

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN**



Disusun Oleh :

Nama : Mochammad Ferdiansyah
NIM : 251080200060
Kelompok : 8

**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 - 2026**

LEMBAR PERSETUJUAN

Telah Diperiksa Dan Disetujui

Isi Laporan Ini

LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN

Disusun Oleh:

Nama : Mochammad Ferdiansyah
NIM : 251080200060
Kelompok : 8

Mengetahui,

Laboran Informatika

(Melina Atikawati, S.Kom.)

**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025-2026**

KATA PENGANTAR

Alhamdulillah segala puji syukur atas kehadiran Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penyusun dapat menyelesaikan Laporan Praktikum Algoritma dan Pemrograman ini tanpa halangan yang berarti.

Keberhasilan penyusun dalam menyusun Laporan Praktikum Algoritma dan Pemrograman ini tidak lepas dari bantuan berbagai pihak. Untuk itu saya selaku penyusun menyampaikan terimakasih yang sebesar-besarnya kepada :

1. **Ir. Iswanto, S.T., M.MT.**, selaku Dekan Fakultas Sains dan Teknologi Universitas Muhammadiyah Sidoarjo.
2. **Ade Eviyanti, S.Kom., M.Kom.**, selaku Kepala Program Studi Informatika Universitas Muhammadiyah Sidoarjo.
3. **Uce Indahyanti, S.Kom., M.Kom.**, selaku Dosen Praktikum Algoritma dan Pemrograman Universitas Muhammadiyah Sidoarjo.
4. **Melina Atikawati, S.Kom.**, selaku Laboran Informatika Universitas Muhammadiyah Sidoarjo.
5. **Muhamad Maulana Musafir**, selaku Asisten praktikum yang telah membantu terlaksananya Praktikum Algoritma dan Pemrograman.
6. Keluarga dan teman-teman yang telah memberi bantuan baik materi maupun spiritual dalam pembuatan Laporan Algoritma dan Pemrograman

Saya selaku penyusun menyadari bahwa laporan ini masih jauh dari kesempurnaan, untuk itu sangat mengharapkan kritik dan saran dari berbagai pihak yang sifatnya membangun.

Akhir kata, semoga laporan ini dapat menjadi referensi untuk menambah wawasan para pembaca dan tentunya dapat memberikan manfaat sebagaimana yang diharapkan.

Sidoarjo, 13 November 2025



(Mochammad Ferdiansyah)

DAFTAR ISI

KATA PENGANTAR.....	iii
DAFTAR ISI.....	iii
BAB I PENDAHULUAN.....	1
A. PROFIL LABORATORIUM ALGORITMA DAN PEMROGRAMAN	1
B. PENGGUNAAN LABORATORIUM ALGORITMA DAN PEMOGRAMAN ..	7
C. PERALATAN LABORATORIUM	7
D. PERALATAN PENDUKUNG	7
BAB II KURIKULUM.....	8
ANALISIS MATERI / INSTRUKSIONAL	8
SILABUS PRAKTIKUM	8
BAB III MATERI MODUL POKOK BAHASAN 1.....	13
PENDAHULUAN.....	13
PENYAJIAN (TUTORIAL)	13
LEMBAR KERJA DAN TUGAS.....	19
REFERENSI	21
POKOK BAHASAN 2	23
PENDAHULUAN.....	23
PENYAJIAN (TUTORIAL)	23
LEMBAR KERJA DAN TUGAS.....	23
REFERENSI	33
POKOK BAHASAN 3	35
LEMBAR KERJA DAN TUGAS.....	37
REFERENSI	47
POKOK BAHASAN 4 STRUKTUR PEMOGRAMAN : PERULANGAN	49
LEMBAR KERJA DAN TUGAS.....	51
REFERENSI	61
POKOK BAHASAN 5 TIPE DATA KOLEKSI	63

LEMBAR KERJA DAN TUGAS	69
POKOK BAHASAN 6 STRUKTUR PEMOGRAMAN : FUNCTION	78
LEMBAR KERJA DAN TUGAS	81
REFERENSI	82

BAB I

PENDAHULUAN

Profil Laboratorium Algoritma dan Pemrograman Visi Laboratorium

"Mewujudkan Laboratorium Algoritma dan Pemrograman yang bermutu Tingkat Nasional 2025 di Bidang Informatika dan Komputasi, serta Menjadi Unit Pendukung Kegiatan di Unit Kerja Lain di Lingkungan UMSIDA”.

Misi Laboratorium

1. Menyelenggarakan praktikum berkualitas, dan pembimbingan berdasarkan kurikulum yang terintegratif dan berkelanjutan untuk seluruh Mahasiswa Informatika.
2. Menyediakan sarana dan prasarana untuk kegiatan penelitian, pengembangan dan inovasi dalam bidang Informatika dan Komputer.
3. Memberikan pelayanan laboratorium yang baik dan bermutu.
4. Meningkatkan Kerjasama dengan instansi pemerintah/swasta /masyarakat di bidang Informatika dan Komputer di Tingkat Nasional.

Sarana Laboratorium

- Tersedianya sarana dan prasarana untuk penunjang kegiatan praktikum.
- Tersedianya asisten laboratorium dan asisten praktikum yang professional dan sesuai dengan tugasnya.
- Adanya koordinasi atau kerja sama yang baik antara unit-unit yang terkait.
- Meningkatkan pengelolaan data akademik.
- Meningkatkan pelayanan administrasi praktikum.

Tujuan Laboratorium

1. Mengembangkan sumber daya laboratorium terpadu untuk peningkatan kualitas pelayanan praktikum, penelitian, pembelajaran dan pengembangan masyarakat.
2. Mengembangkan sumber-sumber pendanaan laboratorium yang berkelanjutan.
3. Pengembangan sistem manajemen Laboratorium yang sehat dan harmonis.

4. Mengembangkan kelembagaan laboratorium yang kuat dan dinamis serta meningkatkan kapabilitas SDM.

Manfaat Laboratorium Algoritma Dan Pemrograman

1. Diharapkan dapat memfasilitasi pengembangan mata pelajaran TIK (Teknologi Informasi Komputer) sebagai bagian dasar pemanfaatan teknologi untuk mempersiapkan peserta didik yang memadai agar dimasa depan dapat berperan sebagai kontribusi dari penguasaan komputer.
2. Untuk menunjang proses pembelajaran yang bermutu, teratur dan berkelanjutan.
3. Meningkatkan pengalaman dan keterampilan dalam mengimplementasikan penguasaan komputer pada mata pelajaran lainnya.
4. Memberikan dampak kepada siswa untuk lebih terampil mengkomunikasikan teori dengan praktik dalam proses belajar-mengajar.
5. Memberikan pengalaman langsung kepada siswa melalui praktik-praktik lapangan.

SOP (Standard Operasional Prosedur)

SOP (Standard Operasional Prosedur) adalah suatu pedoman tertulis yang dipergunakan untuk memperlancar kegiatan praktikum teknik Informatika.

1. Tujuan

1. Meningkatkan efisiensi pelaksanaan kegiatan praktikum di laboratorium Teknik Informatika.
2. Memberikan sanksi bagi pengguna laboratorium yang tidak memenuhi aturan.

2. Pihak Terkait

- a. Mahasiswa;
- b. Dosen;
- c. Laboran;
- d. Asisten; dan
- e. Kepala Laboratorium;

3. Waktu Dan Tempat Pelaksanaan

Waktu praktikum sesuai dengan jadwal yang telah ditentukan. Tempat pelaksanaan praktikum di Laboratorium Komputasi.

c.1 Jumlah tatap muka

Jumlah tatap muka praktikum Algoritma dan Pemrograman di Laboratorium Algoritma dan Pemrograman menetapkan 6 kali tatap muka.

c.2 Lama praktikum setiap tatap muka

Lama praktikum untuk setiap tatap muka adalah 2 jam dengan pertimbangan bahwa setengah jam pertama untuk persiapan peralatan sedangkan satu setengah jam berikutnya untuk materi praktikum. Jeda waktu antar praktikum 15 menit dengan pertimbangan bahwa diperlukan waktu kurang lebih 15 menit bagi asisten untuk mempersiapkan pelaksanaan praktikum berikutnya.

Prosedur Penggunaan Laboratorium Algoritma Dan Pemograman

1. Tidak menginstal software pada komputer yang digunakan
 - a. Tidak menginstal dan menghapus
 - b. Tidak membuat akun, didirektori
 - c. Tidak menambah atau mengurangi data yang ada, kecuali yang dibutuhkan
2. Pelanggaran atas aturan ini dikenakan sanksi tidak dapat mengikuti praktikum berikutnya.
3. Asisten harus melaporkan terjadinya pelanggaran ke laboran dan mencatat pelanggaran yang terjadi.
4. Kerusakan karena kelalaian praktikan menjadi tanggung jawab praktikan yang bersangkutan.
5. Tidak membawa makanan dan minuman ke dalam laboratorium.

Prosedur Pelaksanaan Praktikum

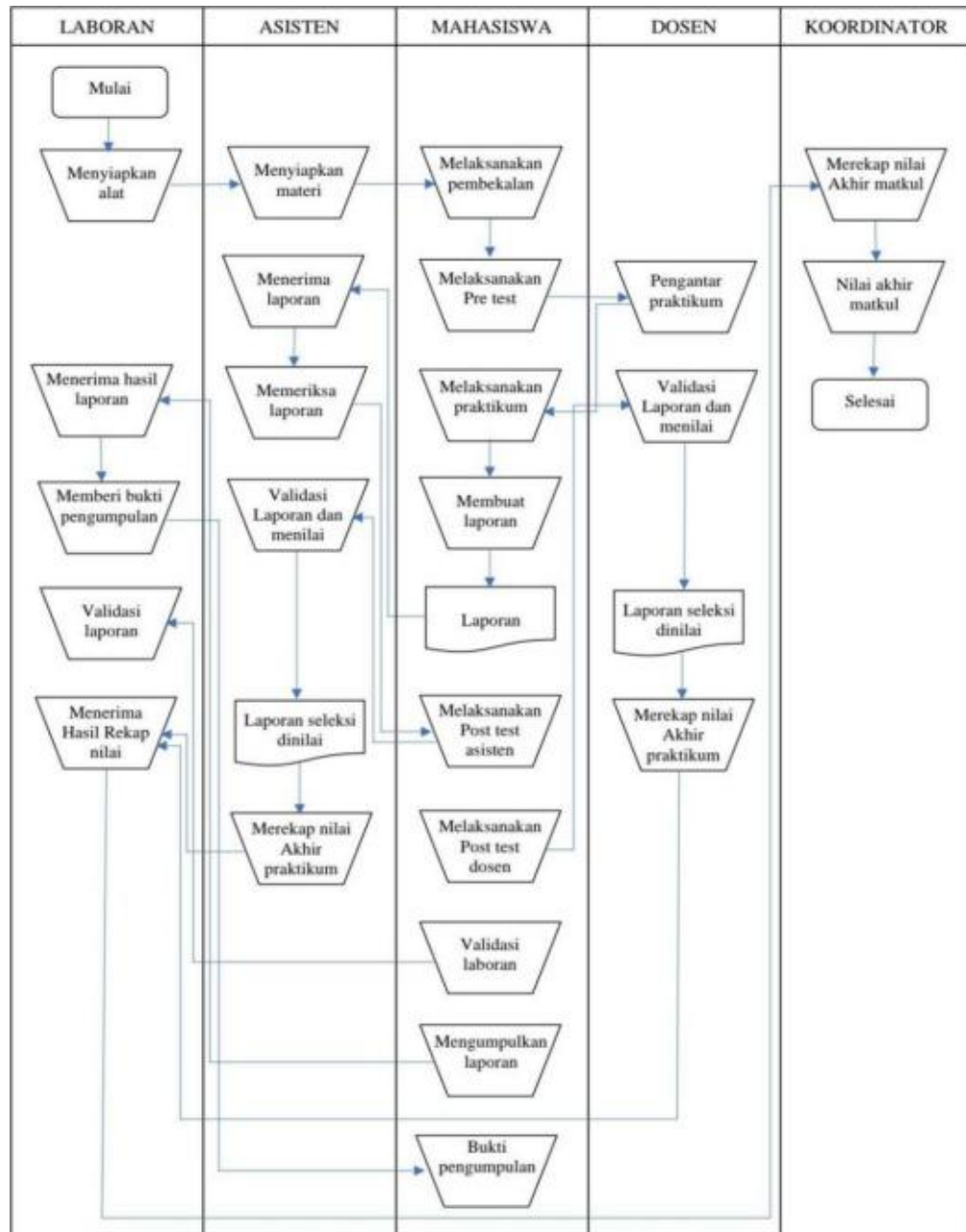
1. Laboran menyiapkan laboratorium dan perlengkapannya.
2. Asisten menyiapkan materi praktikum berdasarkan modul praktikum.
3. Mahasiswa melaksanakan praktikum didampingi asisten dan dosen pembimbing
4. Mahasiswa membuat laporan dan diserahkan kepada asisten pada pertemuan berikutnya.
5. Asisten memeriksa dan menandatangani asistensi laporan praktikum mahasiswa
6. Laporan yang telah diperiksa diserahkan kembali kepada mahasiswa
7. Pada akhir praktikum, Dosen memberikan tes/ujian yang harus diikuti oleh semua mahasiswa.
8. Dosen menyerahkan nilai hasil tes/ujian praktikum yang dilaksanakan mahasiswa ke Kepala Laboratorium.
9. Asisten merekap nilai praktikum (20% tugas + 40 % pelaksanaan praktikum)
10. Asisten menyerahkan nilai ke Kepala laboratorium.
11. Kepala laboratorium merekap nilai praktikum (20% tugas + 40 % pelaksanaan praktikum + 30 % ujian praktikum + 10 % laporan praktikum)
12. Kepala laboratorium menyerahkan nilai praktikum ke BAA.
13. Kalab membuat kartu peserta (lampiran) dengan ketentuan nilai :
Dinyatakan lulus praktikum jika nilai minimal praktikum C.

