Mata kuliah: ALGORITMA DAN STRUKTUR DATA (ITH1B4) / 4 sks / Semester II

**CAPAIAN PEMBELAJARAN LULUSAN PRODI:**

PO9: Kemampuan mendesain dan melaksanakan eksperimen laboratorium dan/atau lapangan serta menganalisis dan mengartikan data untuk memperkuat penilaian dalam bidang informatika.

PO10: Kemampuan mengidentifikasi, merumuskan, menganalisis, dan menyelesaikan permasalahan dalam bidang informatika.

PO11: Kemampuan menerapkan metode, ketrampilan, dan/atau piranti teknik informatika yang terbaru yang diperlukan untuk praktek bidang informatika.

Mampu menentukan dan mengimplementasikan jenis linked list yang tepat suatu kasus pemrograman  
[CLO1 CLO3]

Mampu membuat algoritma dan mengimplementasikan struktur data tree [CLO3]

Mampu membuat algoritma dan mengimplementasikan struktur data graph [CLO3]

Mampu membuat algoritma dan mengimplementasikan struktur data stack [CLO2]

Mampu membuat algoritma dan mengimplementasikan struktur data queue [CLO2]

Mampu membuat algoritma dan mengimplementasikan struktur data circular linked list [CLO1]

Mampu membuat algoritma dan mengimplementasikan struktur data double linked list [CLO1]

Mampu membuat algoritma dan mengimplementasikan struktur data linked list [CLO1]

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **TELKOM UNIVERSITY**  **FAKULTAS INFORMATIKA**  **JURUSAN/PROGRAM STUDI S1 TEKNOLOGI INFORMASI** | | | | | | | | **Kode Dokumen** | |
| **RENCANA PEMBELAJARAN SEMESTER** | | | | | | | | | | |
| **MATA KULIAH (MK)** | | **KODE** | | **Rumpun MK** | | **BOBOT (sks)** | | **SEMESTER** | | **Tgl Penyusunan** |
| **Algoritma dan Struktur Data** | | **ITH1G4** | | Kosongkan saja | | **T=3** | **P=1** | 2 | | 17 Oktober 2017 |
| **OTORISASI** | | **Pengembang RPS** | | | **Koordinator RMK** | | | **Ketua PRODI** | | |
| Untari Novia Wisesty, M.T. | | | Kosongkan saja | | | Endro Ariyanto, S.T., M.T. | | |
| **Capaian Pembelajaran (CP)** | **PLO** |  | | | | | | | | |
| Kosongkan saja | | | | | | | | | |
| **CLO** | |  | | | | | | | |
| 1. Mahasiswa mampu membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. 2. Mahasiswa mampu membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. 3. Mahasiswa mampu membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman | | | | | | | | | |
| **Diskripsi Singkat MK** | Pada mata kuliah ini diajarkan berbagai macam struktur data yang dapat diimplementasikan dalam program komputer, serta bagaimana memilih struktur data yang tepat untuk suatu kasus tertentu. Materi yang diajarkan meliputi: Abstract Data Type (ADT), representasi struktur data linier dan primitif-primitifnya (array, linked list, variasi linked list, stack, queue), representasi struktur data non- linier dan primitif-primitifnya (tree, graph, multilist), dan algorima penelusuran (preorder, inorder, postorder) dan pencarian (BFS, DFS). | | | | | | | | | |
| **Bahan Kajian /**  **Pokok Bahasan** | Adapun bahan kajian atau pokok bahasannya adalah:   * Linear Linked List (termasuk Stack dan Queue) * Circular Linked List * Double Linked List * Non Linear Linked List (termasuk Tree dan Graph) | | | | | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Pustaka** | **Utama:** |  | |
| Diktat Kuliah IF2181 Struktur Data, Inggriani Liem, ITB, 2003. | | |
| **Pendukung:** |  | |
| Standish, Thomas A. Data Structures, Algorithms, & Software Principles in C. Addison wesley Publishing Company 1995.  AHO, Alfred V., John E. Hopcroft, Jeffrey D. Ullman. Data Structures and Algorithm. Addison Wesley Publishing Company. 1987. | | |
| **Media Pembelajaran** | **Perangkat lunak** | | **Perangkat keras** |
| OS: Windows; MS Office; Pdf reader; C++ compiler (Code::Block) | | Komputer/Laptop |
| **Dosen Pengampu** | DJN; ASP | | |
| **Matakuliah syarat** | Dasar Algoritma dan Pemrograman. | | |

