	Universitas Negeri Surabaya **S1 Sains Data**	**Kode Dokumen**

RENCANA PEMBELAJARAN SEMESTER

MATA KULIAH (MK)	KODE	Rumpun MK	BOBOT (sks)	SEMESTER	Tgl Penyusunan
Teknik Optimasi	4920203074		**3**		22 November 2026

OTORISASI	Pengembang RPS	Koordinator RMK	Koordinator Program Studi
	Dr. Heri Purnawan, S.Si., M.Si.		YULIANI PUJI ASTUTI

Deskripsi Singkat MK	Mata kuliah Teknik Optimasi memperkenalkan konsep dasar dan metode optimasi untuk menyelesaikan masalah dalam domain sains data. Isi mata kuliah mencakup formulasi masalah optimasi, teknik optimasi linier (seperti Metode Simpleks, Dualitas, dan Analisis Sensitivitas) dan non-linier, model jaringan transportasi dan penugasan, Pemrograman Linier Bulat (Integer Linear Programming), Pemrograman Dinamis (Dynamic Programming), serta pengenalan algoritma metaheuristik dan kompleksitas komputasi. Tujuan mata kuliah ini adalah membekali mahasiswa dengan kemampuan untuk mengidentifikasi, memodelkan, dan menyelesaikan masalah optimasi secara terstruktur serta membandingkan pendekatan solusi eksak dan heuristik menggunakan Python. Ruang lingkup pembelajaran meliputi aplikasi dalam machine learning, pemrosesan data besar (big data), dan pengambilan keputusan berbasis data (data-driven decision making).	
Capaian Pembelajaran (CP)	**CPL-PRODI yang dibebankan pada MK**	
	CPL-3	Mengembangkan pemikiran logis, kritis, sistematis, dan kreatif dalam melakukan pekerjaan yang spesifik di bidang keahliannya serta sesuai dengan standar kompetensi kerja bidang yang bersangkutan
	CPL-21	Mampu mengambil keputusan secara tepat dalam konteks penyelesaian masalah di bidang sains data, berdasarkan hasil analisis informasi dan data
	CPL-27	Menguasai teori matematika, statistika, dan ilmu komputer/informatika
	CPL-23	Mampu mengelola data dan informasi dengan pendekatan model data dan sistem basis data yang tepat untuk kebutuhan organisasi dengan memperhatikan aspek keamanan dan integritas data
	Capaian Pembelajaran Mata Kuliah (CPMK)	
	CPMK-1	CPMK-1: Menerapkan konsep dasar optimasi dalam menyelesaikan masalah sederhana terkait sains data (C3).
	CPMK-2	CPMK-6: Menganalisis struktur masalah optimasi dengan kendala untuk mengidentifikasi pola dan strategi solusi (C4).
	CPMK-3	CPMK-2: Menganalisis karakteristik berbagai algoritma optimasi untuk menentukan pendekatan yang sesuai dengan konteks masalah (C4).
	CPMK-4	CPMK-4: Menciptakan solusi optimasi yang inovatif untuk masalah kompleks dalam domain sains data (C6).
	CPMK-5	CPMK-5: Menerapkan teknik optimasi linier dan non-linier dalam pemodelan masalah nyata (C3).
	CPMK-6	CPMK-6: Menganalisis struktur masalah optimasi dengan kendala untuk mengidentifikasi pola dan strategi solusi (C4).
	CPMK-7	CPMK-3: Mengevaluasi performa algoritma optimasi berdasarkan kriteria efisiensi dan akurasi dalam pengolahan data (C5).
	CPMK-8	CPMK-7: Mengevaluasi implementasi algoritma optimasi dalam sistem basis data dengan mempertimbangkan keamanan dan integritas data (C5).
	CPMK-9	CPMK-8: Menciptakan model optimasi yang terintegrasi dengan pipeline data untuk meningkatkan efisiensi analitik (C6).
	Matrik CPL - CPMK	

CPMK	CPL-3	CPL-21	CPL-27	CPL-23
CPMK-1	✔			
CPMK-2	✔			
CPMK-3		✔		
CPMK-4		✔		
CPMK-5			✔	
CPMK-6			✔	
CPMK-7				✔
CPMK-8				✔
CPMK-9				✔

