



UNIVERSITAS PGRI YOGYAKARTA

Jl. PGRI I Sonosewu No. 117 Yogyakarta - 55182 Telp. (0274) 376808, 373198, 373038 Fax. (0274) 376808

E-mail : info@upy.ac.id

<http://www.upy.ac.id>

P E T I K A N
KEPUTUSAN REKTOR UNIVERSITAS PGRI YOGYAKARTA
Nomor : 039.1 /SK/REKTOR-UPY/III/2024

Tentang

**PENGANGKATAN DOSEN PENGAMPU MATA KULIAH SEMESTER GENAP
TAHUN AKADEMIK 2023/2024 DI LINGKUNGAN UNIVERSITAS PGRI YOGYAKARTA**

DENGAN RAHMAT TUHAN YANG MAHA ESA
REKTOR UNIVERSITAS PGRI YOGYAKARTA

Menimbang : dst.
Mengingat : dst.
Memperhatikan: dst.

M E M U T U S K A N

Menetapkan : PENGANGKATAN DOSEN PENGAMPU MATA KULIAH SEMESTER GENAP TAHUN AKADEMIK 2023/2024 DI LINGKUNGAN UNIVERSITAS PGRI YOGYAKARTA

Pertama : Mengangkat Saudara yang namanya tersebut pada lajur 2 Lampiran keputusan ini sebagai Dosen Pengampu Mata Kuliah pada Semester Genap Tahun Akademik 2023/2024.

Kedua : Menugaskan kepada para Dosen Pengampu Mata Kuliah dimaksud untuk melaksanakan pembelajaran matakuliah sebagaimana tercantum pada lajur 3 lampiran keputusan ini dengan sebaik-baiknya dan kepada yang bersangkutan diberikan honorarium sesuai dengan ketentuan yang berlaku di UNIVERSITAS PGRI YOGYAKARTA

Ketiga : Keputusan ini berlaku sejak tanggal ditetapkan, dengan ketentuan bahwa segala sesuatunya akan ditinjau kembali apabila terdapat kekeliruan dalam penetapan ini.

PETIKAN Keputusan ini disampaikan kepada yang bersangkutan untuk dipergunakan sebagaimana mestinya.

Ditetapkan di : Yogyakarta

Pada tanggal : 01 Maret 2024

Rektor,

ttd

Dr. Ir. Paiman, M.P

NIS. 19650916 199503 1 003 07

Untuk Petikan yang sah
Wakil Rektor Bidang Akademik dan Kelembagaan

Ahmad Riyadi, S.Si, M.Kom
NIS. 19690214 199812 1 006

Tembusan disampaikan kepada :

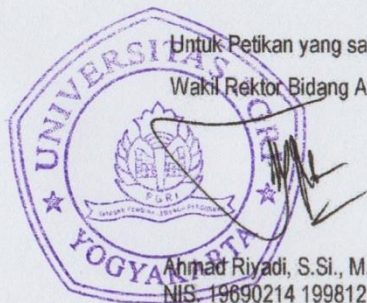
1. Para Wakil Rektor
2. Para Dekan dan Direktur
3. Para Ketua Program Sarjana

Lampiran Keputusan Rektor Universitas PGRI Yogyakarta

Nomor : 039.1/SK/REKTOR-UPY/III/2024

Tanggal : 01 Maret 2024

NO.	NAMA PENGAJAR & NIDN	MATA KULIAH	KODE MK	SKS	SEMESTER/ KELAS	PROGRAM
1. s.d 250						
251	Gema Kharismajati., S.Kom., M.Kom 0514019601	Artificial Intelligence Praktikum Artificial Intelligence Struktur Data	TKM132118 TKM132119 TKM11218	2 1 4	IV/ A IV/ A III/ A3	Program Sarjana Teknologi Rekayasa Elektro-medis Program Sarjana Teknologi Rekayasa Elektro-medis Program Sarjana Informatika
252						
Dst.						



Untuk Petikan yang sah:

Wakil Rektor Bidang Akademik dan Kelembagaan

Ahmad Riyadi, S.Si., M.Kom
NIS. 19690214 199812 1 006

Rektor

ttd

Dr. Ir. Paiman, M.P
NIS. 19650916 199503 1 003

RANCANGAN PEMBELAJARAN SEMESTER
MATA KULIAH STRUKTUR DATA



Oleh:

GEMA KHARISMAJATI, S.Kom.,M.Kom.
NIS. 19960114 202401 1 006

PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PGRI YOGYAKARTA
2024

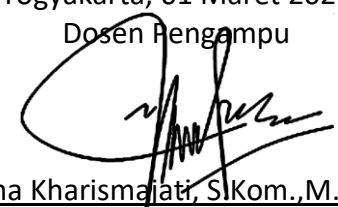
HALAMAN PENGESAHAN

1. RPS Mata Kuliah : Struktur Data
2. Pelaksana
 - a. Nama Lengkap : Gema Kharismajati, S.Kom.,M.Kom.
 - b. Jenis Kelamin : Laki-laki
 - c. Pangkat/Golongan : Penata Muda Tk. I / III b
 - d. NIP/NIS : 19960114 202401 1 006
 - e. Program Studi/Fakultas : Sistem Informasi / Sains dan Teknologi
 - f. Telpon/E-mail : gemakharismajati@upy.ac.id
3. Pembiayaan
 - a. Sumber Dana : Lembaga Pengembangan Pendidikan Universitas PGRI Yogyakarta
Rp 500.000,-
 - b. Jumlah Biaya :

Mengetahui,
Kaprodik Teknik Informatika


Puji Handayani Putri, S.T.,M.K
NIS. 19841227 201604 1 011

Yogyakarta, 01 Maret 2024
Dosen Pengampu


Gema Kharismajati, S.Kom.,M.Kom.
NIS. 19960114 202401 1 006



Menyetujui,
Kepala Lembaga Pengembangan Pendidikan

Selly Rahmawati, M.Pd
NIS. 19870723 201302 2 002



UNIVERSITAS PGRI YOGYAKARTA
FAKULTAS SAINS DAN TEKNOLOGI
PROGRAM STUDI INFORMATIKA

RENCANA PEMBELAJARAN SEMESTER

MATA KULIAH	KODE	Rumpun MK	BOBOT (sks)		SEMESTER	Tgl Penyusunan
Struktur Data	TKM11218	Matakuliah Wajib	T=2	P=2	2	01 / 0 3 / 2024
OTORISASI / PENGESAHAN	Dosen Pengembang RPS		Koordinator RMK		Ka PRODI	
	 Gema Kharismajati, S.Kom.,M.Kom				 Puji Handayani Putri, S.T.,M.K	
Capaian Pembelajaran (CP)	CPL-PRODI					
S1	Mampu menunjukkan sikap religius, disiplin, bertanggung jawab, menghormati nilai-nilai kemanusiaan, saling menghormati, dan patuh pada hukum dalam kehidupan bermasyarakat, berbangsa, dan bernegara berdasarkan nilai-nilai Pancasila					
S2	Mampu menunjukkan sikap bertanggungjawab atas pekerjaan di bidang keahliannya secara mandiri maupun tim					
PP1	Menguasai konsep teoritis di bidang Informatika, khususnya di bidang komputasi, analisis data, sistem cerdas, jaringan komputer, teknologi web, teknologi mobile, teknologi informasi, dan basis data					
PP2	Memiliki pengetahuan yang memadai terkait dengan cara kerja sistem komputer dan mampu merancang dan mengembangkan berbagai algoritma/metode untuk memecahkan masalah					
KU1	Mampu bekerja dan berkomunikasi secara efektif baik secara individu maupun dalam kelompok					
KU4	Mampu merancang dan mengembangkan aplikasi dengan menerapkan prinsip-prinsip sistem cerdas, komputasi dan optimasi untuk menghasilkan aplikasi cerdas dalam berbagai bidang					
KK3	Mampu merancang dan mengembangkan perangkat lunak dengan performance yang baik secara teknis dan manajerial menggunakan prinsip-prinsip proses rekayasa perangkat lunak					

