

**RENCANA PEMBELAJARAN SEMESTER**

Mata Kuliah	Kode MK	Rumpun MK	Semester	Bobot (sks)	Tgl Penyusunan
Algoritma dan Struktur Data	IFC12P4	Pemrograman Dasar	2	T=3 P=1	1 Desember 2022
<b>OTORISASI</b>	<b>Pengembang RPS</b>		<b>Koordinator RMK</b>		<b>Ketua Prodi</b>
	 Kharisma Monika Dian Pertiwi, S.Kom., M.Kom.		 Ahmad Wali Satria Bahari Johan, S.S.T., M. Kom.		 Muhammad Dzulfikar Fauzi, S.kom., M.Cs.
<b>Capaian Pembelajaran (CP)</b>	<b>CPL Prodi (Kode S, P, KU, KK)</b>				
	P.1	Menguasai konsep dan prinsip-prinsip matematika, sains dan sistem cerdas			
	P.4	Menguasai konsep dan prinsip-prinsip algoritma dan pemrograman			
	KU.2	Mampu bekerja secara mandiri dan bekerjasama dalam tim yang interdisiplin dan multidisiplin			
	KK.4	Mampu merancang dan menganalisa algoritma untuk menyelesaikan permasalahan secara efektif dan efisien menggunakan kaidah-kaidah pemrograman dan bahasa pemrograman yang sesuai			
	<b>CPMK (Kode M)</b>				
	M1	Mahasiswa mampu menguasai dasar-dasar pemrograman bahasa C/C++ (P1, P4)			
	M2	Mahasiswa mampu memahami dan mengimplementasikan abstract data type , struct dalam bahasa C/C++ (P4)			
	M3	Mahasiswa mampu memahami konsep dan mengimplementasikan struktur data linear (stack, queue) dalam bahasa pemrograman C/C++ (P4, KU2, KK4)			
	M4	Mahasiswa mampu memahami varian struktur data dinamis (linked-list) dan mengimplementasikan dalam bahasa C/C++ (P4, KU2, KK4)			
	M5	Mahasiswa mampu memahami konsep dan mengimplementasikan struktur data non linear (graph, tree) dalam bahasa pemrograman C/C++ (P4, KU2, KK4)			
	<b>SUB-CPMK (Kode L)</b>				
	L1	Mahasiswa mampu memahami konsep dasar pemrograman (iterative, conditional) dan mengimplementasikan dalam bahasa pemrograman C/C++. [M3][C2,C3]			
	L2	Mahasiswa mampu memahami konsep struktur data array dan mengimplementasikan dalam bahasa pemrograman C/C++. [M3][C2,C3]			
	L3	Mahasiswa mampu memahami konsep fungsi dan pointer serta mengimplementasikan dalam bahasa pemrograman C/C++. [M3][C2,C3]			
	L4	Mahasiswa mampu memahami konsep abstract data type dan struct serta mengimplementasikan dalam bahasa pemrograman C/C++ [M3][C2,C3]			
	L5	Mahasiswa mampu memahami konsep struktur data linier (stack,queue) dan mengimplementasikan dalam bahasa pemrograman C/C++ [M3][C2,C3]			
	L6	Mahasiswa mampu memahami konsep singly linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++. [M4][C2,C3]			
	L7	Mahasiswa mampu memahami konsep doubly linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++ [M4][C2,C3]			
	L8	Mahasiswa mampu memahami konsep circular linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++. [M4][C2,C3]			
	L9	Mahasiswa mampu memahami algoritma rekursif dan mengimplementasikan dalam bahasa pemrograman C/C++. [M4][C2,C3]			
	L10	Mahasiswa mampu memahami konsep struktur data non-linear (tree, graph) dan mengimplementasikan dalam bahasa pemrograman C/C++. [M3][C2,C3]			
<b>Deskripsi Singkat Mata Kuliah</b>	<b>Deskripsi</b>				
	Pada mata kuliah ini mahasiswa akan mempelajari berbagai macam struktur penyimpanan, penyusunan dan pengaturan banyak data serta algoritma terkait. Konsep abstraksi data dibahas untuk menentukan jenis struktur data secara linear maupun non-linear dengan contoh-contoh permasalahan.				