Matrik CPMK pada Kemampuan akhir tiap tahapan belajar (Sub-CPMK)

CPMK	Minggu Ke															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
CPMK-1	✔	✔														
CPMK-2								✔								
CPMK-3			✔	✔												
CPMK-4									✔	✔						
CPMK-5											✔	✔				
CPMK-6						✔	✔									
CPMK-7					✔											
CPMK-8													✔	✔		
CPMK-9															✔	✔

Pustaka

Utama :

1. Boyd & Vandenberghe, Convex Optimization
2. SciPy Documentation, Optimization (scipy.optimize)
3. OpenStax, Contemporary Mathematics, bagian Linear Programming
4. Bombelli et al. (2024), From theORy to application: learning to optimize with Operations Research in an interactive way
5. NTU OpenCourseWare, Operations Research (2): Optimization Algorithms
6. IGNOU eGyanKosh, Linear Programming: Formulation and Graphical Method

Pendukung :

Mg Ke-	Kemampuan akhir tiap tahapan belajar (Sub-CPMK)	Penilaian		Bentuk Pembelajaran, Metode Pembelajaran, Penugasan Mahasiswa, [Estimasi Waktu]		Materi Pembelajaran	Bobot Penilaian (%)
		Indikator	Kriteria & Bentuk	Luring (*offline*)	*Daring (online)*		
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
1	Minggu 1: Mahasiswa mampu merumuskan komponen utama masalah optimasi sains data dalam bentuk variabel keputusan, fungsi tujuan, dan fungsi kendala.	1. Ketepatan menjelaskan definisi dan ruang lingkup teknik optimasi dalam domain sains data. 2. Ketepatan mengidentifikasi variabel keputusan dari deskripsi masalah nyata. 3. Keakuratan merumuskan fungsi tujuan (maksimasi/minimasi) dan fungsi-fungsi kendala matematis.	Kriteria: Ketepatan penjelasan konsep dasar, kelengkapan identifikasi variabel dan fungsi, serta kesesuaian formulasi model matematika dengan studi kasus sains data. Bentuk Penilaian: Tes tertulis (kuis) dan rubrik penilaian hasil pengerjaan tugas studi kasus.	Ceramah Interaktif & Diskusi Studi Kasus		Pengenalan Teknik Optimasi: Definisi, ruang lingkup aplikasi sains data, maksimasi vs minimasi, identifikasi fungsi tujuan dan kendala.	3%