	CPMK
	CPMK1 Capaian Pembelajaran Mata Kuliah (CPMK) dari mata kuliah Struktur Data bertujuan untuk memberikan pemahaman yang kokoh mengenai konsep dasar serta implementasi praktis dari berbagai struktur data yang umum digunakan dalam pemrograman komputer. Mahasiswa diharapkan dapat memahami esensi dari struktur data seperti list, tuple, dictionary, dan set, serta mampu mengimplementasikannya menggunakan bahasa pemrograman Python. Selain itu, mereka juga diarahkan untuk memahami konsep fundamental dari struktur data non-linear, termasuk array, linked list, stack, queue, graph, dan tree, beserta penerapannya dalam pemecahan masalah nyata. Dengan demikian, mahasiswa diharapkan dapat memilih struktur data yang sesuai dan menggunakannya secara efektif dalam menyelesaikan berbagai permasalahan komputasi, serta memiliki keterampilan dalam mengimplementasikan solusi menggunakan struktur data yang tepat dan efisien.
Diskripsi Singkat MK	Mata kuliah ini memberikan dasar-dasar pengenalan dan pemahaman mengenai struktur data yang sering digunakan dalam pemrograman komputer. Mahasiswa akan mempelajari berbagai jenis struktur data seperti list, tuple, dictionary, dan set, serta cara mengimplementasikannya menggunakan bahasa pemrograman Python. Selain itu, mahasiswa juga akan diperkenalkan dengan konsep-konsep dasar dari array, linked list, stack, queue, graph, dan tree, dan cara pengaplikasiannya dalam pemecahan masalah nyata.
Tujuan Pembelajaran	Setelah menyelesaikan mata kuliah ini, mahasiswa diharapkan mampu: <ol style="list-style-type: none"> 1. Memahami konsep dasar berbagai struktur data. 2. Mengimplementasikan berbagai struktur data menggunakan Python. 3. Menggunakan struktur data secara efektif untuk memecahkan masalah komputasi.
Bahan Kajian (Materi pembelajaran)	<ol style="list-style-type: none"> 1. Konsep dasar struktur data 2. Struktur data linear: list, tuple, dictionary, set 3. Konsep Dasar : Array, Linked List, Stack, Queue, Graph & Tree 4. Implementasi struktur data dalam Python 5. Pemanfaatan struktur data dalam pemrograman
Pustaka	Utama: <ol style="list-style-type: none"> 1. "Data Structures and Algorithms in Python" by Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser 2. "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein.

	Pendukung: "Python Data Structures and Algorithms" by Benjamin Baka
Dosen Pengampu	Gema Kharismajati, S.Kom.,M.Kom.
Matakuliah syarat	

Minggu	Kemampuan Akhir yang diharapkan	Bahan Kajian/Materi	Strategi Pembelajaran	Waktu	Kegiatan yang dilakukan	Kriteria Penilaian (Indikator)	Bobot
1	Memahami pengantar kuliah dan rencana pembelajaran	Perkenalan, Kontrak Kuliah, Pengenalan Rencana Pembelajaran	Ceramah, diskusi	3 jam	Pengenalan dosen, perkenalan antar mahasiswa, pembahasan silabus	Kehadiran dan partisipasi aktif	5%
2	Memahami konsep dasar struktur data dan pentingnya	Konsep Dasar Struktur Data, Pentingnya Struktur Data	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, diskusi kelompok	Pemahaman konsep dasar	5%
3	Mengenal dasar-dasar Python dan Pycharm	Struktur Dasar Python: Variable, Tipe Data, Operator, List, Tuple, Dictionary, Set	Ceramah, demo, tanya jawab	3 jam	Instalasi dan pengenalan Pycharm, latihan dasar Python	Kehadiran, partisipasi, dan latihan	5%
4	Memahami dan mengimplementasikan struktur data dasar	Struktur Dasar Python: Variable, Tipe Data, Operator, List, Tuple, Dictionary, Set	Ceramah, demo, tanya jawab	3 jam	Latihan dasar Python lebih lanjut	Pemahaman dan implementasi dasar	5%
5	Memahami konsep dasar struktur data List	Konsep dasar struktur data List, Operasi dasar (indexing, slicing, append, extend, insert, remove, sort)	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, latihan konsep dasar List	Pemahaman dan implementasi List	5%
6	Mengimplementasikan struktur data List dalam	Implementasi menggunakan struktur	Ceramah, demo, tanya jawab	3 jam	Implementasi struktur data List dalam	Keberhasilan implementasi List	10%

	Python	data List, tugas praktikum struktur data List			praktikum		
7	Memahami konsep dasar struktur data Tuple	Konsep dasar struktur data Tuple, Operasi dasar tuple (indexing, slicing, unpacking)	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, latihan konsep dasar Tuple	Pemahaman dan implementasi Tuple	5%
8	Mengimplementasikan struktur data Tuple dalam Python	Mengimplementasikan operasi Tuple menggunakan Python, tugas praktikum struktur data Tuple	Ceramah, demo, tanya jawab	3 jam	Implementasi struktur data Tuple dalam praktikum	Keberhasilan implementasi Tuple	10%
9	Memahami konsep dasar struktur data Dictionary	Konsep dasar struktur data Dictionary, Operasi Dasar (key-value pairs, indexing, slicing, update)	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, latihan konsep dasar Dictionary	Pemahaman dan implementasi Dictionary	5%
10	Mengimplementasikan struktur data Dictionary dalam Python	Mengimplementasikan operasi Dictionary menggunakan Python, tugas praktikum struktur data Dictionary	Ceramah, demo, tanya jawab	3 jam	Implementasi struktur data Dictionary dalam praktikum	Keberhasilan implementasi Dictionary	10%
11	Memahami konsep dasar struktur data Set	Konsep dasar struktur data Set, Operasi dasar set (union, intersection, difference, symmetric difference)	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, latihan konsep dasar Set	Pemahaman dan implementasi Set	5%
13	Mengimplementasikan struktur data Set dalam Python	Mengimplementasikan operasi Set menggunakan Python, tugas praktikum struktur data Set	Ceramah, demo, tanya jawab	3 jam	Implementasi struktur data Set dalam praktikum	Keberhasilan implementasi Set	10%
14	Memahami konsep dasar Array & Linked List	Konsep dasar array dan linked list, implementasi array dan linked list pada python	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, latihan konsep dasar array dan linked list	Pemahaman dan implementasi Array & Linked List	5%

15	Memahami konsep dasar Stack & Queue	Konsep dasar stack dan queue, implementasi stack & queue pada python	Ceramah, tanya jawab, diskusi	3 jam	Penjelasan materi, latihan konsep dasar stack dan queue	Pemahaman dan implementasi Stack & Queue	5%
----	-------------------------------------	--	-------------------------------	-------	---	--	----

FORMAT RANCANGAN TUGAS 1

Nama Mata Kuliah : Struktur Data
Program Studi : Informatika
Fakultas : Sains & Teknologi

Sks : 4 sks (2 sks Teori dan 2 sks Praktik)
Kode MK : TKM11218
Pertemuan ke : 2

Tugas 1: Implementasi Struktur Data List dalam Python

Deskripsi Tugas: Mahasiswa diminta untuk mengimplementasikan berbagai operasi dasar pada struktur data List menggunakan bahasa pemrograman Python. Tugas ini bertujuan untuk menguji pemahaman mahasiswa mengenai konsep dasar dan operasi pada List serta kemampuan mereka dalam mengaplikasikan konsep tersebut dalam bentuk kode yang fungsional.

Instruksi Tugas:

1. Buatlah sebuah program Python yang dapat melakukan operasi dasar pada List, termasuk:
 - Penambahan elemen (append, insert)
 - Penghapusan elemen (remove, pop)
 - Pengurutan elemen (sort)
 - Pengaksesan elemen (indexing, slicing)
2. Program harus memiliki antarmuka pengguna yang memungkinkan pengguna memasukkan pilihan operasi yang akan dilakukan.
3. Setiap operasi harus diimplementasikan dalam fungsi terpisah untuk menjaga modularitas kode.
4. Berikan komentar pada setiap bagian penting dari kode untuk menjelaskan fungsinya.
5. Sertakan dokumentasi singkat tentang cara menjalankan program dan penjelasan fungsi-fungsi yang ada di dalamnya.
6. Tugas harus diserahkan dalam bentuk file Python (.py) dan dokumen pendukung (jika ada) dalam format PDF.

Kriteria Penilaian:

Kriteria	Bobot	Deskripsi Penilaian
Implementasi Operasi Dasar	30%	Kemampuan untuk mengimplementasikan operasi dasar seperti append, remove, sort, indexing, dan slicing dengan benar dan efisien.
Struktur dan Kerapihan Kode	20%	Kode yang ditulis harus terstruktur dengan baik, menggunakan fungsi dengan jelas, dan mengikuti prinsip-prinsip pemrograman yang baik.
Dokumentasi dan Komentar	15%	Memberikan komentar yang jelas pada setiap bagian penting dari kode serta dokumentasi singkat yang menjelaskan cara kerja program.
Fungsi Program dan Antarmuka	20%	Program harus berjalan dengan baik tanpa kesalahan, dan antarmuka pengguna harus mudah digunakan serta interaktif.
Inovasi dan Kreativitas	15%	Kreativitas dalam memperluas fitur program, seperti menambahkan fitur tambahan yang berguna atau membuat antarmuka yang lebih menarik.

