



POLITEKNIK NEGERI MALANG
JURUSAN AKUNTANSI
PROGRAM STUDI : D3 MANAJEMEN INFORMATIKA

RENCANA PEMBELAJARAN SEMESTER (RPS)

MATA KULIAH	KODE	RUMPUN MATA KULIAH	BOBOT (sks)/jam	SEMESTER	TGL. PENYUSUNAN
PEMROGRAMAN BERORIENTASI OBJEK	RIF 4	INFORMATIKA DASAR	2 sks / 3 jam	2	21 Februari 2017
OTORISASI	Dosen Pengembang RPS		Koordinator RMK	Ka PRODI	
	Imam Fahrur Rozi, ST., MT.		Mungki Astiningrum, ST., M.Kom.	Dr. Eng. Rosa Andrie A., ST., MT.	
Capaian Pembelajaran (CP)	Capaian Pembelajaran Lulusan Program Studi (CPL-Prodi)				
	Capaian Pembelajaran Lulusan yang dibebankan pada mata kuliah (CPL-MK)				
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 /	• Konsep OOP				

Pokok Bahasan		<ul style="list-style-type: none">• Class• Object• Enkapsulasi• Inheritance• Polimorfisme• Abstract Class• Interface• Java Basic Programming• Pengenalan GUI• Pengenalan Java API						
Pustaka		Utama :						
		Horstmann, C. S., & Cornell, G. (2007). <i>Core Java Volume I–Fundamentals, Eighth Edition</i> . Network Circle, Santa Clara: Prentice Hall.						
		Horstmann, C. S., & Cornell, G. (2008). <i>Core Java Volume II–Advanced Features, Eighth Edition</i> . Network Circle, Santa Clara: Prentice Hall.						
		Pendukung :						
		Rickyanto, I. (2005). <i>Dasar Pemrograman Berorientasi Objek dengan Java 2</i> . Yogyakarta: Andi Offset						
Media Pembelajaran		Software :		Hardware :				
		JDK 1.8 Netbeans 8		Komputer/Laptop dengan minilmal RAM 512 MB				
Nama Dosen Pengampu		Imam Fahrur Rozi, ST., MT.						
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	1. Pembahasan silabus dan	1. Ceramah	3x45’	• Mendengarkan ceramah	Kriteria:	• Mampu menyebutkan dengan	

		kontrak kuliah 2. Pengenalan Greenfoot 3. Simulasi pembuatan skenario permainan sederhana di Greenfoot	2. Diskusi 3. Simulasi		materi dari dosen dan berdiskusi <ul style="list-style-type: none"> Menyebutkan komponen/objek yang terlibat dalam suatu sistem Menyebutkan objek yang terlibat dalam skenario game sederhana yang akan disimulasikan menggunakan Greenfoot 	Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes Lisan	tepat objek world dan objek aktor yang terlibat dalam skenario game sederhana	
2	Konsep Dasar OOP	1. Pengenalan PBO 2. Perbedaan paradigma berorientasi objek dengan paradigma struktural 3. Konsep dasar PBO <ol style="list-style-type: none"> Class Object Enkapsulasi Inheritance Polimorfisme 4. Pengenalan pemodelan UML	1. Ceramah 2. Diskusi 3. Simulasi	3x45'	<ul style="list-style-type: none"> Mendengarkan ceramah materi dari dosen dan berdiskusi Menyebutkan komponen-komponen sistem (misal: Slakad) baik dari sudut pandang OOP maupun struktural Menyebutkan objek dan ciri-ciri tiap objek (apa yang bisa dilakukan dan apa yang dimiliki oleh objek) yang ada di suatu tempat/sistem Menyebutkan objek-objek yang memiliki kesamaan sifat dan merupakan turunan dari objek lain Menyebutkan objek yang bisa memiliki banyak bentuk 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes Lisan	<ul style="list-style-type: none"> Ketepatan dalam menjelaskan perbedaan antara paradigma OOP dan Struktural Ketepatan dalam menjelaskan konsep Class, object, enkapsulasi, inheritance dan polimorfisme Ketepatan dalam memberikan contoh dari masing-masing konsep dasar OOP 	