2	Minggu 2: Mahasiswa mampu memformulasikan masalah nyata ke dalam model Pemrograman Linier (LP) dan menyelesaikannya menggunakan Metode Grafik (Pre-Simpleks).	1. Ketepatan memformulasikan masalah bisnis/sains data ke dalam model LP 2 variabel. 2. Ketelitian menggambar garis kendala dan mengidentifikasi Daerah Layak (Feasible Region). 3. Keakuratan menentukan titik-titik pojok (corner points) dan mencari nilai solusi optimal.	Kriteria: Ketepatan dalam mendefinisikan variabel keputusan, merumuskan fungsi tujuan dan kendala, ketelitian penggambaran daerah layak (feasible region), serta keakuratan penentuan solusi optimal melalui metode titik pojok. Bentuk Penilaian: Tes tertulis (uraian) dan rubrik unjuk kerja praktik pemodelan grafis.	Problem-Based Learning & Latihan Pemodelan Grafik		Pemrograman Linier Dasar (Pre-Simpleks): Formulasi model LP (variabel keputusan, fungsi tujuan, fungsi kendala), Daerah Layak (Feasible Region), Titik Pojok (Corner Point Method), dan Solusi Grafis 2 Variabel.	4%
3	Minggu 3: Mahasiswa mampu memformulasikan model Pemrograman Linier dan merancang penyelesaiannya menggunakan Metode Simpleks bentuk standar (maksimasi).	1. Ketepatan mengonversi model LP umum ke bentuk standar menggunakan variabel penyelang (slack). 2. Keakuratan menyusun tabel awal Simpleks. 3. Ketelitian melakukan iterasi Operasi Baris Elementer (OBE) hingga mencapai kriteria optimalitas.	Kriteria: Ketepatan dalam memformulasikan model matematika, kelengkapan penggunaan variabel penyelang, serta ketelitian prosedur iterasi tabel Simpleks hingga mencapai kondisi optimal. Bentuk Penilaian: Tes tertulis (uraian) dan rubrik penilaian proses teknis pengerjaan tabel.	Lecture & Praktikum Tabel Simpleks		Pemrograman Linier (LP) I: Formulasi bentuk standar, variabel penyelang (slack), perancangan tabel awal Simpleks, iterasi baris elementer, dan syarat optimalitas maksimasi.	4%
4	Minggu 4: Mahasiswa mampu mendeteksi dan menyelesaikan kondisi-kondisi khusus pada Metode Simpleks.	1. Ketepatan mengidentifikasi indikator kondisi solusi ganda (multiple optimal), unbounded, infeasible, dan degenerasi pada tabel Simpleks. 2. Keakuratan dalam memilih strategi intervensi matematis untuk menyelesaikan kasus degenerasi atau solusi khusus.	Kriteria: Ketepatan dalam mengidentifikasi jenis-jenis kondisi khusus (degenerasi, solusi optimal ganda, solusi tidak terbatas, dan solusi tidak layak) serta ketelitian prosedur matematis dalam menyelesaikan iterasi simpleks hingga mencapai solusi optimal yang valid. Bentuk Penilaian: Tes tertulis berupa soal uraian dan rubrik penilaian analitik terhadap hasil pengerjaan kasus.	Direct Instruction & Case Analysis		Pemrograman Linier (LP) II: Kasus-kasus khusus Simpleks (solusi tak tunggal/banyak, tanpa solusi/infeasible, solusi tak terbatas/unbounded, dan degenerasi).	4%
5	Minggu 5: Mahasiswa mampu memformulasikan dan menyelesaikan masalah Pemrograman Linier minimasi menggunakan variabel artifisial, Metode Big-M, dan Metode Dua Fase.	1. Ketepatan menambahkan variabel artifisial (artificial variables) pada kendala >= atau =. 2. Keakuratan menerapkan teknik penalti Big-M pada fungsi tujuan. 3. Ketepatan melakukan tahapan Metode Dua Fase (Two-Phase Method) hingga konvergensi tabel.	Kriteria: Ketepatan formulasi model matematis minimasi, ketelitian penggunaan variabel artifisial dan teknik penalti, serta ketajaman analisis konvergensi solusi pada setiap iterasi tabel simplex. Bentuk Penilaian: Tes tulis uraian dan rubrik penilaian prosedur pemecahan masalah.	Direct Instruction & Latihan Komputasi		Pemrograman Linier (LP) III: Masalah minimasi LP, penggunaan variabel artifisial, teknik penalti Metode Big-M, dan tahapan Metode Dua Fase (Two-Phase Method).	4%