FORMAT RANCANGAN TUGAS 2

Nama Mata Kuliah : Struktur Data
Program Studi : Informatika
Fakultas : Sains & Teknologi

Sks : 4 sks (2 sks Teori dan 2 sks Praktik)
Kode MK : TKM11218
Pertemuan ke : 2

Tugas 2: Analisis dan Implementasi Struktur Data Tuple dalam Python

Deskripsi Tugas: Mahasiswa diminta untuk melakukan analisis dan mengimplementasikan berbagai operasi dasar pada struktur data Tuple menggunakan bahasa pemrograman Python. Tugas ini bertujuan untuk menguji pemahaman mahasiswa tentang konsep dasar Tuple dan kemampuan mereka dalam mengaplikasikan konsep tersebut secara efektif.

Instruksi Tugas:

1. Buatlah sebuah program Python yang dapat melakukan operasi dasar pada Tuple, termasuk:
 - Pengaksesan elemen (indexing, slicing)
 - Unpacking elemen
 - Menggabungkan beberapa Tuple
 - Menghitung jumlah elemen tertentu dalam Tuple (count)
 - Menemukan indeks elemen tertentu (index)
2. Program harus memiliki antarmuka pengguna yang memungkinkan pengguna memasukkan pilihan operasi yang akan dilakukan.
3. Setiap operasi harus diimplementasikan dalam fungsi terpisah untuk menjaga modularitas kode.
4. Berikan komentar pada setiap bagian penting dari kode untuk menjelaskan fungsinya.
5. Sertakan dokumentasi singkat tentang cara menjalankan program dan penjelasan fungsi-fungsi yang ada di dalamnya.

Tugas harus diserahkan dalam bentuk file Python (.py) dan dokumen pendukung (jika ada) dalam format PDF.

Kriteria Penilaian:

Kriteria	Bobot	Deskripsi Penilaian
Implementasi Operasi Dasar	30%	Kemampuan untuk mengimplementasikan operasi dasar seperti indexing, slicing, unpacking, count, dan index dengan benar dan efisien.
Struktur dan Kerapihan Kode	20%	Kode yang ditulis harus terstruktur dengan baik, menggunakan fungsi dengan jelas, dan mengikuti prinsip-prinsip pemrograman yang baik.
Dokumentasi dan Komentar	15%	Memberikan komentar yang jelas pada setiap bagian penting dari kode serta dokumentasi singkat yang menjelaskan cara kerja program.
Fungsi Program dan Antarmuka	20%	Program harus berjalan dengan baik tanpa kesalahan, dan antarmuka pengguna harus mudah digunakan serta interaktif.
Inovasi dan Kreativitas	15%	Kreativitas dalam memperluas fitur program, seperti menambahkan fitur tambahan yang berguna atau membuat antarmuka yang lebih menarik.

FORMAT RANCANGAN TUGAS 3

Nama Mata Kuliah	: Struktur Data	Sks	: 4 sks (2 sks Teori dan 2 sks Praktik)
Program Studi	: Informatika	Kode MK	: TKM11218
Fakultas	: Sains & Teknologi	Pertemuan ke	: 2

Tugas 3: Penggunaan Struktur Data Dictionary dalam Aplikasi Sederhana

Deskripsi Tugas: Mahasiswa diminta untuk membuat aplikasi sederhana yang menggunakan struktur data Dictionary untuk menyimpan dan mengelola data. Tugas ini bertujuan untuk menguji kemampuan mahasiswa dalam menggunakan Dictionary untuk memecahkan masalah nyata.

Instruksi Tugas:

1. Buatlah sebuah program Python yang menggunakan Dictionary untuk menyimpan data mahasiswa, termasuk:
{“Nama”, “NIM”, “Jurusan”, “IPK”}
2. Program harus memungkinkan pengguna untuk:
 - Menambah data mahasiswa baru
 - Menghapus data mahasiswa berdasarkan NIM
 - Memperbarui data mahasiswa berdasarkan NIM
 - Menampilkan seluruh data mahasiswa
3. Setiap operasi harus diimplementasikan dalam fungsi terpisah untuk menjaga modularitas kode.
4. Berikan komentar pada setiap bagian penting dari kode untuk menjelaskan fungsinya.
5. Sertakan dokumentasi singkat tentang cara menjalankan program dan penjelasan fungsi-fungsi yang ada di dalamnya.
6. Tugas harus diserahkan dalam bentuk file Python (.py) dan dokumen pendukung (jika ada) dalam format PDF.

Kriteria Penilaian:

Kriteria	Bobot	Deskripsi Penilaian
Implementasi Operasi Dasar	30%	Kemampuan untuk mengimplementasikan operasi dasar seperti penambahan, penghapusan, pembaruan, dan penampilan data dengan benar dan efisien.
Struktur dan Kerapihan Kode	20%	Kode yang ditulis harus terstruktur dengan baik, menggunakan fungsi dengan jelas, dan mengikuti prinsip-prinsip pemrograman yang baik.
Dokumentasi dan Komentar	15%	Memberikan komentar yang jelas pada setiap bagian penting dari kode serta dokumentasi singkat yang menjelaskan cara kerja program.
Fungsi Program dan Antarmuka	20%	Program harus berjalan dengan baik tanpa kesalahan, dan antarmuka pengguna harus mudah digunakan serta interaktif.
Inovasi dan Kreativitas	15%	Kreativitas dalam memperluas fitur program, seperti menambahkan fitur tambahan yang berguna atau membuat antarmuka yang lebih menarik.

FORMAT RANCANGAN TUGAS 4

Nama Mata Kuliah : Struktur Data
Program Studi : Informatika
Fakultas : Sains & Teknologi

Sks : 4 sks (2 sks Teori dan 2 sks Praktik)
Kode MK : TKM11218
Pertemuan ke : 2

Tugas 4: Implementasi Struktur Data Set dalam Python

Deskripsi Tugas: Mahasiswa diminta untuk mengimplementasikan berbagai operasi dasar pada struktur data Set menggunakan bahasa pemrograman Python. Tugas ini bertujuan untuk menguji pemahaman mahasiswa mengenai konsep dasar dan operasi pada Set serta kemampuan mereka dalam mengaplikasikan konsep tersebut dalam bentuk kode yang fungsional.

Instruksi Tugas:

1. Buatlah sebuah program Python yang dapat melakukan operasi dasar pada Set, termasuk:
 - Penambahan elemen (add)
 - Penghapusan elemen (remove, discard)
 - Operasi himpunan (union, intersection, difference, symmetric difference)
2. Program harus memiliki antarmuka pengguna yang memungkinkan pengguna memasukkan pilihan operasi yang akan dilakukan.
3. Setiap operasi harus diimplementasikan dalam fungsi terpisah untuk menjaga modularitas kode.
4. Berikan komentar pada setiap bagian penting dari kode untuk menjelaskan fungsinya.
5. Sertakan dokumentasi singkat tentang cara menjalankan program dan penjelasan fungsi-fungsi yang ada di dalamnya.
6. Tugas harus diserahkan dalam bentuk file Python (.py) dan dokumen pendukung (jika ada) dalam format PDF.

Kriteria Penilaian:

Kriteria	Bobot	Deskripsi Penilaian
Implementasi Operasi Dasar	30%	Kemampuan untuk mengimplementasikan operasi dasar seperti add, remove, union, intersection, difference, dan symmetric difference dengan benar dan efisien.
Struktur dan Kerapihan Kode	20%	Kode yang ditulis harus terstruktur dengan baik, menggunakan fungsi dengan jelas, dan mengikuti prinsip-prinsip pemrograman yang baik.
Dokumentasi dan Komentar	15%	Memberikan komentar yang jelas pada setiap bagian penting dari kode serta dokumentasi singkat yang menjelaskan cara kerja program.
Fungsi Program dan Antarmuka	20%	Program harus berjalan dengan baik tanpa kesalahan, dan antarmuka pengguna harus mudah digunakan serta interaktif.
Inovasi dan Kreativitas	15%	Kreativitas dalam memperluas fitur program, seperti menambahkan fitur tambahan yang berguna atau membuat antarmuka yang lebih menarik.