3	Class dan Object	<ol style="list-style-type: none"> 1. Class 2. Object 3. Atribut 4. Method 5. Instansiasi 6. Pengaksesan Atribut dan Method 7. UML:Diagram Class 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi dari dosen dan berdiskusi • Membentuk kelompok kecil dan berdiskusi dalam kelompok untuk mengidentifikasi objek (lengkap dengan atribut dan method) yang ada dalam suatu sistem yang ditentukan • Menggambarkan objek tersebut menggunakan diagram class • Mempresentasikan hasil diskusi dalam kelas 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes Tulis	<ul style="list-style-type: none"> • Ketepatan dalam mengidentifikasi objek • Ketepatan dalam menentukan atribut dan method dari tiap objek yang telah diidentifikasi • Ketepatan dalam penggambarannya ke dalam diagram class • Kemampuan dalam presentasi 	2.5%
4	Dasar Pemrograman Java	<ol style="list-style-type: none"> 1. Tipe data 2. Variabel 3. Seleksi Kondisi 4. Perulangan 5. Array 6. pengenalan try-catch (ArrayIndexOutOfBoundsException, ArithmeticException, NullPointerException) 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi dari dosen dan berdiskusi • Membentuk kelompok kecil dan menyelesaikan masalah sederhana dengan menggunakan konsep dasar pemrograman Java yang telah dijelaskan 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes Tulis	<ul style="list-style-type: none"> • Ketepatan dalam membuat program untuk menyelesaikan suatu kasus, dengan menggunakan konsep dasar pemrograman Java yang telah dijelaskan 	
5	Kuis 1	Materi pertemuan 1-4	Ujian Tulis/Online	2x45'	<ul style="list-style-type: none"> • Menjawab pertanyaan/soal Kuis 	<u>Kriteria:</u> Ketepatan dalam menjawab	<ul style="list-style-type: none"> • Ketepatan dalam menjawab pertanyaan 	15%

						pertanyaan <u>Bentuk:</u> tes Tulis/Online		
6	Enkapsulasi	<ol style="list-style-type: none"> 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 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Mensimulasikan cara kerja konstruktor • Menentukan modifier yang tepat untuk atribut dan method dari suatu class yang telah dibuat/diberikan sebelumnya • Berdiskusi tentang perbedaan antara atribut instansiasi dan atribut class, antara method instansiasi dan method class • Berdiskusi tentang kegunaan setter dan getter 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes lisan, tulis	<ul style="list-style-type: none"> • Ketepatan dalam 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 	
7	Relasi Class	<ol style="list-style-type: none"> 1. Relasi Has-A (Studi kasus 1 class memiliki hubungan has-a dengan 1 objek dari class lain) 2. Penggambaran relasi class dengan diagram class 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Menuliskan program dari diagram class yang telah diberikan • Menggambar desain diagram class dari kasus yang diberikan, dengan berdiskusi dalam kelompok • Presentasi hasil perancangan di kelas 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes lisan, tulis	<ul style="list-style-type: none"> • Ketepatan dalam menerjemahkan diagram class ke dalam program • Ketepatan dalam merancang diagram class dari suatu kasus tertentu • Kemampuan dalam presentasi 	2.5%

8	Relasi Class	<ol style="list-style-type: none"> 1. Relasi Has-A (Studi kasus 1 class memiliki hubungan has-a dengan lebih dari 1 objek dari class lain) 2. Penggambaran relasi class dengan diagram class 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Menuliskan program dari diagram class yang telah diberikan • Menggambarkan desain diagram class dari kasus yang diberikan, dengan berdiskusi dalam kelompok • Presentasi hasil perancangan di kelas 	<p><u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan</p> <p><u>Bentuk:</u> tes lisan, tulis</p>	<ul style="list-style-type: none"> • Ketepatan dalam menerjemahkan diagram class ke dalam program • Ketepatan dalam merancang diagram class dari suatu kasus tertentu 	
9	UTS	Materi pertemuan 6-8	Ujian Tulis/Online	2x45'	Menjawab pertanyaan/soal	<p><u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan</p> <p><u>Bentuk:</u> tes tulis/online</p>	<ul style="list-style-type: none"> • Ketepatan dalam menjawab pertanyaan 	30%
10	Inheritance	<ol style="list-style-type: none"> 1. Pengertian Inheritance 2. Single dan Multilevel Inheritance 3. Super keyword 4. UML:relasi inheritance 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Diskusi dalam kelompok untuk membuat contoh class-class yang memiliki relasi inheritance (single dan multilevel), dan menggambarkannya dalam diagram class • Mempresentasikan hasil diskusi di kelas 	<p><u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan</p> <p><u>Bentuk:</u> tes lisan, tulis</p>	<ul style="list-style-type: none"> • Ketepatan dalam mengidentifikasi class yang memiliki relasi inheritance • Ketepatan membuat desain class yang memiliki relasi inheritance • Kemampuan presentasi 	
11	Overriding dan Overloading	<ol style="list-style-type: none"> 1. Overriding 2. Overloading 	<ol style="list-style-type: none"> 1. Ceramah 2. Diskusi 3. Simulasi 	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Diskusi dalam kelompok untuk membuat contoh overriding dan overloading method 	<p><u>Kriteria:</u> Ketepatan dalam menjawab</p>	<ul style="list-style-type: none"> • Ketepatan dalam mengidentifikasi method overriding dan overloading • Kemampuan presentasi 	2.5%