6	Minggu 6: Mahasiswa mampu merumuskan model Dualitas dari masalah Primal LP dan menginterpretasikan makna ekonomis shadow price.	1. Ketepatan melakukan konversi model Primal ke model Dual (matriks transpose, pembatasan variabel). 2. Keakuratan menentukan nilai solusi Dual dari tabel optimal Primal. 3. Ketajaman menginterpretasikan nilai shadow price dalam konteks alokasi sumber daya.	Aktifitas Partisipasif	Interactive Class & Diskusi Komparatif		Teori Dualitas LP: Perumusan masalah Primal-Dual, konversi matriks koefisien transpose, dan interpretasi ekonomis shadow price pada alokasi sumber daya.	4%
7	Minggu 7: Mahasiswa mampu menganalisis dampak perubahan parameter masukan terhadap stabilitas solusi optimal LP.	1. Ketepatan menghitung interval toleransi perubahan koefisien fungsi tujuan (c_j). 2. Keakuratan menganalisis dampak perubahan konstanta ruas kanan (b_i) terhadap nilai fungsi tujuan. 3. Ketepatan menentukan batas kelayakan dan optimalitas solusi saat terjadi perubahan parameter.	Analytical Problem Solving & Latihan Kasus	Ceramah interaktif, studi kasus, diskusi kelompok, dan demonstrasi analisis masalah.		Analisis Sensitivitas LP: Variasi koefisien fungsi tujuan, perubahan konstanta ruas kanan, serta penetapan interval toleransi perubahan parameter.	4%
8	Minggu 8: Mahasiswa mampu mengevaluasi penyelesaian masalah Pemrograman Linier (formulasi, grafik, simpleks, dualitas, dan analisis sensitivitas) secara komprehensif melalui Ujian Tengah Semester (UTS).	1. Penguasaan konsep formulasi LP dan solusi grafis. 2. Keterampilan komputasi Metode Simpleks, Big-M, dan Dua Fase. 3. Kemampuan analisis Teori Dualitas dan Sensitivitas Parametrik.	Kriteria: Ketepatan dalam memformulasi model matematika, keakuratan prosedur perhitungan menggunakan metode grafik dan simpleks, serta ketajaman analisis pada konsep dualitas dan interpretasi analisis sensitivitas. Bentuk Penilaian: Tes tertulis (Ujian Tengah Semester) berupa soal uraian yang dilengkapi dengan rubrik penilaian analitik.	Ujian Tertulis Terstruktur		Evaluasi Komprehensif UTS: Cakupan materi perkuliahan Minggu ke-1 sampai dengan Minggu ke-7.	20%
9	Minggu 9: Mahasiswa mampu memformulasikan masalah transportasi dan menentukan solusi layak awal.	1. Ketepatan menyusun matriks transportasi berdasarkan data penawaran (supply) dan permintaan (demand). 2. Keakuratan menghitung solusi layak awal dengan metode Northwest Corner (NWC). 3. Keakuratan menghitung solusi awal dengan metode Minimum Cost dan Vogel's Approximation Method (VAM).	Tes	Case-Based Learning & Praktikum Distribusi		Model Jaringan Transportasi I: Formulasi matriks transportasi dan penentuan solusi awal menggunakan metode Northwest Corner (NWC), Minimum-Cost, dan Vogel's Approximation Method (VAM).	4%
10	Minggu 10: Mahasiswa mampu melakukan pengujian optimalitas pada tabel transportasi.	1. Ketepatan melakukan uji optimalitas menggunakan metode MODI (Modified Distribution). 2. Ketepatan melakukan Uji Stepping-Stone untuk mencari jalur tertutup (closed loop). 3. Keakuratan memperbarui alokasi tabel hingga diperoleh biaya distribusi minimum yang optimal.	Kriteria: Ketepatan prosedur pengujian optimalitas menggunakan metode MODI dan Stepping-Stone, efisiensi implementasi algoritma pada tools pemrograman, serta ketajaman analisis dan logika justifikasi hasil optimasi. Bentuk Penilaian: Praktik unjuk kerja (coding), presentasi hasil analisis, dan rubrik penilaian performa.	Problem Solving & Praktikum Uji Optimalitas		Model Jaringan Transportasi II: Uji optimalitas menggunakan metode Modified Distribution (MODI) dan metode Stepping-Stone.	4%