<b>Materi Pembelajaran/Pokok Bahasan</b>	<b>Bahan Kajian</b>				
	Algoritma dan Struktur Data				
	<b>Topik Bahasan</b>				
	1. konsep dasar pemrograman (iterative, conditional) dalam bahasa pemrograman C/C++				
	1.1. Pemrograman Iterative For				
	1.2. Pemrograman Conditional IF				
	2. Array 1 dan 2 Dimensi				
	2.1. Insert Array				
	2.2. Delete Data in Array				
	2.3. Search Data in Array				
	2.4. Edit Data in Array				
	3. konsep fungsi dan pointer dalam bahasa pemrograman C/C++				
	3.1. Fungsi Pass by Value dalam bahasa C				
	3.2. Fungsi Pass by Reference dalam bahasa C				
	3.3. Pointer dalam bahasa C				
	4. konsep abstract data type dan struct dalam bahasa pemrograman C/C++				
	4.1. konsep Abstract Data Type				
	4.2. Struct dalam bahasa C				
	4.3. Add, Edit Delete data struct				
	5. konsep struktur data linier (stack,queue) dalam bahasa pemrograman C/C++				
	5.1. konsep stack dalam bahasa pemrograman C				
	5.2. operasi-operasi stack dalam bahasa Pemrograman C				
	5.3. konsep queue dalam bahasa pemrograman C				
	5.4. operasi-operasi queue dalam bahasa Pemrograman C				
	6. konsep singly linked-list dan implementasi dalam bahasa pemrograman C/C++				
	6.1. konsep singly linked list				
	6.2. operasi-operasi singly linked-list				
	7. konsep doubly linked-list dan implementasi dalam bahasa pemrograman C/C++				
	7.1. konsep doubly linked list				
	7.2. operasi-operasi doublylinked-list				
	8. konsep circular linked-list dan implementasi dalam bahasa pemrograman C/C++				
	8.1. konsep singly linked list circular				
	8.2. operasi-operasi singly linked-list circular				
	8.3. konsep double linked list circular				
	8.4. operasi-operasi Double linked-list circular				
	9. algoritma rekursif dan implementasi dalam bahasa pemrograman C/C++				
	9.1. konsep rekursif				
	9.2. Implementasi algoritma rekursif dalam bahasa pemrograman C				
	10. konsep struktur data non-linear (tree, graph) dan implementasi dalam bahasa pemrograman C/C++				
	10.1. konsep struktur data graph				
	10.2. program untuk create, add print pada Graph				
	10.3. konsep Binary Tree				
	10.4. program untuk create dan insert pada Binary Tree				
	10.5. konsep tree Traversal				
	10.6. program Tree Traversal secara Pre-Order				
	10.7. In-Order Tree Traversal				

	10.8. Post-Order Tree Traversal
<b>Pustaka</b>	<b>Utama</b> 1. Data Structures and Program Design in C++ - Robert L. Kruse
	<b>Pendukung</b> 2. C and Data Structures by Practice - Ramesh Vasappanavara 3. C++ Data Structures 3rd ed - Nell Dale 4. Robert Sedgewick, Philippe Flajolet, "An Introduction to the Analysis of Algorithms 2ed", Addison-Wesley, New Jersey, 2013
<b>Media Pembelajaran</b>	<b>Software</b> PowerPoint, Code Blocks <b>Hardware</b> PC & LCD Projector
<b>Teacher/Team Teaching/Tim LS</b>	-
<b>Assessment</b>	Pengetahuan: Tes tulis (UTS, UAS), Psikomotorik: Kinerja (Tugas), Sikap: Observasi harian
<b>Mata Kuliah Syarat</b>	Algoritma dan Pemrograman

Perte muan Ke	Kemampuan Akhir yang direncanakan	Indikator Pencapaian Kompetensi	Materi Pokok	Bentuk dan Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Estimasi Waktu	Penilaian			Referensi
							Bentuk & Kriteria	Indikator Penilaian	Bobot (%)	
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
1	memahami konsep dasar pemrograman (iterative, conditional) dan mengimplementasikan dalam bahasa pemrograman C/C++	1.1. mengimplementasikan program dengan Perulangan iterative dalam bahasa pemrograman C/C++ 1.2. Mampu mengimplementasikan program dengan Conditional IF	1. konsep dasar pemrograman (iterative, conditional) dalam bahasa pemrograman C/C++ 1.1. Pemrograman Iterative For 1.2. Pemrograman Conditional IF	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	Mengimplementasikan dasar pemrograman dalam bahasa C	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan mengimplementasikan konsep dasar pemrograman (iterative, conditional) dalam bahasa C	10	1, 2
2	mengimplementasikan program pembuatan array 1 dimensi dan 2 dimensi dalam bahasa C	2.1. mengimplementasikan program insert data ke array dalam bahasa C 2.2. mengimplementasikan program delete data array dalam bahasa C 2.3. mengimplementasikan program search data in array dalam bahasa C 2.4. mengimplementasikan program edit data in array dalam bahasa C	2. Array 1 dan 2 Dimensi 2.1. Insert Array 2.2. Delete Data in Array 2.3. Search Data in Array 2.4. Edit Data in Array	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	Mengimplementasikan array 1 dimensi dan 2 dimensi dalam bahasa C	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan Mengimplementasikan array 1 dimensi dan 2 dimensi dalam bahasa C	10	1, 2