Kontrak Kuliah
Fakultas Sains & Teknologi
Universitas PGRI Yogyakarta

Nama Dosen : Gema Kharismajati, S.Kom.,M.Kom.
Mata Kuliah : Struktur Data
Program Studi : Informatika
Kelas/Angkatan : 23.A3/2023
Semester : 2
Tahun Akademik : 2023/2024

Capaian Pembelajaran/Learning Outcome:

Mata kuliah ini bertujuan untuk memperkenalkan konsep dasar struktur data dan pemrograman menggunakan bahasa pemrograman Python. Setelah menyelesaikan mata kuliah ini, mahasiswa diharapkan mampu memahami konsep-konsep dasar struktur data seperti list, tuple, dictionary, dan set. Mereka akan dapat mengimplementasikan struktur data tersebut dalam pemrograman Python untuk menyelesaikan permasalahan komputasi sederhana. Selain itu, mahasiswa diharapkan mampu menerapkan pengetahuan yang diperoleh dalam mengevaluasi dan memilih struktur data yang tepat untuk pengembangan solusi perangkat lunak.

Soft Skills

Selain meningkatkan pemahaman teknis dalam pemrograman dan struktur data, mata kuliah ini juga bertujuan untuk mengembangkan beberapa soft skills penting. Mahasiswa akan belajar untuk bekerja secara kolaboratif dalam tim dalam menyelesaikan tugas-tugas praktikum yang melibatkan implementasi dan analisis struktur data. Mereka akan meningkatkan keterampilan komunikasi mereka melalui presentasi materi dan diskusi kelompok. Selain itu, mahasiswa akan melatih kemampuan pemecahan masalah dan kritis dalam menganalisis dan merancang solusi menggunakan struktur data.

Bahan Kajian


pengantar tentang struktur data, yang meliputi definisi, jenis-jenis, dan keuntungan penggunaannya. Mahasiswa akan mempelajari konsep dasar tentang bagaimana data diatur dan disimpan dalam memori komputer, serta pentingnya memahami berbagai struktur data untuk efisiensi dan efektivitas dalam pengembangan solusi perangkat lunak. Dengan pemahaman yang kuat tentang struktur data, mahasiswa akan mampu mengidentifikasi dan menerapkan struktur data yang sesuai dengan kebutuhan pemrograman mereka.

Ketentuan

1. Kehadiran perkuliahan tatap muka minimal 75%.
2. Toleransi keterlambatan perkuliahan tatap muka 15 menit.
3. Dalam perkuliahan dan konsultasi dilakukan dengan sopan dan menghargai.
4. Menggunakan Pakaian Rapi, Sopan, Berkerah
5. Mahasiswa wajib mengikuti UAS.


Penilaian

No	Uraian	Bobot (%)
1	Tata Krama	15%
2	Kehadiran	15%
3	Tugas	30%
4	Kuis	20%
5	UAS	20%

Kaprodi
Informatika

Puji Handayani Putri, S.T.,M.K.
NIS. 19841227 201604 1 011

Dosen Pengampu

Gema Kharismajati, S.Kom.,M.Kom.
NIS. 19960114 202401 1 006

Yogyakarta, 18 Maret 2024
Ketua Kelas

Muhamad Agil Nuari
NPM. 23111100057

**PRESENSI DOSEN MENGAJAR**

TA.2023/2024 Sem. GENAP

Program Studi : INFORMATIKA
 Mata Kuliah : STRUKTUR DATA [TKM11218]
 Bobot : 4 SKS
 Dosen : GEMA KHARISMAJATI, S.Kom.,M.Kom.

Kelas : 23.A3
 Hari : RABU
 Pukul : 12.00 – 15-20
 Ruang : LM

Pert	Tanggal	Pokok Bahasan	Sub-Pokok Bahasan	Jml Mhs	Paraf
I	06/03/2024	Pengantar Mata Kuliah Struktur Data	- Perkenalan Dosen & Mahasiswa - Kontrak Kuliah - Penjelasan Rencana Pembelajaran	18	
II	13/03/2024	Pengantar Mata Kuliah Struktur Data	- Konsep Dasar Struktur Data - Pentingnya Struktur Data	26	
III	20/03/2024	Pengenalan Python dan Pycharm	- Pengantar Python dan instalasi PyCharm - Navigasi PyCharm/pengenalan antarmuka	25	
IV	27/03/2024	Struktur Data Dasar	- Struktur Dasar Python: Variable, Tipe Data, Operator - Pemahaman tentang List, Tuple, Dictionary, dan Set	25	
V	03/04/2024	Struktur data List	- Konsep dasar struktur data List - Operasi dasar (indexing, slicing, append, extend, insert, remove, sort).	26	
VI	17/04/2024	Implementasi struktur data List	- Implementasi menggunakan struktur data List - tugas praktikum struktur data list	26	
VII	24/04/2024	Struktur data tuple	- Konsep dasar struktur data Tuple - Operasi dasar tuple (indexing, slicing, unpacking)	25	
VIII	01/05/2024	Implementasi struktur data tuple (daring)	- Mengimplementasikan operasi tuple menggunakan Python. - tugas praktikum struktur data tuple	24	
IX	08/05/2024	Struktur data Dictionary	- Konsep dasar struktur data Dictionary - Operasi Dasar (key-value pairs, indexing, slicing, update)	25	
X	15/05/2024	Implementasi Struktur data Dictionary	- Mengimplementasikan operasi Dictionary menggunakan Python - tugas praktikum struktur data dictionary.	26	
XI	22/05/2024	Struktur data set	- Konsep dasar struktur data set - Operasi dasar set (union, intersection, difference, symmetric difference).	23	
XII	29/05/2024	Implementasi Struktur data Set	- Mengimplementasikan operasi set menggunakan Python. - tugas praktikum struktur data set.	26	
XIII	05/06/2024	Array & linked list	- Konsep dasar array dan linked list - implementasi array dan linked list pada python	26	
XIV	12/06/2024	Stack & queue	- Konsep dasar stack dan queue - implementasi stack & queue pada python	26	
XV	19/06/2024	Evaluasi Pembelajaran	Evaluasi materi untuk persiapan uas	26	



DAFTAR HADIR KULIAH

Program Studi : INFORMATIKA
 Tahun Akademik : 2023/2024
 Semester : GENAP
 Dosen : GEMA KHARISMAJATI,S.Kom.,M.Kom.

Kode Matakuliah : TKM11218
 Matakuliah : STRUKTUR DATA
 Bobot : 4 SKS
 Kelas : 23.A3

Semester : 2
 Hari : RABU
 Pukul : 12.00-15.30
 Ruang : LM

No	NP Mahasiswa	Nama mahasiswa	B/U/P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Jumlah Hadir	%
				06/03/24	13/03/24	20/03/24	27/03/24	03/04/24	17/04/24	24/04/24	01/05/24	08/05/24	15/05/24	22/05/24	29/05/24	05/06/24	12/06/24	19/06/24		
1	23111100017	RAVITA INDRIANI	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
2	23111100053	MUHAMMAD BANAR RIZKI	B	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	14	93%
3	23111100055	ADRY MOHAMMAD AZMAN	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	S	✓	✓	✓	✓	14	93%
4	23111100056	RIZKY NUR HIDAYAT	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	I	✓	✓	✓	✓	14	93%
5	23111100057	MUHAMAD AGIL NUARI	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
6	23111100059	MUHAMMAD ALL AMIN	B	✓	✓	✓	✓	✓	✓	✓	i	i	✓	✓	✓	✓	✓	✓	13	87%
7	23111100061	AGUNG UTAMA	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
8	23111100062	SRI CAHYA PUTRA	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
9	23111100064	MUHAMMAD NAUFAL	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
10	23111100065	JAMALUDIN MUHAMAD IQBAL	B	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	14	93%
11	23111100068	NOUVAL ACHMAD RAFIQ	B	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	14	93%
12	23111100070	AHMAD SYAEFUL ARIF	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
13	23111100072	GILANG ARDANU PUTRA	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
14	23111100073	BRAMANTYA NALENDRA	B	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	14	93%
15	23111100076	SILVIA ARDILLA HANDAYANI	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
16	23111100078	DELA RAHMADANI HASIBUAN	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
17	23111100079	REVITA AGUSTIN	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
18	23111100081	FITRA IKA PUTRI	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
19	23111100088	MUHAMMAD LUHFI RISYAMSU	B	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	14	93%
20	23111100089	MIFTAHUL ARIFIN	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	S	✓	✓	✓	✓	14	93%
21	23111100090	RAFI ARKANANTA NAUFAL	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
22	23111100093	FREDERIKUS EDUARDO UMU RERE	B	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	14	93%
23	23111100091	AMANDA DIAH AYU	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
24	23111100082	ROFIK SYARIFUDIN	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	15	100%
25	23111100087	RIZAL NUR SALAM	B	-	✓	-	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	12	80%
26	23111100085	SAVIOLLA WARIH SANTOSO	B	-	✓	✓	-	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	12	80%
TOTAL KEHADIRAN				18	26	25	25	26	26	25	24	25	26	23	26	26	26	26		



PRESENSI UJIAN AKHIR SEMESTER GENAP

TAHUN AKADEMIK 2023/2024

Program Studi : INFORMATIKA
Mata Kuliah : STRUKTUR DATA [TKM11218]
Kode MK : TKM11218
Dosen : GEMA KHARISMAJATI, S.Kom.,M.Kom.