Tabel 1 Daftar Range Nilai

Nilai Huruf	Range Nilai
A	>85,1
AB	76-85
B	66-75
BC	56-65
C	46-55
D	36-45
E	<35

Flow Chart Pelaksanaan Praktikum

Langkah – langkah yang dilakukan oleh Laboran, Asisten, Mahasiswa, Dosen dan Kalab disajikan dalam diagram alir pada Gambar 1.



Gambar 1. Diagram alur langkah-langkah yang dilakukan oleh Laboran, Asisten, Mahasiswa, Dosen Praktikum, dan Koordinator pada pelaksanaan praktikum.

Tata Tertib Praktikum Laboratorium Fakultas Sains dan Teknologi Umsida

1. Praktikum dilaksanakan tepat waktu sesuai dengan jadwal yang ditetapkan.
2. Mahasiswa yang terlambat datang atau absen harus memberikan surat/bukti yang dapat dipercaya (surat dokter atau surat keterangan kerja dari perusahaan).
3. Mahasiswa diperkenankan pindah kelompok/jam/hari praktikum dengan syarat mengkonfirmasi 1 minggu sebelum melaksanakan praktikum melalui Koordinator Praktikum dan Kepala Lab.
4. Mahasiswa yang tidak hadir pada saat jadwal yang telah ditentukan diperkenankan mengikuti praktikum berikutnya dengan membayar denda Rp.25.000,- permodul praktikum selama proses praktikum masih berlangsung.
5. Mahasiswa harus berbusana yang sopan dan rapi (tidak diperkenankan memakai kaos oblong dan Sandal atau sepatu sandal)
6. Praktikum dianggap selesai jika mahasiswa telah menyerahkan laporan sementara dan alat yang dipinjam dalam keadaan baik, bersih, dan rapi.
7. Kerusakan alat yang dipinjam oleh mahasiswa menjadi tanggung jawab penuh kelompok mahasiswa yang bersangkutan.
8. Selama praktikum berlangsung, mahasiswa dilarang.. merokok, makan, bergurau, bermain alat, menghidupkan *hand phone*, atau pun keluar. masuk ruangan tanpa seijin dosen pembimbing / asisten pendamping.
9. Setelah melakukan praktikum, mahasiswa harus membuat laporan sementara hasil pengamatan praktikum rangkap dua dan menyerahkan kepada dosen pembimbing/asisten pada saat meninggalkan ruangan untuk ditanda tangani. (yang nantinya dilampirkan dalam laporan akhir).
10. Mahasiswa yang tidak melaksanakan praktikum 1 Modul dinyatakan tidak lulus.
11. Laporan Akhir Praktikum, *Cover-nya* menggunakan Standar Fakultas dan Laporan diserahkan 2 minggu setelah jadwal masing-masing kelompok.
12. Apabila Laporan diserahkan lebih dari 2 minggu maka dinyatakan TIDAK LULUS dan laporan Praktikum diserahkan ke koordinator praktikum dan kepala lab.

13. Mahasiswa yang dinyatakan tidak lulus Praktikum harus mengulang dijadwal praktikum berikutnya dengan membayar biaya praktikum yang telah ditentukan oleh Universitas melalui bank yang ditunjuk oleh UMSIDA.

Inventaris Laboratorium Algoritma dan Pemrograman

- Monitor : 12
- CPU : 12
- Mouse : 11
- Keyboard : 11
- Meja : 12
- Kursi : 25

Penggunaan Laboratorium

Laboratorium untuk praktikum Algoritma dan Pemrograman

Peralatan Laboratorium

Di Laboratorium Algoritma dan Struktur Data Fakultas Teknik Universitas Muhammadiyah Sidoarjo memiliki peralatan Laboratorium yang ada untuk menunjang pelaksanaan praktikum yang berupa :

1. Modul Praktikum Algoritma dan Struktur Data
2. Komputer
3. LCD
4. Papan Tulis

Peralatan Pendukung

Software yang digunakan dalam praktikum Algoritma dan Pemrograman adalah Flowgorithm dan Pycharm

BAB II

KURIKULUM

Analisis Materi / Instruksional

Mahasiswa diharapkan dapat:

- Memecahkan masalah menjadi sebuah algoritma (langkah-langkah) yang akan dijalankan oleh komputer, kemudian mengimplementasikannya menjadi sebuah program komputer.
- Memecahkan masalah pemrograman yang harus diselesaikan dengan materi yang ada pada pemrograman lanjut seperti pointer, struct, operasi file dsb.
- Mempresentasikan dua yang digunakan dalam pemrograman (baik data input atau data output) dengan struktur data yang tepat.
- Mengetahui & membandingkan macam-macam algoritma dalam proses pengurutan dan pencarian dan dapat menentukan algoritma yang digunakan dalam permasalahan pemrograman yang diselesaikannya.

Silabus Praktikum

Silabus praktikum algoritma dan struktur data:

1. Input/Output dan Jenis Data.
2. Struktur Pemrograman Pencabangan.
3. Struktur pemrograman: Lompatan dan Kalang.
4. Struktur Pemrograman: Kalang.
5. Variabel Larik: Matriks dan String.
6. Fungsi

A. Satuan Acara Praktikum (SAP)

Pertemuan Ke-	Tujuan Instruksional Khusus (TIK)	Topik	Sub Topik
1.	- Menjelaskan dan mendeklarasikan konsep dasar pada input/output dan jenis data. - Memahami operasi-operasi dan fungsi dari puts, scanf, gets.	Input/Output dan jenis data	- Membuat program dengan menggunakan fungsi puts. - Program tampilan kode escape. - Program mengisi input ke variable string - Program matematika bilangan bulat dan rill : pembagian - Program input ke variable bilangan : menghitung akar - Program Invers - Kepresisian bilangan rill - Pengatura tampilan - Input output data diri - Latihan

2.	Memahami Struktur pemrograman lompatan dan kalang	Struktur pemrograman pencabangan	- Pencabangan tunggal if. - Pencabangan ganda if. - Pencabangan bertingkat - Kondisi kombinatorial - Latihan
3.	Memahami Struktur pemrograman lompatan dan kalang	Struktur pemrograman : Lompatan Kalang	- Pencabangan - Pencabangan berganda : SWITCH - Lompatan - Kalang bersyarat - Latihan
4.	- Memahami representasi Queue dengan array dan pointer - Memahami operasi-operasi Queue menggunakan queue	Struktur pemrograman : kalang	- Tabel kuadrat - Memutus kalang BREAK dan CONTINUE - Tabel sinus - Tabel konfersi suhu - Bisection - Kalang
5.	Memahami konsep fungsi Variabel larik	Variabel larik : Matriks dan String	- Sekup variable kalang FOR - Kalang FOR untuk mengisi dan

			menampilkan matriks • Operasi String • Latihan
6.	• Memahami apa itu fungsi dalam algoritma pemrograman	Fungsi	• Fungsi untuk memotong program • Fungsi untuk mengisi dan menampilkan matriks • Program konversi suhu • Overloading • Fungsi penukaran • Menulis ke file teks



**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 - 2026**

Lembar Asistensi

Praktikum Algoritma dan Pemrograman

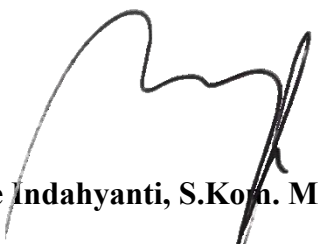
Pokok Bahasan 1

Judul : Flowchart dan Instalasi Python
Nama : Mochammad Ferdyansyah
NIM : 251080200060
Kelompok : 8
Dilaksanakan : 13 November 2025

Mengetahui,

Dosen Praktikum

Asisten Praktikum


(Uce Indahyanti, S.Kom. M.Kom.)

 19/11/2025
(Muhamad Maulana Musafir)

BAB III
MATERI MODUL
POKOK BAHASAN 1

Flowchart dan Instalasi Python

PENDAHULUAN

Pada pokok bahasan ini berisi penjelasan disertai contoh mengenai konsep Flowchart dan Python yang menjadi pemahaman dasar bagi mahasiswa sebelum mempelajari Algoritma dan Pemrograman dimana konsep ini nantinya digunakan untuk mempresentasikan sebuah program, diharapkan mahasiswa dapat :

1. Memahami bahasa pemrograman Python.
2. Mengetahui cara menginstall Python pada system operasi Windows.
3. Membuat program menggunakan bahasa pemrograman Python.
4. Memahami dan mengimplementasikan bahasa pemrograman.

PENYAJIAN (TUTORIAL)

1. Flowchart

a. Pengertian Flowchart

Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program. Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah.

Flowchart berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus. Selain itu dengan menggunakan bagan alur proses dari sebuah program akan lebih jelas, ringkas, dan mengurangi kemungkinan untuk salah penafsiran. Penggunaan flowchart dalam dunia pemrograman juga merupakan cara yang bagus untuk menghubungkan antara kebutuhan teknis dan non-teknis.

b. Fungsi Flowchart

Fungsi utama dari flowchart adalah memberi gambaran jalannya sebuah program dari suatu proses ke proses lainnya. Sehingga, alur program menjadi mudah dipahami oleh semua orang. Selain itu, fungsi lain dari flowchart

adalah untuk menyederhanakan rangkaian prosedur agar memudahkan pemahaman terhadap informasi tersebut.

c. Jenis Flowchart

Flowchart sendiri terdiri dari lima jenis, masing-masing jenis memiliki karakteristik dalam penggunaannya. Berikut adalah jenis-jenisnya:

- **Flowchart Dokumen**

Pertama ada flowchart dokumen (document flowchart) atau bisa disebut dengan paperwork flowchart. Flowchart dokumen berfungsi untuk menelusuri alur form dari satu bagian ke bagian yang lain, termasuk bagaimana laporan diproses, dicatat, dan disimpan.

- **Flowchart Program**

Selanjutnya kita akan membahas flowchart program. Flowchart ini menggambarkan secara rinci prosedur dari proses program. Flowchart program terdiri dari dua macam, antara lain: flowchart logika program (program logic flowchart) dan flowchart program komputer terinci (detailed computer program flowchart).

- **Flowchart Proses**

Flowchart proses adalah cara penggambaran rekayasa industrial dengan cara merinci dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem.

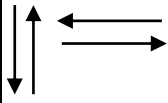
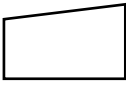
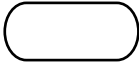
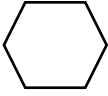
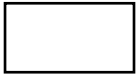
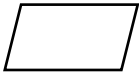
- **Flowchart Sistem**

Yang keempat ada flowchart sistem. Flowchart sistem adalah flowchart yang menampilkan tahapan atau proses kerja yang sedang berlangsung di dalam sistem secara menyeluruh. Selain itu flowchart sistem juga menguraikan urutan dari setiap prosedur yang ada di dalam sistem.