3	memahami konsep fungsi dan pointer serta mengimplementasikan dalam bahasa pemrograman C/C++	3.1. mengimplementasikan program Fungsi Pass by Value dalam bahasa C 3.2. mengimplementasikan program Fungsi Pass by Reference dalam bahasa C 3.3. mengimplementasikan program Pointer dalam bahasa C	3. konsep fungsi dan pointer dalam bahasa pemrograman C/C++ 3.1. Fungsi Pass by Value dalam bahasa C 3.2. Fungsi Pass by Reference dalam bahasa C 3.3. Pointer dalam bahasa C	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep fungsi dan pointer serta mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan memahami konsep fungsi dan pointer serta mengimplementasikan dalam bahasa pemrograman C/C++	10	1, 2
4	memahami konsep abstract data type dan struct serta mengimplementasikan dalam bahasa pemrograman C/C++	4.1. memahami konsep Abstract Data Type 4.2. mengimplementasikan program Membuat Struct dalam bahasa C 4.3. mengimplementasikan program Add, Edit Delete data struct dalam bahasa C	4. konsep abstract data type dan struct dalam bahasa pemrograman C/C++ 4.1. konsep Abstract Data Type 4.2. Struct dalam bahasa C 4.3. Add, Edit Delete data struct	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep abstract data type dan struct serta mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan memahami konsep abstract data type dan struct serta mengimplementasikan dalam bahasa pemrograman C/C++	10	1, 2
5,6	memahami konsep struktur data linier (stack, queue) dan mengimplementasikan dalam bahasa pemrograman C/C++	5.1. memahami konsep stack dan mengimplementasikan operasi-operasi stack dalam bahasa Pemrograman C 5.2. memahami konsep queue dan mengimplementasikan dalam bahasa pemrograman C 5.3. memahami konsep queue dan mengimplementasikan operasi-operasi queue dalam bahasa Pemrograman C 5.4. memahami konsep queue dan mengimplementasikan operasi-operasi queue dalam bahasa Pemrograman C	5. konsep struktur data linier (stack, queue) dalam bahasa pemrograman C/C++ 5.1. konsep stack dalam bahasa pemrograman C 5.2. operasi-operasi stack dalam bahasa Pemrograman C 5.3. konsep queue dalam bahasa pemrograman C 5.4. operasi-operasi queue dalam bahasa Pemrograman C	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep struktur data linier (stack, queue) dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan memahami konsep struktur data linier (stack, queue) dalam bahasa pemrograman C/C++	10	1, 2
7	memahami konsep singly linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	6.1. 6.2.	6. konsep singly linked-list dan implementasi dalam bahasa pemrograman C/C++ 6.1. konsep singly linked list 6.2. operasi-operasi singly linked-list	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep singly linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan dalam memahami konsep singly linked-list dan mengimplementasikan dalam	10	1, 2
8	UTS									
9	memahami konsep doubly linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	7.1. memahami konsep doubly linked list 7.2. mengimplementasikan operasi-operasi doubly linked-list	7. konsep doubly linked-list dan implementasi dalam bahasa pemrograman C/C++ 7.1. konsep doubly linked list 7.2. operasi-operasi doubly linked-list	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep doubly linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan dalam memahami konsep doubly linked-list dan mengimplementasikan dalam bahasa	10	1, 2
10,11	memahami konsep circular linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	8.1. memahami konsep singly linked list circular 8.2. mengimplementasikan operasi-operasi singly linked-list circular 8.3. memahami konsep double linked list circular 8.4. mengimplementasikan operasi-operasi Double linked-list circular	8. konsep circular linked-list dan implementasi dalam bahasa pemrograman C/C++ 8.1. konsep singly linked list circular 8.2. operasi-operasi singly linked-list circular 8.3. konsep double linked list circular 8.4. operasi-operasi Double linked-list circular	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep circular linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan dalam memahami konsep circular linked-list dan mengimplementasikan dalam bahasa pemrograman C/C++	10	1, 2