Kelas : 23.A3
Ruang :
Hari/Tanggal : Senin, 01-07-2024
Waktu : 10:00 – 11:00

No	NPM	Nama Mahasiswa	B/U/P	Kehadiran	Nilai
1	23111100017	RAVITA INDRIANI	B	✓	A
2	23111100053	MUHAMMAD BANAR RIZKI	B	✓	A
3	23111100055	ADRY MOHAMMAD AZMAN	B	✓	A
4	23111100056	RIZKY NUR HIDAYAT	B	✓	A
5	23111100057	MUHAMAD AGIL NUARI	B	✓	A
6	23111100059	MUHAMMAD ALL AMIN	B	✓	A-
7	23111100061	AGUNG UTAMA	B	✓	A-
8	23111100062	SRI CAHYA PUTRA	B	✓	A
9	23111100064	MUHAMMAD NAUFAL	B	✓	A
10	23111100065	JAMALUDIN MUHAMAD IQBAL	B	✓	A
11	23111100068	NOUVAL ACHMAD RAFIQ	B	✓	A
12	23111100070	AHMAD SYAEFUL ARIF	B	✓	A-
13	23111100072	GILANG ARDANU PUTRA	B	✓	A
14	23111100073	BRAMANTYA NALENDRA	B	✓	A
15	23111100076	SILVIA ARDILLA HANDAYANI	B	✓	A
16	23111100078	DELA RAHMADANI HASIBUAN	B	✓	A
17	23111100079	REVITA AGUSTIN	B	✓	A
18	23111100081	FITRA IKA PUTRI	B	✓	A
19	23111100088	MUHAMMAD LUHFI RISYAMSU	B	✓	A
20	23111100089	MIFTAHUL ARIFIN	B	✓	A-
21	23111100090	RAFI ARKANANTA NAUFAL	B	✓	A
22	23111100093	FREDERIKUS EDUARDO UMU RERE	B	✓	A-
23	23111100091	AMANDA DIAH AYU	B	✓	A
24	23111100082	ROFIK SYARIFUDIN	B	✓	A
25	23111100087	RIZAL NUR SALAM	B	✓	B+
26	23111100085	SAVIOLLA WARIH SANTOSO	B	✓	B+

BUKU AJAR

STRUKTUR DATA



GEMA KHARISMAJATI, S.Kom.,M.Kom

Pendahuluan

Modul Struktur Data: ini dirancang sebagai pengantar yang komprehensif untuk memahami dan mengimplementasikan struktur data menggunakan Python, sebuah bahasa pemrograman yang kuat dan banyak digunakan di industri. Relevansi modul ini sangat tinggi, terutama dalam bidang pengembangan perangkat lunak dan analisis data, di mana pengelolaan data yang efisien sangat dibutuhkan. Dengan mengikuti modul ini, mahasiswa diharapkan mampu memahami konsep dasar struktur data, memanipulasi data dengan Python, serta menerapkan struktur data untuk menyelesaikan berbagai masalah dalam dunia nyata.

Deskripsi Materi: Mata kuliah ini membahas berbagai konsep dasar dalam struktur data, termasuk tipe-tipe data dasar seperti variabel, operator, dan tipe data kompleks seperti List, Tuple, Dictionary, dan Set. Setiap konsep akan dijelaskan secara mendetail, disertai dengan contoh-contoh implementasi dalam Python.

Konsep Dasar dan Pentingnya Struktur Data

A. Pendahuluan

Modul ini dirancang untuk memberikan pemahaman fundamental mengenai apa itu struktur data dan mengapa struktur data sangat penting dalam pemrograman dan pengembangan perangkat lunak. Relevansi modul ini terletak pada kemampuan mahasiswa untuk memilih dan menggunakan struktur data yang tepat untuk berbagai aplikasi, yang merupakan keterampilan esensial dalam pengembangan software. Dengan mengikuti modul ini, mahasiswa diharapkan mampu memahami konsep dasar struktur data dan mengaplikasikan konsep tersebut dalam berbagai konteks pemrograman.

B. Deskripsi Materi

1. Konsep Dasar Struktur Data

Struktur data adalah cara pengorganisasian, pengelolaan, dan penyimpanan data sehingga dapat diakses dan dimodifikasi secara efisien. Dalam pemrograman, struktur data digunakan untuk mengatur data sehingga operasi seperti pencarian, penyisipan, penghapusan, dan pengurutan dapat dilakukan dengan lebih efisien. Beberapa contoh struktur data yang umum digunakan adalah array, linked list, stack, queue, tree, dan graph.

2. Jenis – Jenis Struktur Data

- Array: Struktur data linier dengan elemen-elemen yang disusun dalam urutan tertentu dan memiliki ukuran tetap.
- Linked List: Struktur data linier di mana setiap elemen (disebut node) berisi data dan referensi ke elemen berikutnya.
- Stack: Struktur data linier yang mengikuti prinsip LIFO (Last In, First Out).

- Queue: Struktur data linier yang mengikuti prinsip FIFO (First In, First Out).
- Tree: Struktur data hierarkis yang terdiri dari simpul-simpul (nodes), dengan satu simpul sebagai akar (root).
- Graph: Struktur data yang terdiri dari simpul-simpul (vertices) yang dihubungkan oleh tepi-tepi (edges).

3. Pentingnya Struktur Data

- Efisiensi: Struktur data yang tepat dapat mengurangi waktu dan ruang yang diperlukan untuk operasi tertentu, seperti pencarian atau pengurutan.
- Pemecahan Masalah yang Lebih Baik: Dengan memahami struktur data, programmer dapat memilih alat yang tepat untuk memecahkan masalah tertentu, membuat solusi lebih efektif dan efisien.
- Pengembangan Perangkat Lunak yang Skalabel: Struktur data yang tepat memungkinkan pengembangan perangkat lunak yang lebih modular, mudah diuji, dan lebih mudah dioptimalkan.

Pentingnya struktur data terletak pada kemampuannya untuk mengoptimalkan penggunaan sumber daya komputasi seperti waktu dan memori. Dengan memilih struktur data yang tepat untuk sebuah masalah, programmer dapat mengurangi kompleksitas algoritma yang digunakan, sehingga program dapat berjalan lebih cepat dan lebih efisien.

Misalnya, jika Anda perlu menyimpan data yang urutannya penting, maka array atau linked list mungkin merupakan pilihan yang baik. Namun, jika Anda perlu sering menambah atau menghapus data, linked list mungkin lebih efisien dibandingkan array. Selain itu, struktur data seperti tree dan graph sering digunakan dalam pemrosesan data yang kompleks, seperti pencarian jalur terpendek atau pengurutan hierarki.

Pemahaman yang baik tentang struktur data juga penting dalam pengembangan perangkat lunak yang skalabel dan maintainable. Dengan

menggunakan struktur data yang tepat, kode menjadi lebih modular, lebih mudah untuk diuji, dan lebih mudah untuk dioptimalkan di masa depan.

4. Contoh Penggunaan Struktur Data

- **Array:** Cocok digunakan ketika jumlah elemen sudah diketahui dan tidak berubah-ubah, seperti menyimpan nilai-nilai ujian.
- **Linked List:** Berguna ketika sering terjadi penambahan atau penghapusan elemen di tengah-tengah data, seperti antrian pada sistem.
- **Stack dan Queue:** Sering digunakan dalam pengaturan tugas dan pemrosesan data, seperti undo/redo dalam aplikasi atau antrian cetak.
- **Tree dan Graph:** Digunakan untuk masalah yang melibatkan hirarki atau jaringan, seperti sistem file komputer atau pemetaan rute.

C. Latihan

1. **Diskusi Kelompok:** Mahasiswa diminta untuk mendiskusikan contoh-contoh nyata dari kehidupan sehari-hari di mana berbagai jenis struktur data dapat diterapkan. Misalnya, antrian di bank sebagai contoh dari queue.
2. **Tugas Menulis:** Mahasiswa diminta untuk menulis esai singkat tentang pentingnya struktur data dalam pemrograman dan memberikan satu contoh kasus di mana penggunaan struktur data tertentu sangat diperlukan.