- **Flowchart Skematik**

Terakhir ada flowchart skematik. Flowchart ini menampilkan prosedur suatu sistem, hampir sama dengan flowchart sistem. Namun, ada perbedaan dalam penggunaan simbol-simbol dalam menggambarkan alur. Selain simbol-simbol, flowchart skematik juga menggunakan gambar-gambar komputer serta peralatan lainnya mempermudah dalam pembacaan flowchart untuk orang awam.

d. Simbol-Simbol Flowchart

	Flow Direction Symbol Yaitu simbol yang di gunakan untuk menghubungkan antara simbol yang lain. Simbol ini disebut juga connection line.		Simbol Manual Input Simbol untuk pemasukan data secara manual on-line keyboard.
	Terminator Symbol Yaitu simbol untuk pemulaan (start) atau akhir (stop) dari suatu kegiatan.		Simbol Preparation Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.
	Connector Symbol Yaitu simbol untuk keluar – masuk atau penyambungan proses dalam lembar / halaman yang sama.		Simbol Predefine Proses Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
	Connector Symbol Yaitu simbol untuk keluar – masuk atau penyambungan proses dalam lembar / halaman yang beda.		Simbol Display Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, dan sebagainya.
	Processing Symbol Simbol yang menunjukkan pengolahan yang di lakukan oleh komputer.		Simbol disk and On-line Storage Simbol yang menyatakan input yang berasal dari disk atau di simpan ke disk.
	Simbol Manual Operation Simbol yang menunjukkan pengolahan yang tidak di lakukan oleh computer.		Simbol magnetik tape unit Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
	Simbol Decision Simbol pemilihan proses berdasarkan kondisi yang ada.		Simbol Punch Card Simbol yang menyatakan bahwa input berasal dari kartu atau output ditulis ke kartu
	Simbol Input-Output Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.		Simbol Dokumen Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

Gambar 2. Simbol-simbol Flowchart

2. Instalasi Python 3.9.7 64-bit :

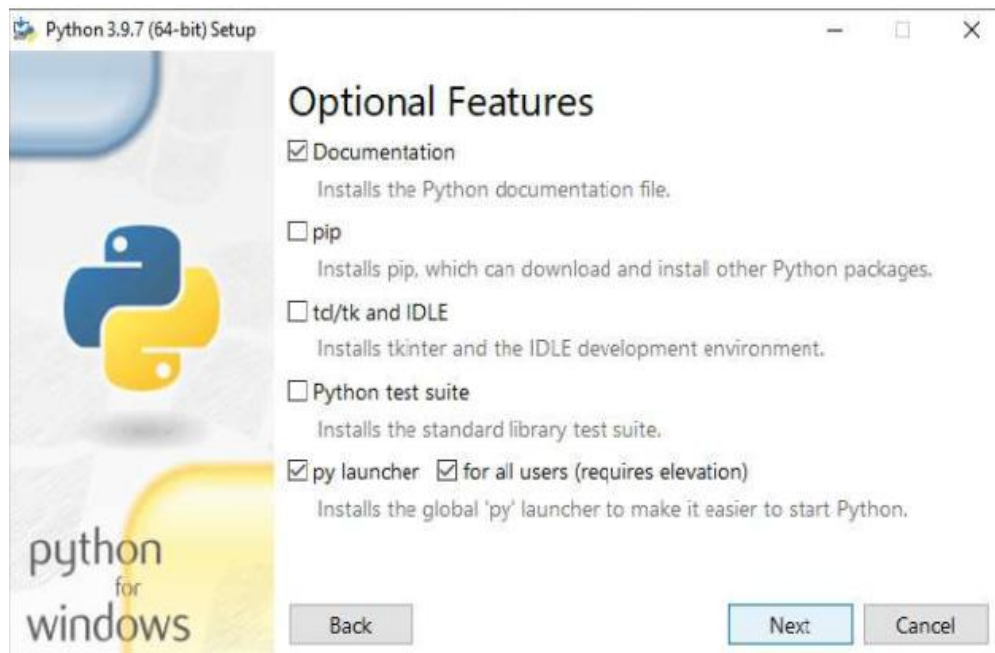
1. Jalankan file setup Python, yaitu **python-3.9.7-amd64** maka akan muncul dialog instalasi sebagai berikut



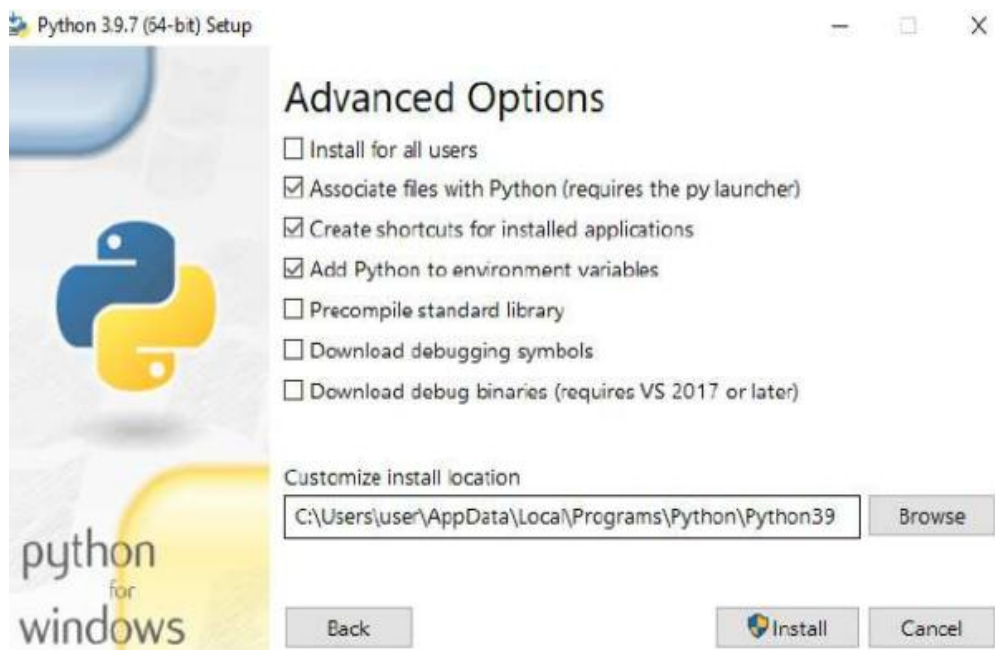
2. Centang kotak dialog **add Python 3.9 to PATH** lalu pilih **customize installation**



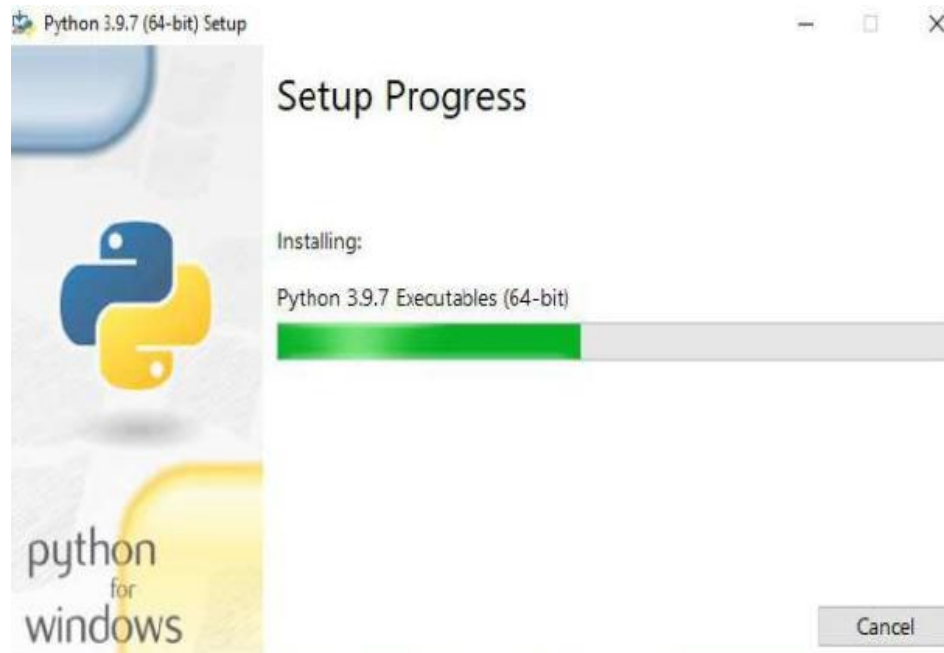
3. Lalu centang beberapa pilihan seperti di bawah ini



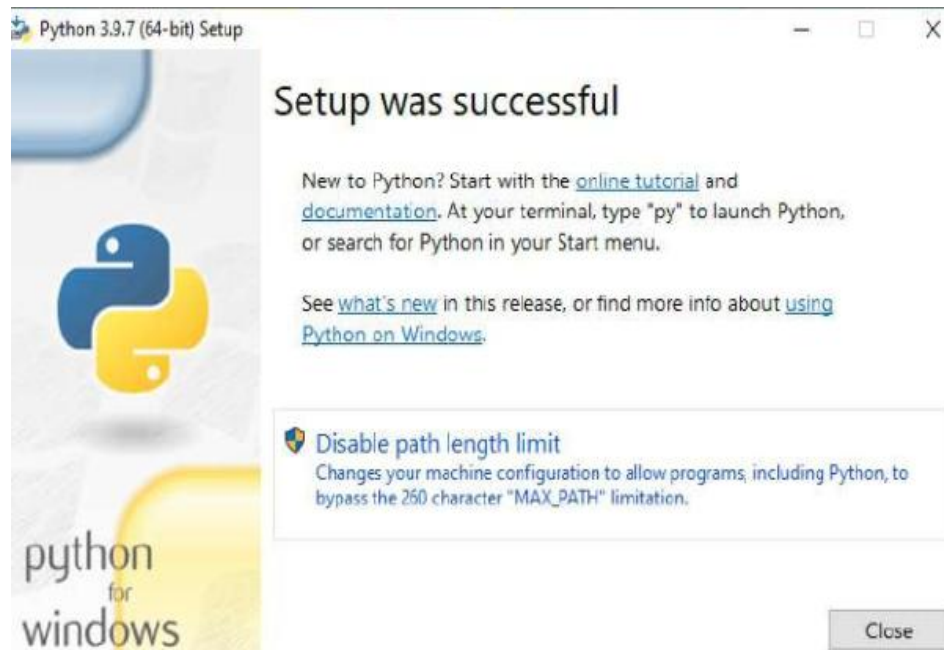
4. Klik Next, lalu muncul kotak dialog seperti di bawah



5. Lalu klik install, lalu muncul kotak dialog seperti di bawah.



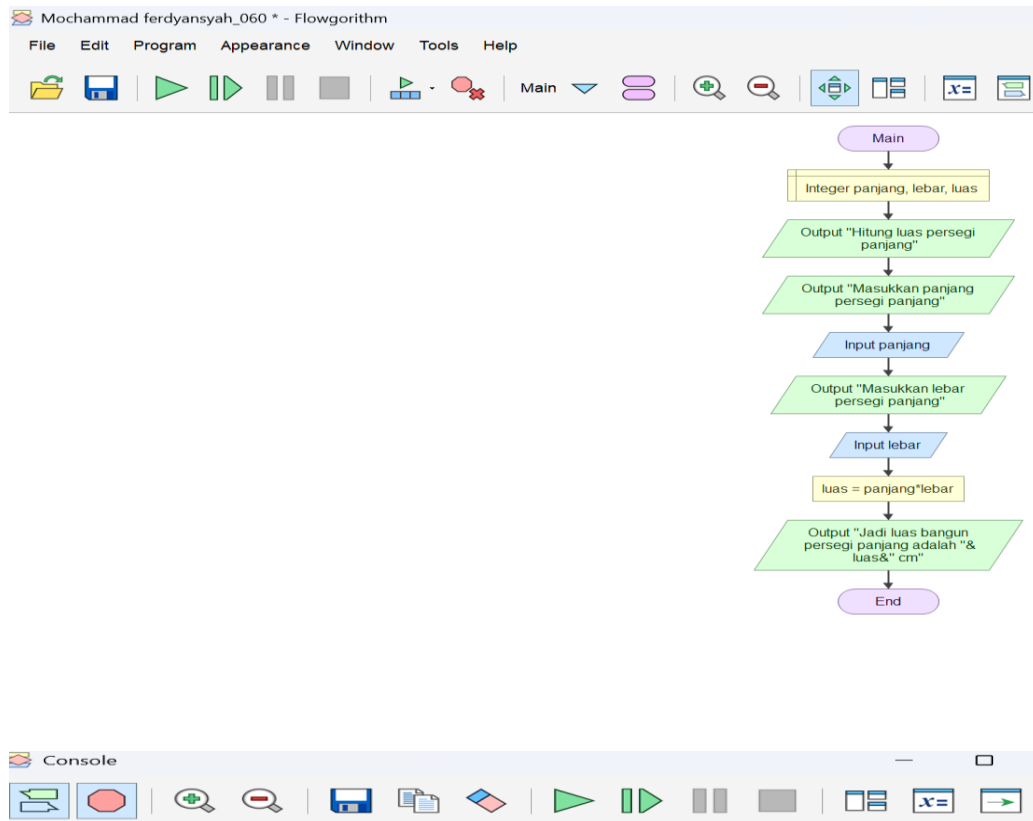
6. Maka Python selesai di install



LEMBAR KERJA DAN TUGAS

1. Membuat rancangan flowchart menggunakan flowgorithm yang menjelaskan cara menghitung sebuah luas bangun!

Flowchart menghitung luas bangun



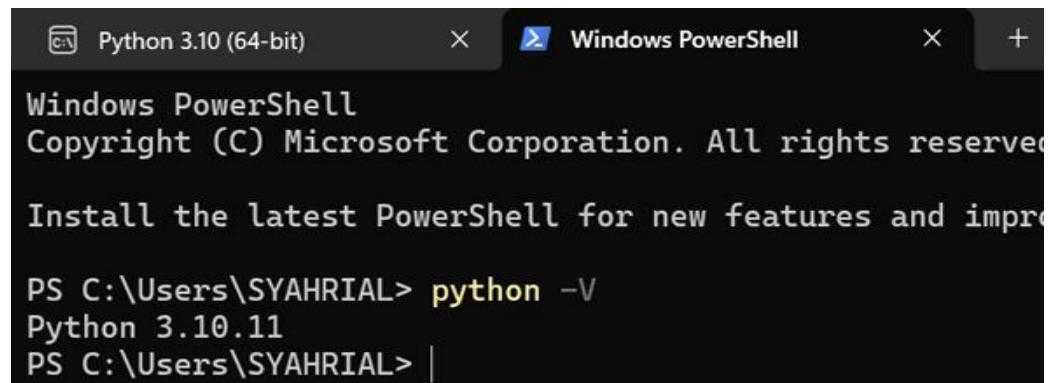
2. Memeriksa apakah Python sudah terinstall di perangkat anda.

```
Microsoft Windows [Version 10.0.26200.7171]
(c) Microsoft Corporation. All rights reserved.

C:\Users\SYAHRIAL>python
Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr  5 2023, 00:38:17) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

3. Memeriksa versi Python melalui CMD.