| **Mg**  **Ke-** | **Kemampuan akhir yang diharapkan (Sub CLO)** | **Indikator Penilaian** | **Kriteria & Bentuk Penilaian** | **Bentuk/Metode Pembelajaran & Penugasan Mahasiswa**  **[Estimasi Waktu]** | **Materi Pembelajaran**  **[Pustaka]** | **Bobot Penilaian (%)** |
| --- | --- | --- | --- | --- | --- | --- |
| **(1)** | **(2)** | **(3)** | **(4)** | **(5)** | **(6)** | **(7)** |
| **1-4** | Mampu membuat algoritma dan mengimplementasikan struktur data linked list [CLO1]. | Ketepatan pembangunan primitif dasar insert dan delete Linear list.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UTS | * Kuliah dan Diskusi: 4x50’x3 = 600’ * Praktikum: 2x100’x1=200’ * Tugas 1: 4x50’x1=200’ | * Pengertian dan kegunaan Abstrak Data Type, Pointer, * Pengertian dan Algoritma Linear List * Implementasi Algoritma Linear List pada bahasa pemrograman C++ | UTS = 15;  Praktikum = 3; |
| **5** | Mampu membuat algoritma dan mengimplementasikan struktur data double linked list [CLO1] | Ketepatan pembangunan primitif dasar insert dan delete Double Linked List.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UTS | * Kuliah dan Diskusi: 1x50’x3 = 150’ * Praktikum: 1x100’x1=100’ * Tugas 1: 1x50’x1=50’ | * Pengertian dan Algoritma Double List * Implementasi Algoritma Double List pada bahasa pemrograman C++ * Implementasi Algoritma Double List pada suatu kasus | UTS = 5; |
| **6** | Mampu membuat algoritma dan mengimplementasikan struktur data circular linked list [CLO1] | Ketepatan pembangunan primitif dasar insert dan delete Circular Linked List.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UTS | * Kuliah dan Diskusi: 1x50’x3 = 150’ * Praktikum: 1x100’x1=100’ * Tugas 1: 1x50’x1=50’ | * Pengertian dan Algoritma Circular List * Implementasi Algoritma Circular List pada bahasa pemrograman C++ * Implementasi Algoritma Circular List pada suatu kasus | UTS = 5; |
| **7** | Mampu membuat algoritma dan mengimplementasikan struktur data linked list with last [CLO1] | Ketepatan pembangunan primitif dasar insert dan delete linked list with last.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UTS | * Kuliah dan Diskusi: 1x50’x3 = 150’ * Praktikum: 1x100’x1=100’ * Tugas 1: 1x50’x1=50’ | * Pengertian dan Algoritma Linked List with Last * Implementasi Algoritma Linked List with Last pada bahasa pemrograman C++ * Implementasi Algoritma Linked List with Last pada suatu kasus | UTS = 5;  Praktikum = 3 (gabungan materi minggu 5,6,7);  Tugas 1 = 6; |
| **UJIAN TENGAH SEMESTER** | | | | | | |
| **8** | Mampu membuat algoritma dan mengimplementasikan struktur data stack [CLO2] | Ketepatan pembangunan primitif dasar insert dan delete struktur data Stack.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UAS | * Kuliah dan Diskusi: 1x50’x3 = 150’ * Praktikum: 1x100’x1=100’ * Tugas 2: 1x50’x1=50’ | * Pengertian dan Algoritma Stack menggunakan Array * Implementasi algoritma Stack pada bahasa pemrograman C++ | UAS = 5; |
| **9-10** | Mampu membuat algoritma dan mengimplementasikan struktur data Queue [CLO2] | Ketepatan pembangunan primitif dasar insert dan delete struktur data Queue.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UAS | * Kuliah dan Diskusi: 2x50’x3 = 300’ * Praktikum: 1x100’x1=100’ * Tugas 2: 2x50’x1=100’ | * Pengertian dan Algoritma Queue menggunakan Array * Implementasi algoritma Queue pada bahasa pemrograman C++ | UAS = 5;  Praktikum = 3 (gabungan materi minggu 8, 9, 10);  Tugas 2 = 4; |
| **11-12** | Mampu membuat algoritma dan mengimplementasikan struktur data Tree [CLO3] | Ketepatan pembangunan primitif dasar insert dan delete struktur data Binary Search Tree.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UAS | * Kuliah dan Diskusi: 2x50’x3 = 300’ * Praktikum: 2x100’x1=200’ * Tugas 3: 2x50’x1=100’ | * Review algoritma rekursif * Pengertian Tree beserta implementasi algoritma penelusurannya (preorder, inorder, postorder) * Pengertian dan Algoritma Binary Search Tree * Implementasi Algoritma Binary Search Tree pada bahasa pemrograman C++ * Implementasi Binary Search Tree pada suatu kasus | UAS = 12;  Praktikum = 3; |
| **13-14** | Mampu membuat algoritma dan mengimplementasikan struktur data Graph [CLO3] | Ketepatan pembangunan primitif dasar Graph.  Ketepatan implementasi algoritma dalam bahasa pemrograman. | **Kriteria penilaian:**  Ketepatan dan penguasaan  Ketepatan implementasi algoritma  **Bentuk penilaian:**  Non-test: Tugas, Praktikum  Test: UAS | * Kuliah dan Diskusi: 2x50’x3 = 300’ * Praktikum: 1x100’x1=100’ * Tugas 3: 2x50’x1=100’ | * Pengertian dan Algoritma Graph * Pengertian dan implementasi algoritma penelusuran Graf (BFS dan DFS) * Implementasi Algoritma Graph pada bahasa pemrograman C++ * Implementasi Graph pada suatu kasus | UAS = 8;  Praktikum = 3;  Tugas 3 = 10; |
| **UJIAN AKHIR SEMESTER** | | | | | | |