D. Rangkuman

Pada intinya, struktur data adalah fondasi dari banyak algoritma dalam pemrograman. Tanpa pemahaman yang mendalam tentang struktur data, akan sulit untuk merancang algoritma yang efisien dan efektif dalam memecahkan masalah komputasi. Oleh karena itu, mempelajari dan memahami konsep-konsep dasar struktur data adalah langkah penting bagi setiap programmer yang ingin mengembangkan keterampilan pemrograman yang kuat dan kompetitif. Implementasi

Pemahaman tentang konsep dasar struktur data sangat penting bagi programmer. Struktur data yang tepat dapat membuat algoritma lebih efisien dan program lebih cepat. Dengan mengenal jenis-jenis struktur data dan aplikasinya, kita dapat memilih alat yang tepat untuk menyelesaikan berbagai masalah pemrograman.

Struktur Dasar Python

A. Pendahuluan

Modul ini dirancang untuk memperkenalkan dan mengajarkan penggunaan struktur dasar dalam Python, termasuk variabel, tipe data, operator, serta struktur data kompleks seperti List, Tuple, Dictionary, dan Set. Relevansi modul ini adalah untuk memastikan bahwa mahasiswa memiliki fondasi yang kuat dalam penggunaan Python sebagai alat pemrograman. Dengan mengikuti modul ini, mahasiswa diharapkan mampu memahami dan mengimplementasikan dasar-dasar Python, serta memanfaatkan berbagai tipe data dan struktur data dalam pemrograman.

B. Deskripsi Materi

Struktur dasar Python mencakup beberapa komponen penting yang membentuk fondasi pemrograman dalam bahasa ini. Variabel digunakan untuk menyimpan data dan tidak memerlukan deklarasi tipe data secara eksplisit. Tipe data bawaan Python meliputi integer, float, string, dan boolean, yang memungkinkan penyimpanan berbagai jenis informasi. Operator digunakan untuk melakukan operasi aritmatika, perbandingan, dan logika. List adalah struktur data yang menyimpan elemen dalam urutan yang dapat diubah, sedangkan tuple, yang mirip dengan list, bersifat immutable dan tidak dapat diubah setelah dibuat. Dictionary menyimpan pasangan key-value yang memungkinkan akses data yang efisien, dan set adalah koleksi elemen unik tanpa urutan yang mendukung operasi matematika seperti union dan intersection. Kombinasi dari komponen-komponen ini memberikan dasar yang kuat untuk manipulasi data dan pengembangan program dalam Python.

C. Penjelasan Materi

1. Variabel

Variabel adalah tempat penyimpanan untuk data yang dapat digunakan dalam program. Python menggunakan penugasan variabel untuk menyimpan nilai dan tidak memerlukan deklarasi tipe data secara eksplisit.

Sintaks Dasar:

```
nama_variabel = nilai
```

Contoh:

```
# Mendeklarasikan variabel
x = 10
y = 3.14
name = "Alice"
is_valid = True
print(x, y, name, is_valid)
```

Petunjuk:

- Gunakan tanda = untuk menetapkan nilai ke variabel.
- Variabel dapat menyimpan berbagai jenis data seperti integer, float, string, dan boolean.

2. Tipe Data

Tipe data menentukan jenis nilai yang dapat disimpan dalam variabel. Python mendukung beberapa tipe data bawaan.

Sintaks Dasar:

```
# Integer
angka = 10
# Float
pecahan = 3.14
# String
teks = "Hello, World!"
# Boolean
status = True
```

Contoh:

```
# Tipe data
integer = 25
```

```
float_number = 5.75
string = "Hello"
boolean = False

print(integer, float_number, string, boolean)
```

Petunjuk:

- Gunakan tipe data yang sesuai dengan kebutuhan data.
- Tipe data dapat diubah selama runtime, tetapi harus konsisten dalam penggunaan.

3. Operator

Operator digunakan untuk melakukan operasi pada nilai atau variabel. Python memiliki berbagai operator aritmatika, perbandingan, dan logika.

Sintaks Dasar:

```
# Operator Aritmatika
a + b # Penjumlahan
a - b # Pengurangan
a * b # Perkalian
a / b # Pembagian

# Operator Perbandingan
a == b # Sama dengan
a != b # Tidak sama dengan
a > b # Lebih besar dari

# Operator Logika
a and b # Dan
a or b # Atau
not a # Tidak
```

Contoh:

```
a = 10
b = 5

print("a + b =", a + b) # Penjumlahan
print("a == b =", a == b) # Perbandingan
```

```
print("a > b and b > 0 =", a > b and b > 0) # Logika
```

Petunjuk:

- Gunakan operator yang sesuai dengan jenis operasi yang ingin dilakukan.
- Hati-hati dengan urutan operasi, gunakan tanda kurung jika diperlukan.

4. List

List adalah struktur data yang menyimpan koleksi item dalam urutan tertentu. List bersifat mutable, yang berarti elemen di dalamnya dapat diubah.

Sintaks Dasar:

```
list_name = [item1, item2, item3]
```

Contoh:

```
# Membuat list
fruits = ["apple", "banana", "cherry"]

# Menambahkan elemen
fruits.append("orange")

# Menghapus elemen
fruits.remove("banana")

# Mengakses elemen
print(fruits[0]) # Output: apple
print(fruits[-1]) # Output: orange
print(fruits[1:3]) # Output: ['cherry', 'orange']
```

Petunjuk:

- Gunakan metode `append()` untuk menambahkan elemen.
- Gunakan metode `remove()` untuk menghapus elemen.
- Gunakan indexing dan slicing untuk mengakses elemen.

5. Tuple

Tuple adalah struktur data yang mirip dengan list tetapi bersifat immutable, artinya tidak bisa diubah setelah dibuat.

Sintaks Dasar:

```
tuple_name = (item1, item2, item3)
```

Contoh:

```
# Membuat tuple
coordinates = (10, 20)

# Mengakses elemen tuple
print(coordinates[0]) # Output: 10

# Unpacking tuple
x, y = coordinates
print(x, y) # Output: 10 20
```

Petunjuk:

- Gunakan tuple ketika data tidak perlu diubah.
- Tuple dapat di-unpack ke dalam variabel terpisah.

6. Dictionary

Dictionary adalah struktur data yang menyimpan pasangan key-value. Key harus unik, dan setiap key berhubungan dengan satu value.

Sintaks Dasar:

```
dict_name = {
    "key1": value1,
    "key2": value2
}
```

Contoh:

```
# Membuat dictionary
person = {
    "name": "Alice",
    "age": 25,
    "city": "Yogyakarta"
}
```

```
# Mengakses value berdasarkan key
print(person["name"]) # Output: Alice

# Menambahkan pasangan key-value
person["email"] = "alice@example.com"

# Menghapus pasangan key-value
del person["age"]

print(person)
```

Petunjuk:

- Gunakan key untuk mengakses nilai dalam dictionary.
- Dictionary bersifat mutable, artinya bisa diubah setelah dibuat.

7. Set

Set adalah koleksi elemen unik yang tidak mempertahankan urutan. Set dapat digunakan untuk operasi matematika seperti union, intersection, dan difference.

Sintaks Dasar:

```
set_name = {item1, item2, item3}
```

Contoh:

```
# Membuat set
unique_numbers = {1, 2, 3, 4, 5}

# Menambahkan elemen
unique_numbers.add(6)

# Menghapus elemen
unique_numbers.remove(3)

# Operasi set
set_a = {1, 2, 3}
set_b = {3, 4, 5}

# Union
```

```
print(set_a.union(set_b)) # Output: {1, 2, 3, 4, 5}

# Intersection
print(set_a.intersection(set_b)) # Output: {3}
```

Petunjuk:

- Gunakan metode `add()` untuk menambahkan elemen.
- Gunakan metode `remove()` untuk menghapus elemen.
- Gunakan metode `union()`, `intersection()`, dan `difference()` untuk operasi set.

D. Latihan**Soal 1:**

Buatlah list yang berisi nama-nama warna dan lakukan operasi berikut:

1. Tambahkan warna "purple" ke dalam list.
2. Hapus warna "red" dari list.
3. Cetak elemen pertama dan terakhir dari list.

Petunjuk:

- Gunakan metode `append()` untuk menambahkan elemen.
- Gunakan metode `remove()` untuk menghapus elemen.
- Gunakan indexing untuk mengakses elemen pertama dan terakhir.

Contoh Hasil:

```
colors = ["red", "blue", "green"]
colors.append("purple")
colors.remove("red")
print(colors[0]) # Output: blue
print(colors[-1]) # Output: purple
```

Soal 2:

Buatlah tuple yang berisi tiga angka dan lakukan unpacking untuk mendapatkan nilai dari ketiga angka tersebut.

Petunjuk:

- Buat tuple dengan tiga angka.
- Unpack tuple ke dalam tiga variabel terpisah.