```
C:\Users\user>python -V
```



The screenshot shows a Windows PowerShell window with the following content:

```
Python 3.10 (64-bit) × Windows PowerShell × +
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PowerShellLatest

PS C:\Users\SYAHRIAL> python -V
Python 3.10.11
PS C:\Users\SYAHRIAL> |
```

REFERENSI

- <https://www.dicoding.com/blog/flowchart-adalah/>
- <https://informatikalogi.com/pengertian-flowchart-dan-jenis-jenisnya/>



**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 – 2026**

Lembar Asistensi

Praktikum Algoritma dan Pemrograman

Pokok Bahasan 2

Judul : Tipe Data dan Operator Aritmatika
Nama : Mochammad Ferdiansyah
NIM : 251080200060
Kelompok : 8
Dilaksanakan : 13 November 2025

Mengetahui,

Dosen Praktikum

Asisten Praktikum

(Uce Indahyanti, S.Kom. M.Kom.)

(Muhamad Maulana Musafir)

POKOK BAHASAN 2

Tipe Data dan Operator Aritmatika

PENDAHULUAN

PENYAJIAN (TUTORIAL)

LEMBAR KERJA DAN TUGAS

1. Mencoba perdana

Input:

```
print ("Hello World!")

def salam():
    print ("Assalamualaikum, Hai Semua\nNama saya adalah
ferdy\nsekarang saya belajar memprogram memakai bahasa
python\nAyo teman-teman belajar juga\nSampai Jumpa.")
salam()
```

Output:

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEM
1/Programs/Python/Python310/python.exe "d:/TUGAS INFORM
KUM/ferdy_060_mencoba-perdana.py"
Hello World!
Assalamualaikum, Hai Semua
Nama saya adalah ferdy
sekarang saya belajar memprogram memakai bahasa python
Ayo teman-teman belajar juga
Sampai Jumpa.
```

Pertanyaan : Apa kegunaan \n di atas ?

Jawaban : Untuk membuat sebuah baris baru dalam pemrograman bahasa python.

2. Program mengisi input ke variable integer

Input program 1:

```
NIM, Umur, Tinggi = (251080200060, 19, 170)
print(NIM, Umur, Tinggi)
```

Output program 1:

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEM
1/Programs/Python/Python310/python.exe "d:/TUGA
KUM/ferdy_060_program-variabel-integer.py"
251080200060 19 170
```

Pertanyaan : Apa yang terjadi bila tidak menggunakan tanda () ?

Jawaban : Jika tidak menggunakan tanda () di bagian atas tidak berpengaruh apapun, namun tanda () di bawah di hapus akan menghasilkan syntax error.

Input program 2:

```
angka_positif=("1,2,3,4,5,6,7,8,9")  
print(angka_positif)
```

Output program 2:

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA S  
1/Programs/Python/Python310/python.exe "d:/T  
KUM/ferdy_060_program-variabel-integer.py"  
(1, 2, 3, 4, 5, 6, 7, 8, 9)
```

Pertanyaan: Mengapa saat print("angka_positif") bilangan yang muncul juga meliputi () dan (.) ?

Jawaban: karena variable yang di inputkan pada perintah print(angka_positif) tetap mengikuti variable (1,2,3,4,5,6,7,8,9), jika pada variable (1,2,3,4,5,6,7,8,9) ditambahkan tanda petik di dalam kurungnya maka outputnya yang di hasilkan juga berbeda, outputnya yaitu hanya 1,2,3,4,5,6,7,8,9 tanda () menghilang.

3. Program variable string

Input program 1:

```
nama = input('Isikan nama kamu :')  
print('Hallo,', nama)  
print("Betulkah nama mu %s?\n"%nama)
```

Output program 1:

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA  
1/Programs/Python/Python310/python.exe "d:  
KUM/ferdy_060_program-variabel-string.py"  
Isikan nama kamu :ferdy  
Hallo, ferdy  
Betulkah nama mu ferdy?
```

Pertanyaan: Apa kegunaan %s pada script tersebut ?

Apa arti input pada program di atas ?

Jawaban: Perbedaannya adalah apabila array [0] akan menyebutkan hanya 1 huruf, sedangkan [8:29] akan menyebutkan huruf ke 9 hingga huruf ke 30 dikarenakan array dimulai dari 0

Input program 2:

```
Nama = 'Budi Eka'
Pesan = "Selamat belajar bahasa python"

print ("Nama:", Nama[0])
print ("Pesan:", Pesan[0:29])
```

Output 2:

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\KUM/ferdy_060_program-variabel-string.py"
Nama: B
Pesan: Selamat belajar bahasa python
```

Pertanyaan 1 : Apa arti angka 0 pada Nama[0] di atas?

Pertanyaan 2 : Apa perbedaan antara [0] dengan [8:29] ?

Jawaban 1 : Angka 0 adalah indeks. Dalam pemrograman, hitungan sering dimulai dari 0. Jadi, [0] artinya mengambil karakter pada posisi pertama dari string Nama (yaitu huruf 'B').

Jawaban 2 : [0] (Indexing): Ini digunakan untuk mengambil satu karakter saja pada posisi indeks yang spesifik (indeks ke-0). Dan [8:29] (Slicing): Ini digunakan untuk mengambil potongan string (substring), mulai dari indeks ke-8 hingga sebelum indeks ke-29.

Input program 3 :

```
print("1\t2\t3\t4\t5\t6\t7\t8\t9")
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\KUM/ferdy_060_program-variabel-string.py"
1      2      3      4      5      6      7      8      9
```

4. Program variable Boolean

Input program 1 :

```
i = 20 < 25  
print(i)  
  
n = 20 > 25  
print(n)
```

Output program 1 :

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA  
1/Programs/Python/Python310/python.exe "d:/  
KUM/ferdy_060_program-variabel-boolean.py"  
True  
False
```

Input program 2 :

```
print("Z" == "z")
```

Output program 2 :

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA  
1/Programs/Python/Python310/python.exe "d:/  
KUM/ferdy_060_program-variabel-boolean.py"  
False
```

Pertanyaan : Apa fungsi == di atas?

Jawaban : Fungsi == adalah operator perbandingan kesetaraan. Operator ini digunakan untuk membandingkan apakah nilai di sebelah kiri sama dengan nilai di sebelah kanan. Pada contoh ini, hasilnya False karena Python membedakan huruf besar dan huruf kecil (case-sensitive).

Input program 3 :

```
nama = "Gema" == "gema"
print(nama)

namamu = "Aldi" != "aldi"
print(namamu)
```

Output program 3 :

```
1/Programs/Python/Python310/python.exe "d:/TUGAS
KUM/ferdy_060_program-variabel-boolean.py"
False
True
```

Pertanyaan : Apa perbedaan antara == dengan != ?

Jawaban :

- = (Sama Dengan): Digunakan untuk memeriksa apakah dua nilai sama. Akan menghasilkan True jika sama, dan False jika berbeda.
- != (Tidak Sama Dengan): Digunakan untuk memeriksa apakah dua nilai tidak sama. Akan menghasilkan True jika berbeda, dan False jika sama.

Input program 4 :

```
print(bool("Hello"))
print(bool(""))
print(bool(15))
print(bool(0))
print(bool(["pisang", "semangka", "pir"]))
print(bool([]))
```

Output program 4 :

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA S
1/Programs/Python/Python310/python.exe "d:/T
KUM/ferdy_060_program-variabel-boolean.py"
True
False
True
False
True
False
```

Pertanyaan :

1. Mengapa output berbeda?
2. Apa kegunaan bool ?

Jawaban :

1. "Truthy": Nilai-nilai yang dianggap "berisi" atau "tidak kosong" akan menjadi True. Contoh: "Hello", angka 15, dan list yang ada isinya ["pisang", ...]. dan "Falsy": Nilai-nilai yang dianggap "kosong" akan menjadi False. Contoh: string kosong "", angka 0, dan list kosong [].
2. Kegunaan bool() adalah untuk mengonversi (mengubah) sebuah nilai (seperti string, angka, atau list) menjadi nilai Boolean (yaitu True atau False) berdasarkan aturan "truthy/falsy" tersebut.

5. Program variable float

Input 1 (menggunakan .)

```
a = float(12.69)
print(a)
```

Output 1 (menggunakan .)

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMU
1/Programs/Python/Python310/python.exe "d
KUM/ferdy_060_program-variabel-float.py"
● 12.69
```

Input 2 (menggunakan ,)

```
a = float (12.69)
print (a)
```

Output 2 (menggunakan ,)

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGA
1/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH
KUM/ferdy_060_program-variabel-float.py"
Traceback (most recent call last):
  File "d:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1
loat.py", line 1, in <module>
    a = float(12,69)
TypeError: float expected at most 1 argument, got 2
```

6. Operator aritmatika

Input :

```
a = int(input("Masukkan nilai a: "))
b = int(input("Masukkan nilai b: "))
c = a + b
d = a - b
e = a * b
f = a / b
i = b % a
h = a ** b
print(c, d, e, f, i, h)
```

Output

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEME
1/Programs/Python/Python310/python.exe "d:/TUGA
KUM/ferdy_060_program-aritmatika.py"
Masukkan nilai a: 1
Masukkan nilai b: 2
3 -1 2 0.5 0 1
```

7. Operator logika

Input :

```
a = True
b = False

c = a and b
print("%r and %r = %r" % (a,b,c))

c = a or b
print("%r or %r = %r" % (a,b,c))

c = not a
print("not %r = %r" % (a,c))
```

Output :

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SE
1/Programs/Python/Python310/python.exe
KUM/ferdy_060_operator_logika.py"
True and False = False
True or False = True
not True = False
```

8. Operator perbandingan

Input :

```
a = int(input("Inputkan nilai a: "))
b = int(input("Inputkan nilai b: "))
c = 0

c = a == b
print ("Apakah %d == %d: %r" % (a,b,c))

c = a < b
print ("Apakah %d < %d: %r" % (a,b,c))

c = a > b
print ("Apakah %d > %d: %r" % (a,b,c))

c = a <= b
print ("Apakah %d <= %d: %r" % (a,b,c))

c = a >= b
print ("Apakah %d >= %d: %r" % (a,b,c))

c = a != b
print ("Apakah %d != %d: %r" % (a,b,c))
```

Output :

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA
1/Programs/Python/Python310/python.exe "d:
KUM/ferdy_060_operator_pembandingan.py"
Inputkan nilai a: 1
Inputkan nilai b: 3
Apakah 1 == 3: False
Apakah 1 < 3: True
Apakah 1 > 3: False
Apakah 1 <= 3: True
Apakah 1 >= 3: False
Apakah 1 != 3: True
```

9. Operator penugasan

Input :

```
a = int(input("Inputkan nilai a: "))
print ("Nilai a = %d" % a)
a += 5
print ("Nilai setelah ditambah 5: ")
print ("a = %d" % a)
```

Output :

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA
1/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA
KUM/ferdy_060_operator_pemograman.py"
Inputkan nilai a: 10
Nilai a = 10
Nilai setelah ditambah 5:
a = 15
```

SOAL :

Buatlah program python untuk menghitung penyelesaian soal-soal di bawah ini.

1. Afa memiliki 260 kelereng lalu dia bermain bersama Eka lalu kalah sebanyak 180 kelereng. Kemudian dia membeli lagi kelereng sebanyak 580. Jadi jumlah kelereng milik Afa sekarang adalah...

```
afa = 260
eka = 180
beli = 580
total = afa - eka + beli
k = "Jadi kelereng yang dimiliki afa saat ini adalah "
print(afa, " - ", eka, " + ", beli, " = ", total)
print(k, total)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_operator_penugasan.py"
260 - 180 + 580 = 660
Jadi kelereng yang dimiliki afa saat ini adalah 660
```

2. Luthfan pergi ke toko alat tulis untuk membeli penghapus. Harga 1 buah penghapus Rp 1.800. Jika Luthfan membeli 1 lusin penghapus dan Ia membayar 3 lembar uang sepuluh ribuan. Berapa uang kembalian yang Luthfan terima?

```
penghapus = 1800
lusin = 12
luthfan = lusin * penghapus
bayar = 3 * 10000
kembalian = bayar - luthfan
print("Jadi kembalian yang diterima oleh Luthfan adalah ",kembalian)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS A
IAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFOR
EMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_operator_penugasan.py"
Jadi kembalian yang diterima oleh Luthfan adalah 8400
```

3. Mali berumur 19 Tahun, dia ingin mendaftar vaksin.