**Catatan:**

1. **Capaian Pembelajaran Lulusan PRODI (PLO)** 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. **PLO yang dibebankan pada mata kuliah** adalah beberapa capaian pembelajaran lulusan program studi yang digunakan untuk pembentukan/pengembangan sebuah mata kuliah yang terdiri dari aspek sikap, ketrampilan umum, ketrampilan khusus dan pengetahuan.
3. **CLO Mata kuliah** adalah kemampuan yang dijabarkan secara spesifik dari PLO yang dibebankan pada mata kuliah, dan bersifat spesifik terhadap bahan kajian atau materi pembelajaran mata kuliah tersebut.
4. **Kemampuan akhir yang diharapkan (CLO) Mata kuliah** adalah kemampuan yang dijabarkan secara spesifik dari CLO 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. **Kriteria Penilaian** adalah patokan yang digunakan sebagai ukuran atau tolok ukur ketercapaian pembelajaran dalam penilaian berdasarkan indikator-indikator yang telah ditetapkan. Kriteria penilaian merupakan pedoman bagi penilai agar penilaian konsisten dan tidak bias. Kriteria 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.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mapping Student Category** | | | | |
| **TEL U INDEX** | **TEL U RANGE** | **TEL U CATEGORY** | | **ABET/JABEE** |
| A | 80.01 ke atas | Istimewa | Excellent | Exemplary |
| AB | 70.01 - 80.00 | Baik Sekali | Very Good | Proficient |
| B | 65.01 - 70.00 | Baik | Good |  |
| BC | 60.01 - 65.00 | Cukup Baik | Fair | Apprentice |
| C | 50.01 - 60.00 | Cukup | Satisfactory |  |
| D | 40.01 - 50.00 | Kurang | Passing | Novice |
| E | 40.00 ke bawah | Sangat Kurang | Poor | Unacceptable |