					<ul style="list-style-type: none"> • Mempresentasikan hasil diskusi di kelas 	<p>pertanyaan</p> <p><u>Bentuk:</u> tes lisan, tulis</p>		
12	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. Simulasi	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Diskusi dalam kelompok untuk membuat contoh overriding dan overloading method • Mempresentasikan hasil diskusi di kelas 	<p><u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan</p> <p><u>Bentuk:</u> tes lisan, tulis</p>	<ul style="list-style-type: none"> • Ketepatan dalam merancang interface dan abstract class • Ketepatan dalam pemodelannya dalam diagram class • Kemampuan presentasi 	2.5%
13	Polimorfisme	1. Konsep polimorfisme 2. heterogeneous collection 3. polymorphic arguments 4. virtual method invocation 5. casting object	1. Ceramah 2. Diskusi 3. Simulasi	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Memperhatikan simulasi 	<p><u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan</p> <p><u>Bentuk:</u> tes lisan</p>	<ul style="list-style-type: none"> • Ketepatan dalam menjelaskan penggunaan polimorfisme 	
14	Kuis 2	Materi pertemuan 10-13	Ujian Tulis/Online	2x45'	Menjawab pertanyaan/soal	<p><u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan</p> <p><u>Bentuk:</u> tes tulis/online</p>	<ul style="list-style-type: none"> • Ketepatan dalam menjawab pertanyaan 	15%
15	GUI	1. Frame, Menu, Textfield, Button, Label, Combobox, Radiobutton, Checkbox	1. Ceramah 2. Diskusi 3. Simulasi	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Memperhatikan simulasi 	<p><u>Kriteria:</u> Ketepatan dalam</p>	<ul style="list-style-type: none"> • Ketepatan dalam membuat window 	

		2. Event Handling (actionperformed)			<ul style="list-style-type: none"> • Merancang suatu tampilan GUI • Membuat Window • Menambahkan komponen-komponen GUI lainnya ke dalam window 	menjawab pertanyaan <u>Bentuk:</u> tes tulis	<ul style="list-style-type: none"> • Ketepatan dalam membuat komponen-komponen yang akan ditambahkan ke dalamnya 	
16	Java API	1. Java Docs 2. String dan Collection (List, Set, Map)	1. Ceramah 2. Diskusi 3. Simulasi	3x45'	<ul style="list-style-type: none"> • Mendengarkan ceramah materi • Memperhatikan simulasi 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes lisan	<ul style="list-style-type: none"> • Ketepatan dalam menjelaskan penggunaan dari Java Docs • Ketepatan dalam menjelaskan penggunaan dari String dan Collection 	
17	Tugas Besar	1. Perancangan diagram class dari kasus yang diberikan	Diskusi	3x45'	<ul style="list-style-type: none"> • Mengidentifikasi class dan menggambarkannya dalam diagram class • Berkonsultasi dengan dosen tentang hasil perancangannya 	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes tulis	<ul style="list-style-type: none"> • Ketepatan dalam mengidentifikasi class • Ketepatan dalam penggambaran diagram class 	
18	UAS	Materi mulai pertemuan 1-16	Ujian Online	1x60'	Menjawab pertanyaan/soal	<u>Kriteria:</u> Ketepatan dalam menjawab pertanyaan <u>Bentuk:</u> tes tulis/online	<ul style="list-style-type: none"> • Ketepatan dalam menjawab pertanyaan 	30%
19	Remidi	Materi mulai pertemuan 1-16	Ujian Tulis	3x45'	Menjawab pertanyaan/soal	<u>Kriteria:</u> Ketepatan	<ul style="list-style-type: none"> • Ketepatan dalam menjawab 	

						dalam menjawab pertanyaan <u>Bentuk:</u> tes tulis/online	pertanyaan	
--	--	--	--	--	--	--	------------	--

Keterangan :

-