```
mali = 19
if mali > 17:
    print("Mali dapat mendaftar Vaksin dosis
Dewasa")
else:
    print("Mali tidak dapat mendaftar Vaksin
dosis Dewasa")
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS A
IAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFOR
EMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_operator_penugasan.py"
Mali dapat mendaftar Vaksin dosis Dewasa
```

REFERENSI

- <https://www.petanikode.com/python-operator/>
- <https://belajarpython.com/tutorial/tipe-data-python>
- <https://belajarpython.com/>



**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 - 2026**

Lembar Asistensi

Praktikum Algoritma dan Pemrograman

Pokok Bahasan 3

Judul : Struktur Pemrograman : Berkondisi
Nama : Mochammad Ferdyansyah
NIM : 251080200060
Kelompok : 8
Dilaksanakan : 13 November 2025

Mengetahui,

Dosen Praktikum

(Uce Indahyanti, S.Kom. M.Kom.)

Asisten Praktikum

(Muhamad Maulana Musafir)

POKOK BAHASAN 3

STRUKTUR PEMROGRAMAN : BERKONDISI

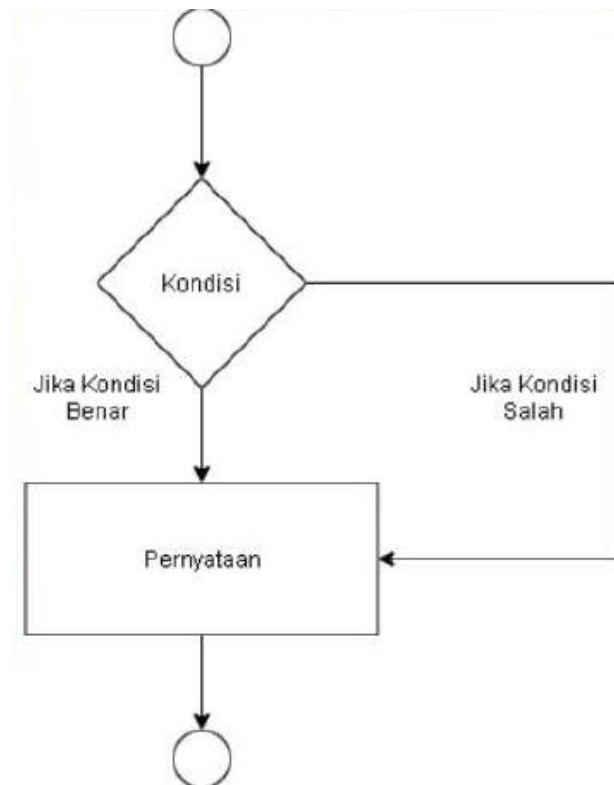
1. Pernyataan If:

Pernyataan berkondisi digunakan untuk membuat alur program agar menjadi lebih efektif dan relevan terhadap permasalahan yang ada. Dalam Bahasa Python pernyataan berkondisi *if* yang digunakan untuk memberikan beberapa alternatif pilihan seperti bilangan golongan dan pangkat jabatan, jenis kelamin perempuan atau laki-laki, bilangan genap atau ganjil, dan lain sebagainya.

Struktur kondisi if dapat digunakan dalam bentuk sebagai berikut :

If (**kondisi**) :

pernyataan1



2. Pernyataan If...Else

Struktur kondisi *if...else* dapat digunakan dalam bentuk sebagai berikut :

```
if (kondisi) :  
    pernyataan1  
    pernyataan2  
    .....  
    pernyataan  
else :  
    pernyataan
```

Pernyataan pada blok *if* akan dijalankan jika **kondisi** bernilai **BENAR**, tetapi jika bernilai **SALAH** maka pernyataan pada blok *else* akan dijalankan.

3. Pernyataan If...Elif...Else

Pernyataan *if...elif...else* digunakan untuk menguji lebih dari dua kondisi. Apabila kondisi pada *if* benar, maka pernyataan didalamnya yang dieksekusi. Apabila salah, maka masuk ke pengujian kondisi *elif*. Terakhir apabila tidak ada *if* atau *elif* yang benar, maka yang dijalankan adalah yang di blok *else*. Struktur kondisi *if...elif...else* dapat digunakan dalam beberapa bentuk sebagai berikut:

```
if (kondisi) :  
    pernyataan1  
  
elif (kondisi2) :  
    pernyataan2  
  
else :  
    pernyataanX
```

LEMBAR KERJA DAN TUGAS

1. Pernyataan Berkondisi

```
umur = int(input("Masukkan Umurmu : "))
if(umur >= 60):
    print("Selamat Siang, Mbah")
    print("Salam buat cucumu ya")
    print("Jadi Umurmu", umur, "tahun")
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PR
IAL\AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULI
EMESTER 1\TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"
Masukkan Umurmu : 65
Selamat Siang, Mbah
Salam buat cucumu ya
Jadi Umurmu 65 tahun
```

Jalankan dan isikan dengan umur 25 tahun, apa yang terjadi?

Output komentar tidak muncul.

Jelaskan mengapa komentar tidak muncul?

Karena jika memasukkan nilai 25 maka akan menghasilkan kondisi False yang tidak ada pernyataannya.

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALG
IAL\AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULI
EMESTER 1\TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"
Masukkan Umurmu : 25
```

Coba ganti program di atas menjadi program di bawah ini!

```
tahun = int(input("Masukkan tahun kelahiranmu : "))
umur = 2013 - tahun

print("Umurmu", umur, "tahun")

if(umur < 17):
    print("Kamu belum sweet seventeen")
    print("Sorry kamu belum cukup umur")
```

Jalankan dan isikan tahun kelahiran 1988 dan 1975 serta nilai lain.

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALG  
IAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA  
SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"  
Masukkan tahun kelahiranmu : 1988  
Umurmu 25 tahun  
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALG  
IAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA  
SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"  
Masukkan tahun kelahiranmu : 1975  
Umurmu 38 tahun
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALG  
IAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA  
SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"  
Masukkan tahun kelahiranmu : 2003  
Umurmu 10 tahun  
Kamu belum sweet seventeen  
Sorry kamu belum cukup umur
```

Kesalahan apa yang terjadi dan bagaimana yang benar?

Kesalahan yang terjadi adalah umurnya tidak akurat untuk waktu sekarang ini, karena tahun yang dikurang adalah 2013. Seharusnya umur = 2013-tahun diganti menjadi umur = 2021-tahun.

2. Struktur Pernyataan Berkondisi

Buatlah program untuk menghitung total bayar tiket bioskop tiap – tiap pengunjung.

```
bioskop = input("Nama Bioskop : ")  
tiket = int(input("Harga Tiket : "))  
pengunjung = int (input("Masukkan Pengunjung :  
"))  
if(tiket == 50000) & (pengunjung == 1):  
    print("Tiket VIP")  
else:  
    print("Tiket Regular")  
total_bayar = pengunjung * tiket  
print(("Total yang harus dibayar adalah",  
total_bayar))
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALG  
IAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA  
SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"  
Nama Bioskop : CINEMA  
Harga Tiket : 50000  
Masukkan Pengunjung : 2  
Tiket Regular  
(Total yang harus dibayar adalah', 100000)
```

Coba ganti program di atas menjadi program di bawah ini :

```
bioskop = input("Nama Bioskop : ")
tiket = int(input("Harga Tiket : "))
pengunjung = int (input("Masukkan Pengunjung : "))

if(tiket == 50000) & (pengunjung == 1):
    print("Tiket VIP")
else:
    print("Tiket Regular")

total_bayar = pengunjung * tiket
print(("Total yang harus dibayar adalah",
total_bayar))
```

Jalankan dan jelaskan apa yang terjadi? Mengapa hasil tidak seperti sebelumnya?

Hasilnya menjadi berbeda meskipun harga tiket sudah 50000, Karena terdapat kondisi yang mengharuskan pengunjung == 1 atau pengunjung sama dengan 1 agar hasilnya menjadi True (Tiket VIP)

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\
AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"
Nama Bioskop : XXI
Harga Tiket : 50000
Masukkan Pengunjung : 2
Tiket Regular
('Total yang harus dibayar adalah', 100000)
```

3. Perintah Pernyataan Berkondisi

Buatlah program untuk menentukan grade dan apakah mahasiswa dinyatakan lulus atau tidak menggunakan nilai.

```
nama = input("Nama : ")
kelas = input("Kelas : ")
matkul = input("Mata Kuliah : ")
nilai = int(input("Berapa Nilai Mata Kuliah : "))

if(nilai >= 80):
    grade = "A"
elif(nilai >= 60):
    grade = "B"
elif(nilai >= 40):
    grade = "C"
elif(nilai < 40):
    grade = "D"
print("Memperoleh Grade : ", grade)
print('Nilai Mata Kuliah', matkul, 'atas nama',
nama, 'dinyatakan')
if(nilai >=60):
    print("LULUS")
elif(nilai >= 40):
    print("REMIDI")
elif(nilai > 40):
    print("TIDAK LULUS")
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\
AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA\SEMESTER 1\TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi.py"
Nama : Mochammad Ferdyansyah
Kelas : A2
Mata Kuliah : ALGORITMA
Berapa Nilai Mata Kuliah : 90
Memperoleh Grade : A
Nilai Mata Kuliah ALGORITMA atas nama Mochammad Ferdyansyah dinyatakan
LULUS
```

Dari program di atas, apa yang terjadi jika menentukan lulus atau tidaknya menggunakan grade?

Jika menggunakan grade, hal yang menentukan lulus atau tidaknya adalah dari nilai grade A, B, C, D. Jika memperoleh grade A/B maka dinyatakan LULUS, jika memperoleh C maka dinyatakan REMIDI, jika memperoleh grade D maka dinyatakan TIDAK LULUS

4. Perintah Pernyataan Berkondisi Bersarang

```
nama = input("Nama Mahasiswa : ")
nim = input("NIM Mahasiswa : ")
prodi = input("Prodi : ")
kelas = input("Kelas : ")
if(prodi == "Informatika"):
    semester = 1
    if(kelas == "A"):
        matkul = "Algoritma Pemrograman"
    elif(kelas == "B"):
        matkul = "Algoritma Struktur Data"
    elif(kelas == "C"):
        matkul = "Sistem Digital"
elif(prodi != "Informatika"):
    semester = 1
    if(kelas == "A"):
        matkul = "Pendidikan Agama Islam"
    elif(kelas == "B"):
        matkul = "Pancasila"
    elif(kelas == "C"):
        matkul = "Seni Budaya"
print("semester:", "semester")
print("matkul:", "matkul")
```

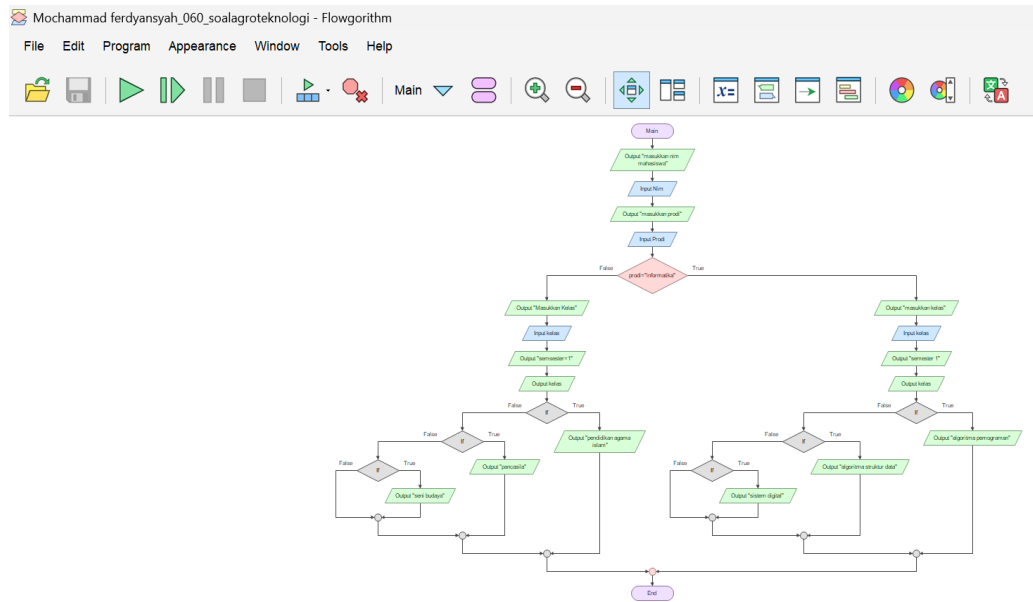
```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM\
IAL\AppData\Local\Programs\Python\Python310\python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi_bersarang.py"
Nama Mahasiswa : Mochammad Ferdiansyah
NIM Mahasiswa : 251080200060
Prodi : INFORMATIKA
Kelas : A2
semester: semester
matkul: matkul
```

Coba jalankan program di atas dengan menggunakan nama prodi **AGROTEKNOLOGI**. Jelaskan apa yang terjadi? Buatlah flowchart dari program di atas!