| **Course Learning Outcomes**  **(CLO)** | **Indicator Scales and Score Ranges** | | | | |
| --- | --- | --- | --- | --- | --- |
| **Unacceptable** | **Novice** | **Apprentice** | **Proficient** | **Exemplary** |
| **0 (E)** | **1 (D)** | **2 (BC,C)** | **3 (AB, B)** | **4 (A)** |
| **0 - 40** | **41 - 50** | **51 - 65** | **66 - 80** | **81 - 100** |
| **CLO 1:**  Mahasiswa mampu membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. | * **Tidak mampu** membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. | * **Kurang mampu** membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. | * **Cukup mampu** membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. | * **Mampu** membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. | * **Sangat mampu** membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman. |
| **CLO 2:**  Mahasiswa mampu membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Tidak mampu** membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Kurang mampu** membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Cukup mampu** membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Mampu** membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Sangat mampu** membangun algoritma untuk primitif struktur data Stack dan Queue, serta mengimplementasikannya dalam Bahasa Pemrograman. |
| **CLO 3:**  Mahasiswa mampu membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Tidak mampu** membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Kurang mampu** membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Cukup mampu** membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Mampu** membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman. | * **Sangat mampu** membangun algoritma untuk primitive struktur data non-linear Linked List, Graph dan Tree, serta mengimplementasikannya dalam Bahasa Pemrograman. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Mapping Pembobotan Nilai** | | | | | |
| **Komponen Nilai** | **CLO 1** | **CLO 2** | **CLO 3** | **Total per Komponen Nilai** |
| **UTS** | 30 |  |  | **30** |
| **UAS** |  | 10 | 20 | **30** |
| **Tugas 1** | 7 |  |  | **20** |
| **Tugas 2** |  | 4 |  |
| **Tugas 3** |  |  | 10 |
| **Praktikum** | 6 | 3 | 6 | **15** |
| **Presensi** | 1 | 2 | 2 | **5** |
| **Total per Komponen Nilai** | **44** | **19** | **38** | **100** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **RENCANA TUGAS MAHASISWA** | | | | | |
| Mata Kuliah | **ALGORITMA DAN STRUKTUR DATA** | | | | |
| Kode | **ITH1G4** | SKS | 4 | SEMESTER | 2 |
| Dosen Pengampu |  | | | | |
| **BENTUK TUGAS** | | | | | |
| **Tugas 1:** Membangun Algoritma dan Mengimplementasikan Linier Linked List dalam Bahasa C++. | | | | | |
| **JUDUL TUGAS** | | | | | |
| Implementasi Linier Linked List. | | | | | |
| **SUB CAPAIAN PEMBELAJARAN** | | | | | |
| Mahasiswa mampu membangun algoritma untuk primitif struktur data Linked List dan mengimplementasikannya dalam Bahasa Pemrograman (CLO 1) | | | | | |
| **DESKRIPSI TUGAS** | | | | | |
| Tuliskan semua primitive (prosedur) beserta main program dari list yang berisi data mahasiswa (NIM, Nama, Nilai) yang terurut secara descending berdasarkan nilainya dalam bahasa C++. Prosedur yang perlu dibangun yaitu:   1. Insert 1 mahasiswa, dimana isi data list tetap terurut secara descending. 2. Delete 1 mahasiswa berdasarkan NIM yang ingin didelete, dimana isi data list tetap terurut secara descending. 3. Edit nilai mahasiswa berdasarkan NIM, dimana isi data list tetap terurut secara descending. 4. Mencari data mahasiswa dan menampilkan datanya berdasarkan NIM 5. Memecah (split) List mahasiswa ke L1 (berisi mahasiswa dengan NIM ganjil) dan L2 (berisi mahasiswa dengan NIM genap). 6. Menggabungkan (merge) isi list L1 dan L2 ke dalam L3 (isi list L3 harus terurut secara descending). 7. View isi list.   Jenis Linier Linked List yang digunakan yaitu:   1. Double Linked List 2. Circular List 3. Single List with Last 4. Double Linked List with Last 5. Circular List with Last | | | | | |