12	memahami algoritma rekursif dan mengimplementasikan dalam bahasa pemrograman C/C++	9.1. memahami konsep rekursif 9.2. mengimplementasikan algoritma rekursif dalam bahasa pemrograman C	9. algoritma rekursif dan implementasi dalam bahasa pemrograman C/C++ 9.1. konsep rekursif 9.2. Implementasi algoritma rekursif dalam bahasa pemrograman C	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami algoritma rekursif dan mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan dalam memahami algoritma rekursif dan mengimplementasikan dalam bahasa pemrograman C/C++	10	1, 2
13,14,15	memahami konsep struktur data non-linear (tree, graph) dan mengimplementasikan dalam bahasa pemrograman C/C++	10.1. memahami konsep struktur data graph 10.2. mengimplementasikan program untuk create, add print pada Graph	10. konsep struktur data non-linear (tree, graph) dan implementasi dalam bahasa pemrograman C/C++ 10.1. konsep struktur data graph	Bentuk: Kuliah dan Praktikum Terbimbing Metode: Ceramah, diskusi kelompok/kelas, praktikum, tanya jawab, presentasi	memahami konsep struktur data non-linear (tree, graph) dan mengimplementasikan dalam bahasa pemrograman C/C++	TM: 1x(4x50") BT: 1x(4x60") BM: 1x(4x60")	Tes: Tulis Pedoman Penskoran	Ketepatan dalam memahami konsep struktur data non-linear (tree, graph) dan mengimplementasikan	10	1, 2

		10.3. memahami konsep Binary Tree	10.2. program untuk create, add print pada Graph					mengimplementasikan dalam bahasa pemrograman C/C++	
		10.4. mengimplementasikan program untuk create dan insert pada Binary Tree	10.3. konsep Binary Tree						
		10.5. memahami konsep tree Traversal	10.4. program untuk create dan insert pada Binary Tree						
		10.6. mengimplementasikan program Tree Traversal secara Pre-Order	10.5. konsep tree Traversal						
		10.7. mengimplementasikan program Tree Traversal secara In-Order	10.6. program Tree Traversal secara Pre-Order						
		10.8. mengimplementasikan program Tree Traversal secara Post-Order	10.7. In-Order Tree Traversal						
			10.8. Post-Order Tree Traversal						

16	UAS								
Catatan :									
1. Capaian Pembelajaran Lulusan PRODI (CPL-PRODI) adalah kemampuan yang dimiliki oleh setiap lulusan PRODI yang merupakan internalisasi dari sikap, penguasaan pengetahuan dan ketrampilan sesuai dengan jenjang prodinya yang diperoleh melalui proses pembelajaran.									
2. CPL yang dibebankan pada mata kuliah adalah beberapa capaian pembelajaran lulusan program studi (CPL-PRODI) yang digunakan untuk pembentukan/pengembangan sebuah mata kuliah yang terdiri dari aspek sikap, ketrampilan umum, ketrampilan khusus dan pengetahuan.									
3. CP Mata kuliah (CPMK) adalah kemampuan yang dijabarkan secara spesifik dari CPL yang dibebankan pada mata kuliah, dan bersifat spesifik terhadap bahan kajian atau materi pembelajaran mata kuliah tersebut.									
4. Sub-CP Mata kuliah (Sub-CPMK) adalah kemampuan yang dijabarkan secara spesifik dari CPMK yang dapat diukur atau diamati dan merupakan kemampuan akhir yang direncanakan pada tiap tahap pembelajaran, dan bersifat spesifik terhadap materi pembelajaran mata kuliah tersebut.									
5. Kreteria Penilaian adalah patokan yang digunakan sebagai ukuran atau tolok ukur ketercapaian pembelajaran dalam penilaian berdasarkan indikator-indikator yang telah ditetapkan. Kreteria penilaian merupakan pedoman bagi penilai agar penilaian konsisten dan tidak bias. Kreteria dapat berupa kuantitatif ataupun kualitatif.									
6. Indikator penilaian kemampuan dalam proses maupun hasil belajar mahasiswa adalah pernyataan spesifik dan terukur yang mengidentifikasi kemampuan atau kinerja hasil belajar mahasiswa yang disertai bukti-bukti.									
Catatan tambahan:									
(1). Bobot SKS (P = Praktek; T= Teori).									
(2). TM: Tatap Muka; BT: Beban Tugas; BM: Belajar Mandiri.									
(3). 1 sks = (50' TM + 50' PT + 60' BM)/Minggu									
(4). Simbol-simbol elemen KKNI pada CPL-Prod: S = Sikap; KU = Ketrampilan Umum; KK = Ketrampilan Khusus; P = Pengetahuan									