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM\
IAL\AppData\Local\Programs\Python\Python310\python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM/ferdy_060_pernyataan_berkondisi_bersarang.py"
Nama Mahasiswa : Mochammad Ferdiansyah
NIM Mahasiswa : 251080200060
Prodi : AGROTEKNOLOGI
Kelas : A
semester: semester
matkul: matkul
```

Apa yang terjadi adalah menghasilkan output Matkul yang berbeda-beda tergantung dari kelasnya, jika Kelas = A maka Matkul = Pendidikan Agama Islam, kemudian Kelas B = Pancasila, dan Kelas C = Seni Budaya.

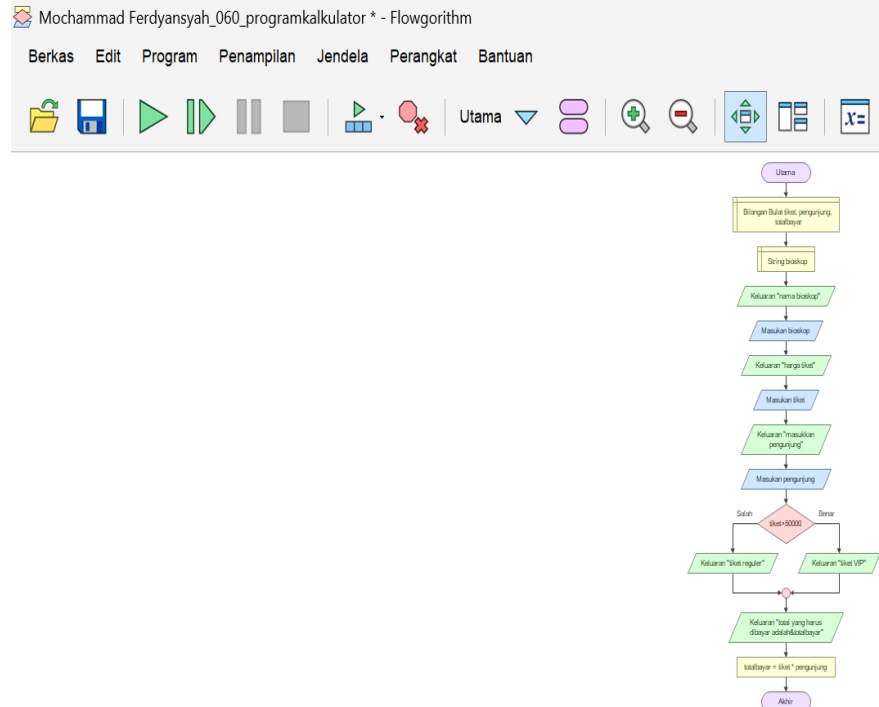
Buatlah Flowchart dari program di atas!



TUGAS

Menu : Membuat Program Kalkulator

1. Buatkan Flowchart dari program pada no. 2



2. Buatlah program kalkulator menggunakan operator aritmatika dalam memproses data dan menggunakan percabangan untuk memilih operatornya. Sehingga terdapat menu :

Daftar Operator :

1. Pertambahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Sisa bagi
6. Kuadrat

Source :

```
print("Daftar Operator")
print(" 1 = Pertambahan ")
print(" 2 = Pengurangan ")
print(" 3 = Perkalian ")
print(" 4 = Pembagian ")
print(" 5 = Sisa bagi ")
print(" 6 = Kuadrat")

pilihan = int(input("Pilih nomor operasi yang akan
digunakan = "))
if(0 < pilihan < 7):
    if(pilihan == 1):
        print("Anda memilih pertambahan ")
        bill1 = float (input("Nilai A = "))
        bil2 = float (input("Nilai B = "))
        hasil = (bill1 + bil2)
        print("Hasil pertambahan = %.2f" % hasil)
    elif(pilihan == 2):
        print("Anda memilih pegurangan ")
        bill1 = float (input("Nilai A = "))
        bil2 = float (input("Nilai B = "))
        hasil = (bill1 - bil2)
        print("Hasil pengurangan = %.2f" % hasil)
    elif (pilihan == 3):
        print("Anda memilih perkalian ")
        bill1 = float(input("Nilai A = "))
        bil2 = float(input("Nilai B = "))
        hasil = (bill1 * bil2)
        print("Hasil perkalian = %.2f" % hasil)
    elif (pilihan == 4):
        print("Anda memilih pembagian ")
        bill1 = float(input("Nilai A = "))
        bil2 = float(input("Nilai B = "))
        hasil = (bill1 / bil2)
        print("Hasil pembagian = %.2f" % hasil)
    elif (pilihan == 5):
        print("Anda memilih sisa bagi ")
        bill1 = float(input("Nilai A = "))
```

```

        bil2 = float(input("Nilai B = "))
        hasil = (bil1 % bil2)
        print("Hasil sisa bagi = %.2f" % hasil)
    elif (pilihan == 6):
        print("Anda memilih kuadrat ")
        bil1 = float(input("Nilai A = "))
        bil2 = float(input("Nilai B = "))
        hasil = (bil1 ** bil2)
        print("Hasil kuadrat = %.2f" % hasil)
    else :
        print("Masukkan nilai operator yang benar")

```

```

PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEME:
/Programs/Python/Python310/python.exe "d:/TUGAS
M/ferdy_060_programkalulatorsederhana.py"

```

Daftar Operator

- 1 = Pertambahan
- 2 = Pengurangan
- 3 = Perkalian
- 4 = Pembagian
- 5 = Sisa bagi
- 6 = Kuadrat

Pilih nomor operasi yang akan digunakan = 1

Anda memilih pertambahan

Nilai A = 10

Nilai B = 10

Hasil pertambahan = 20.00

```

PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SE
/Programs/Python/Python310/python.exe "d:/TUG
M/ferdy_060_programkalulatorsederhana.py"

```

Daftar Operator

- 1 = Pertambahan
- 2 = Pengurangan
- 3 = Perkalian
- 4 = Pembagian
- 5 = Sisa bagi
- 6 = Kuadrat

Pilih nomor operasi yang akan digunakan = 2

Anda memilih pengurangan

Nilai A = 10

Nilai B = 10

Hasil pengurangan = 0.00

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMES  
/Programs/Python/Python310/python.exe "d:/TUGAS  
M/ferdy_060_programkalulatorsederhana.py"  
Daftar Operator  
1 = Pertambahan  
2 = Pengurangan  
3 = Perkalian  
4 = Pembagian  
5 = Sisa bagi  
6 = Kuadrat  
Pilih nomor operasi yang akan digunakan = 3  
Anda memilih perkalian  
Nilai A = 10  
Nilai B = 10  
Hasil perkalian = 100.00
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMES  
/Programs/Python/Python310/python.exe "d:/TUGAS  
M/ferdy_060_programkalulatorsederhana.py"  
Daftar Operator  
1 = Pertambahan  
2 = Pengurangan  
3 = Perkalian  
4 = Pembagian  
5 = Sisa bagi  
6 = Kuadrat  
Pilih nomor operasi yang akan digunakan = 4  
Anda memilih pembagian  
Nilai A = 10  
Nilai B = 10  
Hasil pembagian = 1.00
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMES  
/Programs/Python/Python310/python.exe "d:/TUGAS  
M/ferdy_060_programkalulatorsederhana.py"  
Daftar Operator  
1 = Pertambahan  
2 = Pengurangan  
3 = Perkalian  
4 = Pembagian  
5 = Sisa bagi  
6 = Kuadrat  
Pilih nomor operasi yang akan digunakan = 5  
Anda memilih sisa bagi  
Nilai A = 10  
Nilai B = 10  
Hasil sisa bagi = 0.00
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEI
/Programs/Python/Python310/python.exe "d:/TUGA
M/ferdy_060_programkalulatorsederhana.py"
Daftar Operator
 1 = Pertambahan
 2 = Pengurangan
 3 = Perkalian
 4 = Pembagian
 5 = Sisa bagi
 6 = Kuadrat
Pilih nomor operasi yang akan digunakan = 6
Anda memilih kuadrat
Nilai A = 10
Nilai B = 10
Hasil kuadrat = 10000000000.00
```

REFERENSI

1. <https://www.petanikode.com/python-operator/>
2. <https://belajarpython.com/>
3. Muhammad Khudzaifah, Hisyam Fahmi, dan Mohammad Jamhuri. 2018.
Buku Pendamping Praktikum Pemograman Komputer 1. March 8, 2018
4. Universitas Pancasila. Modul Praktikum Dasar Pemograman Dengan
Bahasa Python.
5. Rizqi Maulana. 2020. Python Untuk Pemula February 13, 2020.



**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 - 2026**

Lembar Asistensi

Praktikum Algoritma dan Pemrograman

Pokok Bahasan 4

Judul : Struktur Pemograman : Perulangan
Nama : Mochammad Ferdyansyah
NIM : 251080200060
Kelompok : 8
Dilaksanakan : 09 Januari 2026

Mengetahui,

Dosen Praktikum

Asisten Praktikum

(Uce Indahyanti, S.Kom. M.Kom.)

(Muhamad Maulana Musafir)

POKOK BAHASAN 4

STRUKTUR PEMROGRAMAN : PERULANGAN

4.1 Perulangan For:

Perulangan For adalah suatu perintah yang di ulang, dimana perintah tersebut dikendalikan oleh sebuah variable, sehingga dapat memudahkan kita untuk memproses setiap elemen pada list dengan variable sementara dan mengaplikasikan kode yang sama. Dalam perulangan for di pyhton sedikit berbeda dengan bahasa lainnya. Perulangan ini tentunya dirancang untuk menelusuri nilai ataupun elemen yang terdapat pada String, list, tuple dictionary, dan objek yang mempunyai tipe range.

Struktur perulangan for dapat digunakan dalam bentuk sebagai berikut :

for elemen in tipe_koleksi:

perintah1

perintah2

Dilihat dari bentuk diatas bahwa perintah1 dan perintah2 akan dilakukan perulangan lewat elemen yang terdapat didalam tipe koleksi. Jadi, ada 3 bagian penting. Sequence adalah sebuah nilai yang bersifat iterable alias bisa diulang-ulang. Di antara tipe data yang bersifat sequence atau iterable adalah:

- List
- Tuple
- String

Dan lain sebagainya. Nilai adalah setiap item yang diekstrak dari sequence. Blok kode yaitu statemen-statemen atau perintah-perintah tertentu yang akan dieksekusi secara berulang.

4.2 Perulangan While

Struktur perulangan (atau dalam bahasa Inggris disebut dengan loop) adalah instruksi kode program yang bertujuan untuk mengulang beberapa baris perintah.

Dalam merancang perulangan, kita setidaknya harus mengetahui 3 komponen:

1. Kondisi awal perulangan.
2. Kondisi pada saat perulangan.
3. Kondisi yang harus dipenuhi agar perulangan berhenti.

while (kondisi) :

perintah

Kondisi yang dimaksud merupakan suatu kondisi atau keadaan yang kita buat. Selama nilai yang diulang masih sesuai dengan kondisi yang sudah ditetapkan yang berarti True dan tidak bernilai 0 maka while akan terus berjalan.

Lembar Kerja dan Tugas

1. Perulangan For

```
for kata in 'Halo':  
    print (kata)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2> python.exe "d:/TUGAS ALGORITMA/PRAKTIKUM 2/modul4_1_ferdy060.py"  
H  
a  
l  
o  
Open folder in new window (ctrl + click)
```

Gantikan kata dengan nama dan masukkan string dengan nama panggilan dan nim terakhir anda, lalu jalankan.

```
for nama in 'Ferdy 060':  
    print(nama)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2> python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER/SA  
TIKUM 2/perulanganfor_ferdy_060.py"  
F  
e  
r  
d  
y  
  
0  
6  
0  
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2>
```

2. For dengan list

```
buah = ['mangga' , 'melon' , 'jeruk']  
for buahku in buah :  
    print('Saya suka sekali dengan' + buah)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2> python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER/SA  
TIKUM 2/perulanganforlist_ferdy_060.py"  
Saya suka sekali dengan mangga  
Saya suka sekali dengan melon  
Saya suka sekali dengan jeruk  
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2>
```

Selain tipe string dan list, dalam penggunaan for dapat diterapkan pada dictionary, dictionary ini digunakan untuk mengelola kelompok data. Perbedaannya adalah dictionary menggunakan kunci(key). Pada dictionary, sebuah kunci nantinya dipasangkan dengan sebuah nilai(value) untuk nantinya dapat membentuk suatu elemen, yang dikenal sebagai kunci-nilai (key-value).

Dictionary dapat dibuat sebagai berikut :

```
{kunci1 : nilai1, kunci2 : nilai2, kunci3 :  
nilai3}
```

Contoh dari dictionary bisa diperhatikan pada kode di bawah:

```
buah = {'mangga' : 'kuning', 'melon' : 'hijau',  
'jeruk' : 'orange', 'anggur' : 'ungu'}  
  
for buahku in buah :  
  
    print(buahku + ' warnanya adalah ' +  
        buah[buahku])
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2>  
cal/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER/S  
TIKUM 2/lanjutanforlistwarna_ferdy060.py"  
mangga warnanya adalah kuning  
melon warnanya adalah hijau  
jeruk warnanya adalah orange  
anggur warnanya adalah ungu  
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\PRAKTIKUM 2>
```

Setelah mencoba elemen list dari kedua program tersebut, silahkan jelaskan perbedaannya!

Perbedaannya adalah Jika menggunakan List maka index – nya adalah berupa numerik. Jika menggunakan Dictionary maka index – nya adalah berupa string yang dapat ditetapkan.