| **METODE PENGERJAAN TUGAS** | | | | | |
| 1. Yang harus dikerjakan dan batasan-batasan: 2. Membuat kelompok yang terdiri dari maksimal 2 mahasiswa. 3. Jenis Linier Linked List yang digunakan oleh setiap kelompok ditentukan oleh dosen yang bersangkutan. 4. Membangun dan mengimplementasikan primitive-primitif dalam deskripsi tugas dalam Bahasa C++. 5. Tahapan Pengerjaan: 6. Merancang algoritma dari primitive-primitif dan main program yang akan diimplementasikan. 7. Mengimplementasikan primitive-primitif dan main program yang sudah dirancang dalam Bahasa C++. 8. Memastikan program yang dibangun dapat berjalan (running) sesuai deskripsi tugas. 9. Menulis laporan tugas. | | | | | |
| **BENTUK DAN FORMAT LUARAN** | | | | | |
| 1. Source code program C++. 2. Laporan tugas yang terdiri dari permasalahan yang dihadapi, gambaran struktur data beserta penjelasannya, rancangan algoritma, dan print screen output dari setiap primitive yang dibangun. 3. Pada laporan tugas harus dituliskan Kelas, Nomor kelompok, Nama lengkap, NIM, peran dari setiap anggota kelompok, dan tanda tangan mahasiswa yang ikut mengerjakan. 4. Source Code dan Laporan tugas dikumpulkan pada waktu dan tempat yang telah ditentukan oleh dosen. | | | | | |
| **INDIKATOR, KRITERIA DAN BOBOT PENILAIAN** | | | | | |
| 1. Source code program: 70%. 2. Laporan tugas: 30%. 3. Bagi yang melakukan kecurangan/plagiarism nilainya akan diturunkan sesuai dengan tingkat plagiarismenya. | | | | | |
| **JADWAL PELAKSANAAN** | | | | | |
| Tugas diberikan pada Minggu-1 perkuliahan.  Batas akhir pengumpulan pada Minggu-6 perkuliahan. | | | | | |
| **LAIN-LAIN** | | | | | |
| Bobot nilai **Tugas 1** adalah **9%** dari total bobot mata kuliah Algoritma dan Struktur Data. | | | | | |
| **DAFTAR RUJUKAN** | | | | | |
| Diktat Kuliah IF2181 Struktur Data, Inggriani Liem, ITB, 2003. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **RENCANA TUGAS MAHASISWA** | | | | | |
| Mata Kuliah | **ALGORITMA DAN STRUKTUR DATA** | | | | |
| Kode | **ITH1G4** | SKS | 4 | SEMESTER | 2 |
| Dosen Pengampu |  | | | | |
| **BENTUK TUGAS** | | | | | |
| **Tugas 2:** Membangun Algoritma dan Mengimplementasikan Priority Queue/Stack dalam Bahasa C++. | | | | | |
| **JUDUL TUGAS** | | | | | |
| Implementasi Priority Queue/Stack. | | | | | |
| **SUB CAPAIAN PEMBELAJARAN** | | | | | |
| Mahasiswa mampu membangun algoritma untuk primitif struktur data Priority Queue/Stack dan mengimplementasikannya dalam Bahasa Pemrograman C++ (CLO 2). | | | | | |
| **DESKRIPSI TUGAS** | | | | | |
| Priority Queue/Stack merupakan struktur data Queue atau Stack dimana isi datanya terurut berdasarkan prioritasnya. Data dengan prioritas tertinggi akan dilayani terlebih dahulu dibandingkan dengan data dengan prioritas yang lebih rendah. Implementasikan primitive-primitif dalam Priority Queue/Stack tersebut dalam Bahasa C++. Prosedur yang perlu dibangun yaitu:   1. Insert 1 data, dimana isi data tetap terurut berdasarkan prioritasnya. 2. Delete 1 data. 3. View isi stack/queue.   Jenis struktur data yang digunakan yaitu:   1. Stack, dengan representasi kontigu. 2. Queue alternative I, dengan representasi kontigu. 3. Queue alternative II, dengan representasi kontigu. 4. Queue alternative III, dengan representasi kontigu. | | | | | |
| **METODE PENGERJAAN TUGAS** | | | | | |
| 1. Yang harus dikerjakan dan batasan-batasan: 2. Membuat kelompok yang terdiri dari maksimal 2 mahasiswa. 3. Jenis struktur data yang digunakan oleh setiap kelompok ditentukan oleh dosen yang bersangkutan. 4. Membangun dan mengimplementasikan primitive-primitif dalam deskripsi tugas dalam Bahasa C++. 5. Tahapan Pengerjaan: 6. Merancang algoritma dari primitive-primitif dan main program yang akan diimplementasikan. 7. Mengimplementasikan primitive-primitif dan main program yang sudah dirancang dalam Bahasa C++. 8. Memastikan program yang dibangun dapat berjalan (running) sesuai deskripsi tugas. 9. Menulis laporan tugas. | | | | | |
| **BENTUK DAN FORMAT LUARAN** | | | | | |