11	Minggu 11: Mahasiswa mampu memformulasikan masalah penugasan dan memahami transisinya menuju Pemrograman Linier Bulat.	1. Ketepatan menyusun matriks biaya/waktu penugasan (kasus seimbang maupun tidak seimbang/dummy). 2. Keakuratan menerapkan tahapan Algoritma Hungarian untuk alokasi optimal. 3. Ketepatan menjelaskan hubungan formulasi masalah penugasan sebagai bentuk khusus Integer Programming.	Kriteria: Ketepatan formulasi model matematika, keakuratan penerapan Algoritma Hungarian pada kasus seimbang maupun tak seimbang, serta kelogisan penjelasan transisi model menuju Pemrograman Linier Bulat. Bentuk Penilaian: Tes tertulis (kuis/ujian) dan rubrik penilaian tugas pemecahan masalah.	Problem-Based Learning & Latihan Kasus		Masalah Penugasan (Assignment Problem): Algoritma Hungarian (kasus seimbang & tak seimbang/dummy), transisi menuju Integer Programming.	4%
12	Minggu 12: Mahasiswa mampu memformulasikan dan menyelesaikan model Pemrograman Linier Bulat menggunakan teknik eksak.	1. Ketepatan mendefinisikan variabel keputusan integer. 2. Keakuratan menjelaskan keterbatasan metode pembulatan (Round-Off). 3. Ketepatan melakukan percabangan (tree branching) dan bounding pada Algoritma Branch and Bound (BnB).	Kriteria: Ketepatan formulasi model ILP, akurasi struktur pohon (tree branching) pada algoritma Branch and Bound, serta ketuntasan prosedur penyelesaian eksak. Bentuk Penilaian: Tes tertulis (Kuis) dan Rubrik Penilaian Jawaban.	Direct Instruction & Tree Branching Exercise		Integer Linear Programming (ILP): Keterbatasan pembulatan (Round-Off Method), Algoritma Branch and Bound (BnB).	4%
13	Minggu 13: Mahasiswa mampu menerapkan Pemrograman Dinamis untuk memecahkan masalah lintasan terpendek.	1. Ketepatan menguraikan masalah ke dalam struktur Stage (Tahap) dan State (Keadaan). 2. Keakuratan merumuskan persamaan rekursif (Bellman's Principle of Optimality). 3. Ketepatan melakukan komputasi rekursi maju (Forward) atau mundur (Backward) untuk menemukan lintasan terpendek.	Kriteria: Ketepatan dalam menerapkan Prinsip Optimalitas Bellman, kelengkapan tahapan rekursi (maju/mundur), serta akurasi hasil perhitungan penyelesaian masalah lintasan terpendek. Bentuk Penilaian: Tes tertulis dan rubrik penilaian analitik.	Lecture & Algorithmic Step-by-Step		Dynamic Programming (DP) I: Prinsip Optimalitas Bellman, rekursi maju/mundur (Forward/Backward Recursion), Shortest Route Problem.	4%
14	Minggu 14: Mahasiswa mampu menerapkan Pemrograman Dinamis untuk menyelesaikan masalah alokasi sumber daya berkendala kapasitas.	1. Ketepatan merumuskan model Knapsack Problem (0/1, Integer, dan Fractional). 2. Keakuratan menyusun tabel keputusan pada setiap stage alokasi sumber daya. 3. Ketepatan menentukan kombinasi barang/sumber daya yang memaksimalkan nilai dengan kendala kapasitas.	Kriteria: Ketepatan perumusan fungsi rekursif, keakuratan identifikasi stage dan state, serta presisi hasil akhir dalam penyelesaian model Knapsack (0/1, Integer, dan Fractional). Bentuk Penilaian: Tugas mandiri pemodelan dan rubrik penilaian.	Problem Solving & Exercise		Dynamic Programming (DP) II: Model Knapsack Problem (0/1 Knapsack, Integer Knapsack, Fractional Knapsack).	4%

15	Minggu 15: Mahasiswa mampu membedakan pendekatan eksak dan metaheuristik, memahami teori kompleksitas komputasi, serta menghitung persentase Optimality GAP menggunakan Python.	1. Ketepatan membandingkan karakteristik solusi eksak dan metaheuristik (GA, ACO, ALNS). 2. Ketepatan menjelaskan konsep kompleksitas komputasi (NP-Hard, NP-Complete, dan local optima). 3. Keakuratan menuliskan kode Python untuk menghitung persentase Optimality GAP.	Kriteria: Ketepatan dalam membedakan metode eksak dan metaheuristik, ketajaman analisis teori kompleksitas (NP-Hard/NP-Complete), serta keakuratan logika pemrograman dan hasil perhitungan Optimality GAP menggunakan Python. Bentuk Penilaian: Tes tertulis objektif (kuis) dan rubrik penilaian unjuk kerja (review kode Python).	Interactive Lecture & Practical Python Demonstration		Teori Kompleksitas Komputasi & Metaheuristik: Solusi Eksak vs Metaheuristik (GA, ACO, ALNS), konsep NP-Hard/NP-Complete, karakter solusi lokal optimal, dan perhitungan persentase Optimality GAP berbasis Python.	4%
16	Minggu 16: Evaluasi Akhir Perkuliahan (Ujian Akhir Semester, Presentasi Mini Project, demonstrasi kode Python dual-solver, dan penyerahan Laporan Proyek).	1. Kualitas integrasi pemodelan optimasi dengan pipeline data pada produk Mini Project. 2. Ketepatan implementasi dan validitas logika program Python dual-solver (Eksak vs Metaheuristik). 3. Kejelasan presentasi, kelengkapan sistematika Laporan Proyek, dan kemampuan menjawab pada sesi diskusi.	Kriteria: Ketepatan penyelesaian masalah pada UAS, validitas logika algoritma dual-solver (eksak dan metaheuristik), kelengkapan sistematika laporan, serta ketajaman analisis komparasi model optimasi. Bentuk Penilaian: Tes tertulis, rubrik unjuk kerja presentasi, demonstrasi kode program, dan portofolio laporan proyek.	Presentasi Proyek & Evaluasi Akhir		Evaluasi Akhir Perkuliahan: Ujian Akhir Semester, Presentasi Mini Project, demonstrasi kode Python dual-solver (Eksak vs Metaheuristik), dan penyerahan Laporan Proyek.	25%

