class dengan diagram class 3. Relasi Has-A (Studi kasus 1 class memiliki hubungan has-a dengan lebih dari 1 objek dari klass lain) | | | | 1. Praktikum 2. Diskusi | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan relasi class dipandu dengan jobsheet * Praktek menggambarkan desain diagram class dari kasus yang diberikan, dengan berdiskusi | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam menerjemahkan diagram class ke dalam program * Ketepatan dalam merancang diagram class dari suatu kasus tertentu * Ketepatan dalam mempraktekkan instruksi pada jobsheet | 2.5% |
| **5** | **Kuis 1** | Materi pertemuan 1-4 | | | | **Ujian Tulis/Online** | | 4x50’ | | * Mengerjakan soal dalam bentuk praktikum | | Kriteria: Ketepatan dalam menerapkan konsep OOP  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mengerjakan soal praktikum | **15%** |
| 6 | Inheritance | 1. Pengertian Inheritance 2. Single dan Multilevel Inheritance 3. Super keyword 4. UML:relasi inheritance | | | | 1. Praktikum 2. Diskusi | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan konsep inheritance dengan dipandu jobsheet * Menggambarkan hubungan inheritance menggunakan diagram UML | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mengidentifikasi class yang memiliki relasi inheritance * Ketepatan membuat desain class yang memiliki relasi inheritance * Ketepatan dalam mempraktekkan instruksi pada jobsheet |  |
| 7 | Overriding dan Overloading | 1. Overriding 2. Overloading | | | | 1. Praktikum 2. Diskusi | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan overriding dan overloading dipandu dengan jobsheet | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mengidentifikasi method overriding dan overloading * Ketepatan dalam mempraktekkan instruksi pada jobsheet | 2.5% |
| **8** | **UTS** | Materi pertemuan 6-8 | | | | **Ujian Tulis/Online** | | 4x50’ | | **Mengerjakan soal dalam bentuk praktikum** | | Kriteria: Ketepatan dalam menerapkan konsep OOP  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mengerjakan soal praktikum | **30%** |
| 9 | Abstract Class dan Interface | 1. Konsep Abstract Class 2. Abstract method 3. UML:notasi abstract 4. Konsep Interface 5. Beda Interface dan Abstract Class 6. Pembuatan interface 7. Cara membuat class yang implements ke interface 8. UML:notasi interface dan relasi implements | | | | 1. Ceramah 2. Diskusi 3. Praktikum | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan konsep abstract class dan method serta interface dengan dipandu jobsheet * Menggambarkan hubungan abstract class dan interface menggunakan diagram UML | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam merancang interface dan abstract class * Ketepatan dalam pemodelannya dalam diagram class * Ketepatan dalam mempraktekkan instruksi pada jobsheet | 2.5% |
| 10 | Polimorfisme | 1. Konsep polimorfisme 2. heterogeneous collection 3. polymorphic arguments 4. virtual method invocation 5. casting object | | | | 1. Ceramah 2. Diskusi 3. Praktikum | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan konsep polimorfisme dengan dipandu jobsheet | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam menjelaskan penggunaan polimorfisme * Ketepatan dalam mempraktekkan instruksi pada jobsheet |  |
| **11** | GUI | 1. Frame, Menu, Textfield, Button, Label, Combobox, Radiobutton, Checkbox 2. Event Handling (actionperformed) | | | | 1. Ceramah 2. Diskusi 3. Praktikum | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan pembuatan GUI beserta komponen-komponennya dengan dipandu jobsheet * Praktek menerapkan event handling dengan dipandu jobsheet | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam membuat window * Ketepatan dalam membuat komponen-komponen yang akan ditambahkan ke dalamnya * Ketepatan dalam mempraktekkan instruksi pada jobsheet |  |
| 12 | Java API | 1. Java Docs 2. Collection 3. Database (JDBC) MySQL dengan GUI | | | | 1. Ceramah 2. Diskusi 3. Praktikum | | 4x50’ | | * Mendengarkan instruksi praktikum * Praktek menerapkan koneksi JAVA API Database (JDBC)untuk berinteraksi dengan database secara langsung dari class view (form) dengan dipandu jobsheet | | Kriteria: Ketepatan dalam mengikuti jobsheet praktikum  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam menjelaskan penggunaan dari Java Docs * Ketapatan dalam menjelaskan penggunaan dari JDBC * Ketepatan dalam mempraktekkan instruksi pada jobsheet |  |
| 13 | **Kuis 2** | Materi pertemuan 10-13 | | | | **Ujian Tulis/Online** | | 4x50’ | | **Mengerjakan soal dalam bentuk praktikum** | | Kriteria: Ketepatan dalam menerapkan konsep OOP  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam menyelesaikan soal praktikum | **15%** |
| 14 | GUI dan Database | 1. JDBC MySQL 2. CRUD dengan GUI 3. Model class, DAO class | | | | 1. Ceramah 2. Diskusi 3. Praktikum | |  | | * Mendengarkan instruksi praktikum * Praktek menerapkan JDBC dengan CRUD pada database sederhana dengan dipandu jobsheet * Menerapkan konsep class Model dalam menginterpretasikan data dalam bentuk entity dan class Data Access Object (DAO) untuk berinteraksi dengan database | | Kriteria: Ketepatan dalam menerapkan konsep OOP  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mempraktekkan instruksi pada jobsheet * Ketepatan menjawab pertanyaan dalam jobsheet |  |
| 15 | Unit Testing | 1. Unit Testing | | | | 1. Ceramah 2. Diskusi 3. Praktikum | |  | | * Mendengarkan instruksi praktikum * Praktek menerapkan Test Unit dengan dipandu jobsheet | | Kriteria: Ketepatan dalam menerapkan konsep OOP  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mempraktekkan instruksi pada jobsheet * Ketepatan menjawab pertanyaan dalam jobsheet |  |
| 16 | Tugas Besar | 1. Perancangan diagram class dari kasus yang diberikan | | | | Praktikum | | 4x50’ | | * Mengidentifikasi class dan menggambarkannya dalam diagram class * Praktek mengimplementasikan hasil rancangan kedalam program | | Kriteria: Ketepatan dalam mengembangkan proyek/tugas besar  Bentuk: penilaian aktifitas Praktek | | * Ketepatan dalam mengidentifikasi class * Ketepatan dalam penggambaran diagram class * Ketepatan dalam mempraktekkan instruksi pada jobsheet |  |
| **17** | **UAS** | **Materi mulai pertemuan 1-16** | | | | **Ujian Online** | | 4x50’ | | **Mengerjakan soal dalam bentuk praktikum** | | Kriteria: Ketepatan dalam menjawab pertanyaan  Bentuk: tes tulis/online | | * Ketepatan dalam menyelesaikan soal praktikum | **30%** |

**Keterangan :**

-