| 1. Source code program C++. 2. Laporan tugas yang terdiri dari permasalahan yang dihadapi, gambaran struktur data beserta penjelasannya, rancangan algoritma, dan print screen output dari setiap primitive yang dibangun. 3. Pada laporan tugas harus dituliskan Kelas, Nomor kelompok, Nama lengkap, NIM, peran dari setiap anggota kelompok, dan tanda tangan mahasiswa yang ikut mengerjakan. 4. Source Code dan Laporan tugas dikumpulkan pada waktu dan tempat yang telah ditentukan oleh dosen. | | | | | |
| **INDIKATOR, KRITERIA DAN BOBOT PENILAIAN** | | | | | |
| 1. Source code program: 70%. 2. Laporan tugas: 30%. 3. Bagi yang melakukan kecurangan/plagiarism nilainya akan diturunkan sesuai dengan tingkat plagiarismenya. | | | | | |
| **JADWAL PELAKSANAAN** | | | | | |
| Tugas diberikan pada Minggu-1 perkuliahan setelah UTS.  Batas akhir pengumpulan pada Minggu-5 perkuliahan setelah UTS. | | | | | |
| **LAIN-LAIN** | | | | | |
| Bobot nilai **Tugas 2** adalah **5%** dari total bobot mata kuliah Algoritma dan Struktur Data. | | | | | |
| **DAFTAR RUJUKAN** | | | | | |
| Diktat Kuliah IF2181 Struktur Data, Inggriani Liem, ITB, 2003. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **RENCANA TUGAS MAHASISWA** | | | | | |
| Mata Kuliah | **ALGORITMA DAN STRUKTUR DATA** | | | | |
| Kode | **ITH1G4** | SKS | 4 | SEMESTER | 2 |
| Dosen Pengampu |  | | | | |
| **BENTUK TUGAS** | | | | | |
| **Tugas 3:** Membangun Algoritma dan Mengimplementasikan Struktur Data Graf dalam Bahasa C++. | | | | | |
| **JUDUL TUGAS** | | | | | |
| Implementasi Struktur Data Graf. | | | | | |
| **SUB CAPAIAN PEMBELAJARAN** | | | | | |
| Mahasiswa mampu membangun algoritma untuk primitif struktur data Graf dan mengimplementasikannya dalam Bahasa Pemrograman C++ (CLO 3). | | | | | |
| **DESKRIPSI TUGAS** | | | | | |
| Implementasikan primitive-primitif dalam Graph tak berarah dan graph berarah yang meliputi:   1. Deklarasi struktur data. 2. Insert vertex dan insert tetangga. 3. Delete vertex dan delete tetangga.   Jenis struktur data yang digunakan yaitu:   1. Double Linked List 2. Circular List 3. Single List with Last 4. Double Linked List with Last 5. Circular List with Last | | | | | |
| **METODE PENGERJAAN TUGAS** | | | | | |
| 1. Yang harus dikerjakan dan batasan-batasan: 2. Membuat kelompok yang terdiri dari maksimal 2 mahasiswa. 3. Jenis struktur data yang digunakan oleh setiap kelompok ditentukan oleh dosen yang bersangkutan. 4. Membangun dan mengimplementasikan primitive-primitif dalam deskripsi tugas dalam Bahasa C++. 5. Tahapan Pengerjaan: 6. Merancang algoritma dari primitive-primitif dan main program yang akan diimplementasikan. 7. Mengimplementasikan primitive-primitif dan main program yang sudah dirancang dalam Bahasa C++. 8. Memastikan program yang dibangun dapat berjalan (running) sesuai deskripsi tugas. 9. Menulis laporan tugas. | | | | | |
| **BENTUK DAN FORMAT LUARAN** | | | | | |
| 1. Source code program C++. 2. Laporan tugas yang terdiri dari permasalahan yang dihadapi, gambaran struktur data beserta penjelasannya, rancangan algoritma, dan print screen output dari setiap primitive yang dibangun. 3. Pada laporan tugas harus dituliskan Kelas, Nomor kelompok, Nama lengkap, NIM, peran dari setiap anggota kelompok, dan tanda tangan mahasiswa yang ikut mengerjakan. 4. Source Code dan Laporan tugas dikumpulkan pada waktu dan tempat yang telah ditentukan oleh dosen. | | | | | |
| **INDIKATOR, KRITERIA DAN BOBOT PENILAIAN** | | | | | |
| 1. Source code program: 70%. 2. Laporan tugas: 30%. 3. Bagi yang melakukan kecurangan/plagiarism nilainya akan diturunkan sesuai dengan tingkat plagiarismenya. | | | | | |
| **JADWAL PELAKSANAAN** | | | | | |
| Tugas diberikan pada Minggu-2 perkuliahan setelah UTS.  Batas akhir pengumpulan pada Minggu-6 perkuliahan setelah UTS. | | | | | |
| **LAIN-LAIN** | | | | | |
| Bobot nilai **Tugas 3** adalah **9%** dari total bobot mata kuliah Algoritma dan Struktur Data. | | | | | |
| **DAFTAR RUJUKAN** | | | | | |
| Diktat Kuliah IF2181 Struktur Data, Inggriani Liem, ITB, 2003. | | | | | |