Contoh Hasil:

```
numbers = (1, 2, 3)
a, b, c = numbers
print(a, b, c) # Output: 1 2 3
```

Soal 3:

Buatlah sebuah dictionary yang berisi informasi tentang sebuah buku, seperti judul, penulis, dan tahun terbit. Tambahkan informasi penerbit, kemudian cetak seluruh informasi.

Petunjuk:

- Gunakan pasangan key-value untuk membuat dictionary.
- Tambahkan informasi baru dengan key yang sesuai.
- Cetak seluruh dictionary.

Contoh Hasil:

```
book = {
    "title": "Python Programming",
    "author": "John Doe",
    "year": 2022
}
book["publisher"] = "Tech Books"
print(book)
```

Soal 4:

Buatlah sebuah set yang berisi angka-angka dari 1 hingga 5. Tambahkan angka 7, hapus angka 2, dan cetak hasil union dengan set yang berisi angka 5 hingga 8.

Petunjuk:

- Gunakan metode `add()` untuk menambahkan elemen.
- Gunakan metode `remove()` untuk menghapus elemen.
- Gunakan metode `union()` untuk melakukan union dengan set lain.

Contoh Hasil:

```
numbers_set = {1, 2, 3, 4, 5}
numbers_set.add(7)
numbers_set.remove(2)
other_set = {5, 6, 7, 8}
print(numbers_set.union(other_set)) # Output: {1, 3, 4, 5, 6, 7, 8}
```

E. Tugas

1. Latihan List:

- Buatlah list yang berisi nama-nama hewan dan lakukan operasi berikut:
 1. Tambahkan hewan "tiger" ke dalam list.
 2. Hapus hewan "cat" dari list.
 3. Cetak elemen ketiga dari list.
- **Petunjuk:**
 - Gunakan metode `append()` untuk menambahkan elemen.
 - Gunakan metode `remove()` untuk menghapus elemen.
 - Gunakan indexing untuk mengakses elemen ketiga.

2. Latihan Tuple:

- Buatlah tuple yang berisi tiga nama negara dan lakukan unpacking untuk mendapatkan nama negara pertama, kedua, dan ketiga ke dalam variabel terpisah.
- **Petunjuk:**
 - Buat tuple dengan tiga nama negara.
 - Unpack tuple ke dalam variabel terpisah dan cetak nilai masing-masing.

3. Latihan Dictionary:

- Buatlah sebuah dictionary yang berisi informasi tentang seorang mahasiswa, seperti nama, usia, dan jurusan. Tambahkan informasi tentang alamat, kemudian cetak seluruh informasi.
- **Petunjuk:**
 - Gunakan pasangan key-value untuk membuat dictionary.
 - Tambahkan key baru dengan metode update() atau dengan penugasan langsung.
 - Cetak seluruh dictionary.

4. Latihan Set:

- Buatlah sebuah set yang berisi angka dari 1 hingga 5. Tambahkan angka 10, hapus angka 4, dan cetak hasil intersection dengan set yang berisi angka 3 hingga 6.
- **Petunjuk:**
 - Gunakan metode add() untuk menambahkan elemen.
 - Gunakan metode remove() untuk menghapus elemen.
 - Gunakan metode intersection() untuk menemukan elemen yang sama antara dua set.

F. Rangkuman

1. Variabel: Tempat penyimpanan data dalam program yang dapat menyimpan berbagai tipe data seperti integer, float, string, dan boolean. Variabel di Python tidak memerlukan deklarasi tipe data eksplisit.
2. Tipe Data: Python memiliki berbagai tipe data bawaan seperti integer (bilangan bulat), float (bilangan pecahan), string (teks), dan boolean (True/False). Pilih tipe data yang sesuai untuk data yang akan disimpan.
3. Operator: Digunakan untuk melakukan operasi aritmatika (seperti +, -, *, /), perbandingan (seperti ==, !=, >), dan logika (seperti and, or, not). Operator memanipulasi nilai dan variabel dalam program.
4. List: Struktur data yang menyimpan koleksi elemen dalam urutan. List bersifat mutable, artinya elemen di dalamnya dapat diubah. Akses elemen menggunakan indeks dan lakukan operasi seperti penambahan, penghapusan, dan slicing.
5. Tuple: Mirip dengan list tetapi bersifat immutable. Tuple tidak dapat diubah setelah dibuat dan dapat digunakan untuk menyimpan data yang tidak perlu diubah. Gunakan unpacking untuk memisahkan elemen tuple ke dalam variabel.
6. Dictionary: Struktur data yang menyimpan pasangan key-value. Key harus unik dan digunakan untuk mengakses nilai yang berhubungan dengan key tersebut. Dictionary bersifat mutable dan memungkinkan penambahan serta penghapusan pasangan key-value.
7. Set: Struktur data yang menyimpan elemen unik tanpa urutan. Gunakan set untuk operasi matematika seperti union, intersection, dan difference. Set bersifat mutable.

Konsep Dasar Struktur Data List

A. Pendahuluan

Modul ini dirancang untuk mengajarkan penggunaan Tuple sebagai struktur data yang efisien dalam Python. Relevansi modul ini terletak pada pentingnya Tuple sebagai struktur data yang tidak dapat diubah, yang berguna dalam banyak aplikasi pemrograman yang membutuhkan data konstan. Dengan mengikuti modul ini, mahasiswa diharapkan mampu memahami dan mengimplementasikan Tuple dalam Python.

B. Deskripsi Materi

Tuple adalah struktur data yang mirip dengan List, namun memiliki sifat tidak dapat diubah (immutable). Modul ini akan membahas bagaimana cara kerja Tuple, operasi dasar seperti indexing, slicing, dan unpacking, serta kapan dan mengapa menggunakan Tuple dalam pemrograman.

Contoh:

```
# Membuat Tuple dan operasi dasar
my_tuple = (10, 20, 30)
print(my_tuple[0]) # Output: 10
```

C. Implementasi

Mahasiswa akan mengimplementasikan Tuple dalam aplikasi nyata, seperti menyimpan koordinat dalam grafik atau data yang tidak boleh berubah.

D. Latihan dan Tugas

- Latihan: Buatlah program yang menggunakan Tuple untuk menyimpan dan menampilkan data.
- Tugas: Implementasikan Tuple dalam program yang membutuhkan data konstan, misalnya menyimpan koordinat titik pada grafik.

E. Rangkuman

Modul ini memberikan pemahaman tentang penggunaan Tuple sebagai struktur data yang efisien untuk menyimpan data yang tidak berubah.

Konsep Dasar Struktur Dictionary

A. Pendahuluan

Modul ini berfokus pada penggunaan Dictionary dalam Python, sebuah struktur data yang sangat berguna untuk menyimpan dan mengakses data dengan cepat menggunakan pasangan kunci-nilai (key-value pairs). Relevansi modul ini adalah kemampuan Dictionary untuk menyimpan data yang memiliki relasi dan dapat diakses secara efisien. Mahasiswa diharapkan mampu memahami operasi dasar Dictionary dan mengimplementasikannya dalam berbagai aplikasi pemrograman.

B. Deskripsi Materi

Dictionary adalah struktur data yang memungkinkan penyimpanan data dalam bentuk pasangan kunci-nilai, di mana setiap kunci unik terkait dengan satu nilai. Modul ini akan membahas operasi dasar seperti menambah, menghapus, dan memperbarui pasangan kunci-nilai, serta bagaimana menggunakan Dictionary dalam pemrograman sehari-hari.

Contoh:

```
# Membuat Dictionary dan operasi dasar
student = {"name": "Alice", "age": 21}
print(student["name"]) # Output: Alice
student["age"] = 22 # Memperbarui nilai pada key 'age'
```

C. Implementasi

Mahasiswa akan diminta untuk mengimplementasikan Dictionary dalam skenario nyata, seperti menyimpan data mahasiswa atau informasi kontak.

D. Latihan dan Tugas

- Latihan: Buatlah program yang menyimpan data mahasiswa dalam Dictionary, dan buatlah fungsi untuk mengupdate dan menampilkan data tersebut.

- Tugas: Implementasikan Dictionary untuk menyimpan data kontak dengan pasangan nama dan nomor telepon, serta tambahkan fitur pencarian.

E. Rangkuman

Modul ini memberikan pemahaman tentang bagaimana Dictionary dapat digunakan untuk mengelola data yang terhubung melalui kunci unik.

Konsep Dasar Struktur Data Set

A. Pendahuluan

Modul ini dirancang untuk memperkenalkan penggunaan Set dalam Python, sebuah struktur data yang sangat efisien untuk menyimpan elemen unik. Relevansi modul ini adalah kemampuan Set untuk mengelola data yang tidak boleh memiliki duplikasi, yang sangat penting dalam banyak aplikasi pemrograman. Dengan mengikuti modul ini, mahasiswa diharapkan mampu memahami dan mengimplementasikan Set dalam berbagai aplikasi.