3. For dengan range

```
for i in range (10) :  
    print('perulangan ke - ', i);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEM  
cal/Programs/Python/Python310/python.exe  
TIKUM 2/fordenganrange_ferdy060.py"  
--- Range 0 sampai 9 ---  
Perulangan ke 0  
Perulangan ke 1  
Perulangan ke 2  
Perulangan ke 3  
Perulangan ke 4  
Perulangan ke 5  
Perulangan ke 6  
Perulangan ke 7  
Perulangan ke 8  
Perulangan ke 9
```

Dengan fungsi range, kita bisa melakukan perulangan dari 0, sampai kurang dari nilai range yang kita definisikan. Kita juga bisa memulai range selain dari 0.

```
for i in range (10, 20) :  
    print('Nilai i = ', i);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMU  
cal/Programs/Python/Python310/python.exe  
TIKUM 2/fordenganrange_ferdy060.py"  
--- Range 10 sampai 19 ---  
Nilai i = 10  
Nilai i = 11  
Nilai i = 12  
Nilai i = 13  
Nilai i = 14  
Nilai i = 15  
Nilai i = 16  
Nilai i = 17  
Nilai i = 18  
Nilai i = 19
```

Selain itu juga bisa mendefinisikan kelipatannya

```
for i in range (2, 30, 2) :  
  
    print('Nilai i = ', i);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY S  
cal/Programs/Python/Python310/python.e  
TIKUM 2/fordenganrange_ferdy060.py"  
--- Range Kelipatan 2 (Genap) ---  
Nilai i = 2  
Nilai i = 4  
Nilai i = 6  
Nilai i = 8  
Nilai i = 10  
Nilai i = 12  
Nilai i = 14  
Nilai i = 16  
Nilai i = 18  
Nilai i = 20  
Nilai i = 22  
Nilai i = 24  
Nilai i = 26  
Nilai i = 28
```

Sistem akan melakukan perulangan dimulai dari angka 2, hingga kurang dari 30 dengan interval/kelipatan sebanyak 2. Dan juga bisa dengan bilangan ganjil

```
for i in range (1, 30, 2) :  
  
    print('Nilai i = ', i);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMESTER  
cal/Programs/Python/Python310/python.exe "d:/TUGAS I  
TIKUM 2/fordenganrange_ferdy060.py"  
--- Range Ganjil ---  
Nilai i = 1  
Nilai i = 3  
Nilai i = 5  
Nilai i = 7  
Nilai i = 9  
Nilai i = 11  
Nilai i = 13  
Nilai i = 15  
Nilai i = 17  
Nilai i = 19  
Nilai i = 21  
Nilai i = 23  
Nilai i = 25  
Nilai i = 27  
Nilai i = 29
```

4. For dengan tuple

```
tupleBinatang = ('Kucing', 'Anjing', 'Burung', 'Ayam')

for Binatang in tupleBinatang :

    print(Binatang);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA/PRAKTIKUM 2/fortuplebinatang_ferdy060.py"
Kucing
Anjing
Burung
Ayam
```

Tuple bersifat iterable sehingga bisa dilakukan sebagai objek perulangan. Kalian coba buat dengan nama-nama kota yang ada di Jawa Timur.

```
tuplekota = ('Sidoarjo', 'Surabaya', 'Gresik', 'Ponorogo')

for Kota in tuplekota :

    print(Kota)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA/PRAKTIKUM 2/forstringmodul_ferdy060.py"
Sidoarjo
Surabaya
Malang
Gresik
Mojokerto
```

5. For dengan string

```
karakter = 'Indonesia';

for kata in karakter :

    print(kata);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA/PRAKTIKUM 2/forstringmodul_ferdy060.py"
I
n
d
o
n
e
s
i
a
```

String juga bersifat iterable sehingga dapat dijadikan objek perulangan. Kalian coba buat dengan string nama kalian masing-masing.

```
karakter = 'Ferdy';  
  
for nama in karakter :  
  
    print(nama);
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY S  
cal/Programs/Python/Python310/python.  
TIKUM 2/fordenganstring_ferdy060.py"  
F  
e  
r  
d  
y
```

6. For dengan break dan continue

```
for i in range (10) :  
    if (i == 6) :  
        continue  
    print(i)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\  
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUG  
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/forbreak_ferdy060.py"  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9
```

Jika menggunakan continue pada sebuah perulangan maka berfungsi untuk menskip ke iterasi selanjutnya. Pada kode diatas jika nilai i sama dengan 6 akan dilanjutkan ke iterasi berikutnya.

```
for i in range (10) :  
    if (i == 5) :  
        break  
    print(i)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\  
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUG  
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/forinrange_ferdy060.py"  
0  
1  
2  
3  
4
```

Jika menggunakan break maka perulangan akan dihentikan meskipun belum sampai literasi terakhir.

7. For dengan else

```
namakota = ['Sidoarjo', 'Surabaya', 'Mojokerto']
for kota in namakota :
    print(kota)
else :
    print('Selesai')
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS PRAKTIKUM 2\perulanganfor_ferdy060.py>
cal/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS PRAKTIKUM 2/fordgnelse_ferdy060.py"
Sidoarjo
Surabaya
Mojokerto
Selesai
```

dengan menggunakan else, perintah pada blok else hanya dieksekusi ketika perulangan selesai secara natural tanpa interupsi

8. Perulangan While

```
i = 1
while i <= 10 :
    print('Hello World')
    i += 1
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS PRAKTIKUM 2\perulanganwhile_ferdy060.py>
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS PRAKTIKUM 2/contohwhile_ferdy060.py"
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
```

Di dalam blok perulangan terdapat perintah print("Hello World") di baris 3, ubah Hello World menjadi Nama masing-masing dan 3 digit NIM akhir.

```
i = 1
while i <= 10 :
    print('Ferdy 060')
    i += 1
```

```
[Running] python -u "d:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TUGAS PRAKTIKUM 2\perulanganwhile_ferdy060.py"
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
Ferdy 060
```

9. While untuk list

```
buah = ['Apel', 'Durian', 'Jeruk', 'Melon',  
        'Semangka']  
  
i = 0  
while i < len(buah):  
    print(buah[i])  
    i += 1
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1  
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/T  
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/whilelist_ferdy060.py"  
Apel  
Durian  
Jeruk  
Melon  
Semangka
```

untuk menampilkan semua list cara yang mudah dapat menggunakan metode for, tetapi kita juga bisa menggunakan perulangan while. Kalian cobat buat dengan nama-nama teman kelas kalian 5 orang.

```
teman = ['Faqih', 'Sukma', 'Kurniawan', 'Rama',  
        'Riyadh']  
  
i = 0  
while i < len(buah):  
    print(buah[i])  
    i += 1
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY  
cal/Programs/Python/Python310/python  
TIKUM 2/whileuntuklist_ferdy060.py"  
Andi  
Budi  
Citra  
Dedi  
Eka
```

10. While dengan break dan continue

```
i = 6
while i >= 6 :
    print(i)
    if i == 15 :
        break
    i += 1
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\cal/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TIKUM 2/nomor10_ferdy060.py"
6
7
8
9
10
11
12
13
14
15
```

Coba jalankan perintah diatas, jika sudah benar lalu ubah i pada print menjadi string Informatika dan cobalah untuk mengganti pernyataan break menjadi pernyataan continue, kemudian jelaskan perbedaan dari kedua pernyataan tersebut.

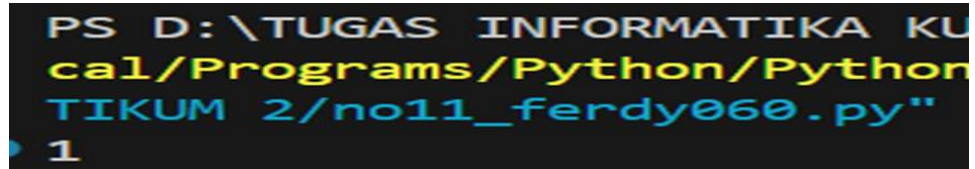
```
i = 6
while i >= 6 :
    print('Informatika')
    if i == 15 :
        continue
    i += 1
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\cal/Programs/Python/Python310/python.exe "d:/TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER 1\TIKUM 2/whiledgnbreakdancontinue_ferdy060.py"
Informatika
Informatika
Informatika
Informatika
Informatika
Informatika
Informatika
Informatika
Informatika
Informatika
```

Penjelasan : Perbedaan statement continue dan break. Ketika program membaca statement continue dalam pengulangan/looping, maka proses looping akan tetap dijalankan tanpa membaca/melompati statement yang berada dibawahnya (jika ada). Dan jika program membaca statement break pada perulangan, maka proses pengulangan akan secara otomatis dihentikan dan keluar dari proses looping serta jika terdapat statement yang berada dibawah break juga tidak dijalankan. Pada dasarnya statement ini memiliki persamaan akan melewati atau melompati statement yang berada dibawahnya.

11. While .. else

```
x = 1;
while x < 10 :
    print(x)
    break
else:
    print('Selesai')
```



```
PS D:\TUGAS INFORMATIKA KUKUcal/Programs/Python/PythonTIKUM 2/no11_ferdy060.py"
1
```

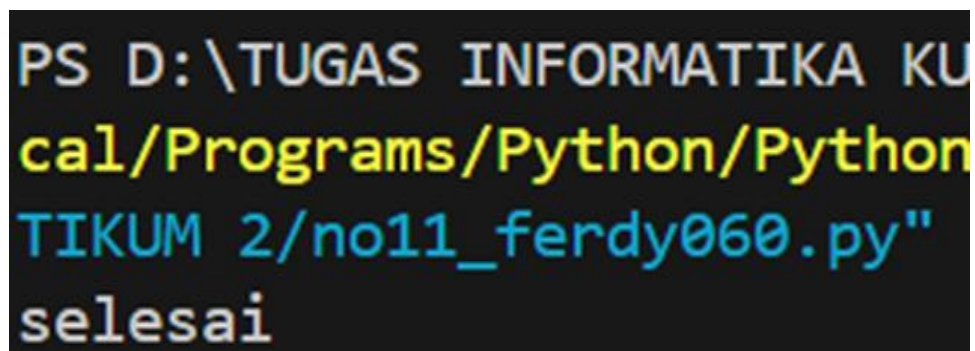
Jika

```
x = 20;

while x < 10:

    print
(x) break
    else:

    print ("selesai")
```



```
PS D:\TUGAS INFORMATIKA KUKUcal/Programs/Python/PythonTIKUM 2/no11_ferdy060.py"
selesai
```

dengan menggunakan else print selesai tidak dieksekusi karena terjadi penghentian perulangan dengan syntax break

Tugas!

Buatlah program sederhana dari elemen pada list dalam perulangan for menggunakan dictionary, yang kata kunci serta nilainya berisi nama, nim, jenis kelamin, alamat temanmu!

```
# List teman
teman = [
    {'a': 'Mochammad Ferdyansyah', 'b': '251080200060', 'c': 'Laki-laki', 'd': 'Gempol, Pasuruan'},
    {'a': 'Fajar Damanhuri', 'b': '251080200065', 'c': 'laki laki', 'd': 'Beji, Pasuruan'}
]

# Perulangan for untuk menampilkan data teman
for data in teman:
    print(f>Nama: {data['a']}")
    print(f"NIM Mahasiswa: {data['b']}")
    print(f"Jenis Kelamin: {data['c']}")
    print(f"Alamat: {data['d']}\n")
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\TUGAS ALGORITMA\
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TUGAS ALGORITMA/
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/tugasmodul4_ferdy060.py"
Nama: Mochammad Ferdyansyah
NIM Mahasiswa: 251080200060
Jenis Kelamin: Laki-laki
Alamat: Gempol, Pasuruan

Nama: Fajar Damanhuri
NIM Mahasiswa: 251080200065
Jenis Kelamin: laki laki
Alamat: Beji, Pasuruan
```

REFERENSI

- Muhammad Khudzaifah, Hisyam Fahmi, dan Mohammad Jamhuri. 2018. Buku Pendamping Praktikum Pemrograman Komputer I. March 8, 2018.
- Universitas Pancasila. Modul Praktikum Dasar Pemrograman Dengan Bahasa Python.
- Rizqi Maulana. 2020. Python Untuk Pemula. Pebruary 13, 2020.
- Jago Ngoding. 2021. Perulangan For, Perulangan While.



**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 - 2026**

Lembar Asistensi

Praktikum Algoritma dan Pemrograman

Pokok Bahasan 5

Judul : Tipe Data Koleksi
Nama : Mochammad Ferdyansyah
NIM : 251080200060
Kelompok : 8
Dilaksanakan : 08 Januari 2026

Mengetahui,

Dosen Praktikum

Asisten Praktikum

(Uce Indahyanti, S.Kom. M.Kom.)

(Muhamad Maulana Musafir)

POKOK BAHASAN 5

TIPE DATA KOLEKSI

5.1 List

Dalam bahasa pemrograman Python, struktur data yang paling dasar adalah urutan atau lists. Setiap elemen-elemen berurutan akan diberi nomor posisi atau indeks. Indeks pertama dalam list adalah nol, indeks kedua adalah satu dan seterusnya.

Python memiliki enam jenis urutan built-in, namun yang paling umum adalah list dan tuple. Ada beberapa hal yang dapat Anda lakukan dengan semua jenis list. Operasi ini meliputi pengindeksan, pengiris, penambahan, perbanyak, dan pengecekan keanggotaan. Selain itu, Python memiliki fungsi built-in untuk menemukan panjang list dan untuk menemukan elemen terbesar dan terkecilnya.

1. Membuat List Python

List adalah tipe data yang paling serbaguna yang tersedia dalam bahasa Python, yang dapat ditulis sebagai daftar nilai yang dipisahkan koma (item) antara tanda kurung siku. Hal penting tentang daftar adalah item dalam list tidak boleh sama jenisnya.

Membuat list sangat sederhana, tinggal memasukkan berbagai nilai yang dipisahkan koma di antara tanda kurung siku. Dibawah ini adalah contoh sederhana pembuatan list dalam bahasa Python.