SDGS (Sustainable Development Goals)



Catatan

1. **Capaian Pembelajaran Lulusan Program Studi (CPL-Prodi)** adalah rumusan kemampuan yang harus dimiliki oleh lulusan program studi yang mencakup sikap, penguasaan pengetahuan, keterampilan umum, dan keterampilan khusus sesuai dengan jenjang kualifikasi, yang diperoleh melalui proses pembelajaran.
2. **CPL-Prodi yang dibebankan pada Mata Kuliah** adalah bagian dari CPL-Prodi yang dipilih dan relevan untuk dikembangkan melalui suatu mata kuliah, sebagai dasar perumusan capaian pembelajaran mata kuliah.
3. **Capaian Pembelajaran Mata Kuliah (CPMK)** adalah kemampuan yang dirumuskan secara spesifik dan diturunkan dari CPL-Prodi yang dibebankan pada mata kuliah, serta menggambarkan kompetensi yang harus dicapai mahasiswa setelah menyelesaikan mata kuliah.
4. **Sub-CPMK** adalah kemampuan akhir yang direncanakan pada setiap tahap pembelajaran, sebagai penjabaran lebih rinci dari CPMK, bersifat terukur dan dapat diamati sesuai dengan materi pembelajaran.
5. **Indikator Penilaian** adalah pernyataan spesifik dan terukur yang digunakan untuk menilai ketercapaian Sub-CPMK, baik dari proses maupun hasil belajar mahasiswa, yang didukung dengan bukti penilaian.
6. **Kriteria Penilaian** adalah tolok ukur yang digunakan untuk menentukan tingkat ketercapaian hasil belajar mahasiswa berdasarkan indikator penilaian, guna menjamin penilaian yang objektif, konsisten, dan adil. Kriteria penilaian dapat bersifat kuantitatif dan/atau kualitatif.
7. **Bentuk Penilaian** meliputi penilaian tes dan non-tes sesuai dengan karakteristik capaian pembelajaran.
8. **Bentuk Pembelajaran** meliputi kuliah, responsi, tutorial, seminar atau bentuk lain yang setara, praktikum, praktik studio, praktik bengkel, praktik lapangan, penelitian, dan pengabdian kepada masyarakat.
9. **Metode Pembelajaran** merupakan cara penyampaian pembelajaran yang digunakan untuk mencapai capaian pembelajaran, antara lain Small Group Discussion, Project Based Learning, Case Based Learning, Collaborative Learning, Discovery Learning, Self-Directed Learning, serta metode lain yang relevan.
10. **Materi Pembelajaran** adalah bahan kajian yang disusun secara sistematis dalam bentuk pokok dan sub-pokok bahasan untuk mendukung pencapaian Sub-CPMK.
11. **Bobot Penilaian** adalah persentase kontribusi setiap komponen penilaian terhadap nilai akhir mata kuliah, yang ditetapkan secara proporsional sesuai tingkat kompleksitas dan ketercapaian Sub-CPMK, dengan total keseluruhan sebesar 100%.
12. TM = Tatap Muka, PT = Penugasan Terstruktur, BM = Belajar Mandiri.