B. Deskripsi Materi

Set adalah struktur data yang menyimpan elemen-elemen unik dan tidak terurut. Modul ini akan membahas operasi dasar seperti penambahan, penghapusan, serta operasi himpunan seperti union, intersection, dan difference. Mahasiswa akan belajar bagaimana menggunakan Set untuk menyelesaikan masalah yang melibatkan data unik.

Contoh:

```
# Membuat Set dan operasi dasar
unique_numbers = {1, 2, 3, 3, 4}
print(unique_numbers) # Output: {1, 2, 3, 4}
unique_numbers.add(5) # Menambah elemen ke Set
```

C. Implementasi

Mahasiswa diminta untuk mengimplementasikan Set dalam skenario nyata, seperti menghilangkan elemen duplikat dari sebuah daftar atau mengelola data unik.

D. Latihan dan Tugas

Latihan: Buatlah program yang menggunakan Set untuk menyimpan data unik dan melakukan operasi himpunan.

Tugas: Implementasikan Set dalam program yang memfilter elemen duplikat dari daftar besar.

E. Rangkuman

Modul ini memberikan pemahaman mendalam tentang bagaimana Set dapat digunakan untuk mengelola data unik dan melakukan operasi himpunan.

Konsep Dasar Array dan Linked List

A. Pendahuluan

Modul ini dirancang untuk memberikan pemahaman mendalam tentang dua struktur data dasar yang penting: Array dan Linked List. Relevansi modul ini terletak pada kemampuan mahasiswa untuk memahami perbedaan antara kedua struktur data ini dan mengimplementasikannya dalam Python. Dengan mengikuti modul ini, mahasiswa diharapkan mampu memilih dan menggunakan struktur data yang tepat berdasarkan kebutuhan aplikasi serta memahami kelebihan dan kekurangan masing-masing struktur data.

B. Deskripsi Materi

- 1. Array:** Array adalah struktur data yang menyimpan elemen-elemen dengan tipe yang sama dalam urutan yang terkontrol. Dalam Python, Array biasanya diimplementasikan menggunakan List. Array memungkinkan akses elemen secara langsung melalui indeks, yang membuatnya sangat efisien untuk operasi akses data.

Contoh:

```
# Membuat Array menggunakan List
array = [10, 20, 30, 40, 50]
print(array[2]) # Output: 30
```

- 2. Linked List:** Linked List adalah struktur data yang terdiri dari node-node yang masing-masing menyimpan data dan referensi ke node berikutnya. Linked List memungkinkan penyisipan dan penghapusan elemen yang efisien, tetapi akses elemen dilakukan secara sekuensial, yang bisa lebih lambat dibandingkan Array.

Contoh:

```
class Node:
    def __init__(self, data):
        self.data = data
```

```
        self.next = None

class LinkedList:
    def __init__(self):
        self.head = None

    def append(self, data):
        new_node = Node(data)
        if not self.head:
            self.head = new_node
        else:
            current = self.head
            while current.next:
                current = current.next
            current.next = new_node

    def display(self):
        elements = []
        current = self.head
        while current:
            elements.append(current.data)
            current = current.next
        return elements
```

C. Implementasi

Mahasiswa diminta untuk mengimplementasikan Array dan Linked List dalam berbagai skenario pemrograman, seperti menyimpan dan mengelola data yang sering berubah.

D. Latihan dan Tugas

- Latihan: Implementasikan Array menggunakan List dan Linked List. Buatlah fungsi untuk menambah, menghapus, dan menampilkan elemen pada kedua struktur data tersebut.
- Tugas: Buatlah aplikasi yang memanfaatkan Array dan Linked List untuk menyimpan data pengguna, termasuk kemampuan untuk menambahkan, menghapus, dan mencari data.

E. Rangkuman

Modul ini menjelaskan perbedaan antara Array dan Linked List, serta cara mengimplementasikannya dalam Python. Array menawarkan akses langsung dan efisiensi dalam operasi baca, sedangkan Linked List menawarkan fleksibilitas dalam penambahan dan penghapusan elemen.

Konsep Dasar Struktur Data List

A. Pendahuluan

Modul ini dirancang untuk memberikan pemahaman tentang dua struktur data penting lainnya: Stack dan Queue. Relevansi modul ini terletak pada kemampuan mahasiswa untuk memahami prinsip-prinsip dasar Stack dan Queue serta mengimplementasikan struktur data ini dalam Python. Dengan mengikuti modul ini, mahasiswa diharapkan mampu menggunakan Stack dan Queue dalam berbagai aplikasi pemrograman, seperti dalam algoritma dan pengolahan data.

B. Deskripsi Materi

- 1. Stack:** Stack adalah struktur data yang mengikuti prinsip Last In, First Out (LIFO), di mana elemen terakhir yang dimasukkan adalah yang pertama dihapus. Stack sering digunakan dalam aplikasi seperti undo/redo, pemrosesan ekspresi matematika, dan rekursi.

Contoh:

```
class Stack:
    def __init__(self):
        self.stack = []

    def push(self, item):
        self.stack.append(item)

    def pop(self):
        if not self.is_empty():
            return self.stack.pop()
        else:
            raise IndexError("Pop from an empty stack")

    def peek(self):
        if not self.is_empty():
```

```
        return self.stack[-1]
    else:
        raise IndexError("Peek from an empty stack")

def is_empty(self):
    return len(self.stack) == 0
```

2. **Queue:** Queue adalah struktur data yang mengikuti prinsip First In, First Out (FIFO), di mana elemen pertama yang dimasukkan adalah yang pertama dihapus. Queue sering digunakan dalam antrian proses, penjadwalan tugas, dan buffer data.

Contoh:

```
class Queue:
    def __init__(self):
        self.queue = []

    def enqueue(self, item):
        self.queue.append(item)

    def dequeue(self):
        if not self.is_empty():
            return self.queue.pop(0)
        else:
            raise IndexError("Dequeue from an empty queue")

    def peek(self):
        if not self.is_empty():
            return self.queue[0]
        else:
            raise IndexError("Peek from an empty queue")

    def is_empty(self):
        return len(self.queue) == 0
```

C. Implementasi

Mahasiswa akan diminta untuk mengimplementasikan Stack dan Queue dalam skenario nyata, seperti manajemen tugas atau simulasi antrian.

D. Latihan dan Tugas

- Latihan: Implementasikan Stack dan Queue, serta buatlah fungsi untuk menambah, menghapus, dan memeriksa elemen pada kedua struktur data.
- Tugas: Buatlah program yang memanfaatkan Stack untuk memproses ekspresi matematika (seperti notasi postfix) dan Queue untuk mengelola antrian tugas pada sistem.

E. Rangkuman

Modul ini memberikan pemahaman mendalam tentang Stack dan Queue sebagai struktur data yang penting dalam pemrograman, menjelaskan prinsip dasar mereka serta cara mengimplementasikan dan menggunakannya dalam berbagai aplikasi.

DAFTAR PUSTAKA

- A. P. S. Arifin, "Implementasi Algoritma Genetika pada Masalah Penjadwalan," *J. Teknologi dan Sistem Komputer*, vol. 10, no. 1, pp. 1–12, 2022.
- B. S. Nugroho, F. A. Hidayat, dan I. M. Putra, "Analisis Performa Algoritma Sorting pada Data Besar Menggunakan Python," *J. Komputer dan Informatika*, vol. 9, no. 2, pp. 15–25, 2021.
- Downey, A. B. (2015). *Think Python: How to Think Like a Computer Scientist* (2nd ed.). Green Tea Press.
- F. Rizki, "Tutorial Dasar Python untuk Pemula," diambil pada tanggal 10 Agustus 2024 dari <https://www.example.com/tutorial-python-pemula>.
- GeeksforGeeks. (2024). Data Structures. Retrieved from <https://www.geeksforgeeks.org/data-structures/>
- Hetland, M. L. (2019). *Python Algorithms*. Apress.
- J. R. Anwar, "Pengertian dan Implementasi Struktur Data," diambil pada tanggal 12 Agustus 2024 dari <https://www.example.com/pengertian-struktur-data>.
- Lutz, M. (2021). *Learning Python* (5th ed.). O'Reilly Media.
- Python Software Foundation. (2024). *Python Documentation*. Retrieved from <https://docs.python.org/3/>
- Van Rossum, G., & Drake, F. L. (2009). *The Python Language Reference*. Python Software Foundation.
- W3Schools. (2024). Python Data Structures. Retrieved from https://www.w3schools.com/python/python_data_structures.asp
- Zelle, J. M. (2017). *Python Programming: An Introduction to Computer Science* (3rd ed.). Franklin, Beedle & Associates Inc.