2. Akses Nilai Dalam List Python

Untuk mengakses nilai dalam list python, gunakan tanda kurung siku untuk mengiris beserta indeks atau indeks untuk mendapatkan nilai yang tersedia pada indeks tersebut.

3. Update Nilai Dalam List Python

Anda dapat memperbarui satu atau beberapa nilai di dalam list dengan memberikan potongan di sisi kiri operator penugasan dan anda dapat menambahkan nilai ke dalam list dengan metode `append()`.

4. Hapus Nilai Dalam List Python

Untuk menghapus nilai di dalam list python, Anda dapat menggunakan salah satu pernyataan `del` jika Anda tahu persis elemen yang Anda hapus. Anda dapat menggunakan metode `remove()` jika Anda tidak tahu persis item mana yang akan dihapus.

5.2 Tuple

Sebuah tuple adalah urutan objek Python yang tidak berubah. Tuple adalah urutan, seperti daftar. Tuple bersifat immutable, artinya isi tuple tidak bisa kita ubah dan hapus. Namun, dapat kita isi dengan berbagai macam nilai dan objek. Tuple menggunakan tanda kurung, sedangkan List Python menggunakan tanda kurung siku.

Membuat tuple semudah memasukkan nilai-nilai yang dipisahkan koma. Secara opsional, dapat memasukkan nilai-nilai yang dipisahkan koma ini di antara tanda kurung juga.

1. Membuat tuple kosong dan singleton

untuk membuat Tuple yang hanya berisi satu (singleton), maka harus menambahkan tanda koma di belakangnya. Karena jika tidak ditambahkan koma, akan dianggap sebagai string.

2. Mengakses dan memotong(slicing) nilai tuple

Untuk mengakses nilai dalam tuple, gunakan tanda kurung siku untuk mengiris beserta indeks atau indeks untuk mendapatkan nilai yang tersedia pada indeks tersebut. Sama seperti list, Tuple juga memiliki indeks untuk Mengakses item di dalamnya. Indeks Tuple dan list selalu dimulai dari nol.

3. Menghapus nilai tuple

secara eksplisit menghapus keseluruhan tuple, cukup gunakan `del` statement.

5.3 Dictionary

Dictionary adalah tipe data yang anggotanya terdiri dari pasangan-pasangan kunci:nilai (key:value). Dictionary dibuat dengan menempatkan anggota di dalam tanda kurung kurawal `{ }`, dipisahkan oleh tanda koma.

Anggota dictionary terdiri dari pasangan kunci:nilai. Kunci harus bersifat unik, tidak boleh ada dua kunci yang sama dalam dictionary.

1. Mengakses elemen dictionary

Dictionary tidak menggunakan indeks. Anggota dictionary diakses dengan menggunakan kuncinya. Selain itu, bisa juga diakses dengan menggunakan fungsi `get()`.

Dengan menggunakan fungsi `get()`, bila kunci tidak ada di dalam dictionary, maka akan kembali `None`. Bila tidak menggunakan fungsi `get()`, maka akan terjadi eror `KeyError` bila kunci hendak diakses tidak ada dalam dictionary.

2. Update Nilai Dalam Dictionary Python

3. Hapus Elemen Dictionary Python

untuk menghapus elemen dalam dictionary, kita bisa menghapus anggota tertentu dengan menggunakan fungsi `pop()`. Fungsi ini menghapus anggota dengan mengembalikan kunci dari anggota tersebut. Terdapat juga fungsi lainnya yaitu `popitem()` yang digunakan untuk menghapus anggota acak dari dictionary. Untuk menghapus semua anggota dictionary, bisa menggunakan fungsi `clear()`. Selain itu kita juga bisa menggunakan kata kunci `del` untuk menghapus anggota tertentu atau menghapus dictionary itu sendiri.

Python menyertakan fungsi built-in sebagai berikut :

Fungsi Python	Penjelasan
<code>cmp(dict1, dict2)</code>	Membandingkan unsur keduanya.
Fungsi Python	Penjelasan
<code>len(dict)</code>	Memberikan panjang total Dictionary. Ini sama dengan jumlah item dalam Dictionary.
<code>str(dict)</code>	Menghasilkan representasi string yang dapat dicetak dari Dictionary

<code>type(variable)</code>	Mengembalikan tipe variabel yang lulus. Jika variabel yang dilewatkan adalah Dictionary, maka akan mengembalikan tipe Dictionary.
-----------------------------	---

Method Build-in Pada Dictionary Python

Python menyertakan method built-in sebagai berikut:

Method Python	Penjelasan
<code>dict.clear()</code>	Menghapus semua elemen Dictionary
<code>dict.copy()</code>	Mengembalikan salinan Dictionary
<code>dict.fromkeys()</code>	Buat Dictionary baru dengan kunci dari seq dan nilai yang disetel ke nilai.
<code>dict.get(key, default=None)</code>	For key, nilai pengembalian atau default jika tombol tidak ada dalam Dictionary
<code>dict.has_key(key)</code>	Mengembalikan true jika key dalam Dictionary, false sebaliknya
<code>dict.items()</code>	Mengembalikan daftar dari pasangan tuple dictionary (key, value)
<code>dict.keys()</code>	Mengembalikan daftar key dictionary
<code>dict.setdefault(key, default=None)</code>	Mirip dengan get (), tapi akan mengatur dict [key] — default jika kunci belum ada di dict
<code>dict.update(dict2)</code>	Menambahkan pasangan kunci kata kunci dict2 ke dict
<code>dict.values()</code>	Mengembalikan daftar nilai dictionary

5.4 Set

Set adalah salah satu tipe data di Python yang tidak berurut (unordered). Set memiliki anggota yang unik (tidak ada duplikasi). Jadi jika ada dua anggota yang sama di dalam set, maka otomatis set akan menghilangkan salah satunya.

Set dibuat dengan meletakkan anggota — anggotanya di dalam tanda kurung kurawal { }, dipisahkan menggunakan tanda koma. Kita juga bisa membuat set dari list dengan memasukkan list ke dalam fungsi set ().

1. Mengakses item set

Karena set tidak terindeks maka untuk mengaksesnya tidak bisa menggunakan indeks seperti tuple dan list. Untuk mengakses elemen-elemen dalam set bisa dengan menggunakan perulangan (for ...).

2. Mengecek elemen set

Cara mengecek elemen dalam set adalah menggunakan if in set: kemudian print(set).

3. Menambahkan item kedalam set

Setelah set dibuat maka tidak dapat diubah karena set bersifat unchangeable. Namun set dapat ditambah elemennya dengan menggunakan fungsi add() dan fungsi update(). Add() digunakan untuk hanya menambah 1 item kedalam set, sedangkan update() bisa digunakan untuk menambahkan beberapa item ke dalam set.

4. Mengukur jumlah elemen dalam set dengan fungsi len()

Untuk mengukur jumlah elemen dalam set caranya dengan menggunakan fungsi len().

5. Menghapus elemen set

Kita bisa menghapus anggota set dengan menggunakan fungsi discard() dan remove(). Perbedaannya, fungsi discard() tidak akan memunculkan error bila anggota yang ingin dihapus ternyata tidak ada di dalam set, sedangkan remove() sebaliknya.

6. Menghapus dengan fungsi pop

Didalam set ada fungsi `pop()` yang bisa digunakan untuk menghapus elemen dalam set. Karena set tidak tertata (`unordered`) maka tidak bisa diprediksi elemen mana yang akan di hapus.

7. Menghapus set dengan fungsi `clear()` dan `del()`.

Fungsi `clear()` untuk mengkosongkan set. Sedangkan, fungsi `del()` untuk menghapus keseluruhan set

8. Menggabungkan 2 set

Satu set bisa digabungkan dengan set yang lain dengan menggunakan fungsi `union()` dan fungsi `update`.

LEMBAR KERJA DAN TUGAS

1. List adalah tipe data yang paling serbaguna yang tersedia dalam bahasa Python, yang dapat ditulis sebagai daftar nilai yang dipisahkan koma (item) antara tanda kurung siku.

```
#Membuat list kosong
Buah = []
#Membuat List Dengan 1 Item
Sayur = ['Bayam']

print(Buah)
print(Sayur)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/T
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_1_ferdy060.py"
[]
['Bayam']
```

```
#Contoh Sederhana Pembuatan List Pad Bahasa Jepang
Raman Python
list1 = ['informatika', 'umsida', 2025, 2026]
list2 = [1, 2, 3, 4, 5 ]
list3 = ['a', 'b', 'c', 'd']

print(list1)
print(list2)
print(list3)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/T
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_2_ferdy060.py"
['informatika', 'umsida', 2025, 2026]
[1, 2, 3, 4, 5]
['a', 'b', 'c', 'd']
```

2. Untuk mengakses nilai dalam list python, gunakan tanda kurung siku [].

```
#contoh 1
angka = [1, 2, 3, 4, 5]
#contoh 2
List = ['ini', 'adalah', 'contoh','list']
#contoh 3
Kampus = ['sidoarjo' , 2021, "A", "B"]

print (angka[3])
print (List[2])
print (Kampus[0])
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_3_ferdy060.py"
4
contoh
sidoarjo
```

```
#cara mengakses nilai di dalam list python
list1 = ['informatika', 'umsida', 2020, 2021]
list2 = [1, 2, 3, 4, 5, 6, 7]

print ("list1[0]: ", list1[0])
print ("list2[1:5]: ", list2[1:5])
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_4_ferdy060.py"
list1[0]: informatika
list2[1:5]: [2, 3, 4, 5]
```

3. Untuk update nilai dalam list python dengan menggunakan metode append ().

```
list = ['aslab', 'informatika',1945, 2021]
print ("Sekarang kita berada pada tahun ; ",list[2])

list[2] = 2022
print ("Sekarang kita berada pada tahun : ", list[2])
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_5_ferdy060.py"
Sekarang kita berada pada tahun ; 1945
Sekarang kita berada pada tahun : 2022
```

```
list = ['aslab', 'informatika', 2020, 2021]
print ("Nilai ada pada index 2 ; ",list[2])

list[2] = 2023
print ("Nilai baru pada index 2 : ", list[2])
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modu;5_6_ferdy060.py"
Nilai ada pada index 2 ; 2020
Nilai baru pada index 2 : 2023
```

4. Untuk menghapus nilai di dalam list python, dapat menggunakan metode `remove()`

```
#contoh cara menghapus nilai pada list python
list = ['aslab', 'informatika', 1945, 2021]
print (list)
del list[2]
print ("Setelah dihapus : ", list)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TU
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_7_ferdy060.py"
['aslab', 'informatika', 1945, 2021]
Setelah dihapus :  ['aslab', 'informatika', 2021]
```

Contoh program yang menggunakan list python ganjil genap

```
#program contoh yang memanfaatkan list genap
num = [-1, 2, 53, 5, 50, 153, 91, 87]
genap = [x for x in num if x%2==0]
print ("list angka", num )
print ("angka genap pada list ", genap)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_8_ferdy060.py"
list angka [-1, 2, 53, 5, 50, 153, 91, 87]
angka genap pada list  [2, 50]
```

```
#program contoh yang memanfaatkan list ganjil
num = [-1, 2, 53, 5, 50, 153, 91, 87]
ganjil = [x for x in num if x%2!=0]
print ("\nlist angka", num )
print ("angka ganjil pada list ", ganjil)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_9_ferdy060.py"
list angka [-1, 2, 53, 5, 50, 153, 91, 87]
angka ganjil pada list  [-1, 53, 5, 153, 91, 87]
```

5. Membuat tuple semudah memasukkan nilai-nilai yang dipisahkan koma. Secara opsional, dapat memasukkan nilai-nilai yang dipisahkan koma ini di antara tanda kurung juga.

```
tup1 = ('fisika', 'kimia', 1993, 2017)
tup2 = (1, 2, 3, 4, 5)
tup3 = "a", "b", "c", "d"

print (tup1)
print (tup2)
print (tup3)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TU
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_10_ferdy060.py"
('fisika', 'kimia', 1993, 2017)
(1, 2, 3, 4, 5)
('a', 'b', 'c', 'd')
```

```
kosong = ()
print (kosong)
```

```
PS D:\TUGAS INFORMATIKA KULIAH FERDY SEMUA SEMESTER\SEMESTER 1\
HRIAL/AppData/Local/Programs/Python/Python310/python.exe "d:/TU
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_11_ferdy060.py"
()
```

untuk membuat Tuple yang hanya berisi satu (singleton), maka harus menambahkan tanda koma di belakangnya. Karena jika tidak ditambahkan koma, akan dianggap sebagai string.

```
prodi = (' Informatika ',)
fakultas = "Saintek",
print (prodi)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_12_ferdy060.py"
(' Informatika ',)
```

6. Untuk mengakses nilai dalam tuple, gunakan tanda kurung siku [].

```
tup1 = ('informatika', 'mesin', 2000, 2021)
tup2 = (1, 2, 3, 4, 5, 6, 7)

print ("tup1[0] : ", tup1[0])
#slicing
print ("tup2[1:5]: ", tup2[1:5])
#Output
#tup1 [0] : Informatika
#tup2[1:5]: (2, 3, 4, 5)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_13_ferdy060.py"
tup1[0] : informatika
tup2[1:5]: (2, 3, 4, 5)
```

7. secara eksplisit menghapus keseluruhan tuple, cukup gunakan del statement.

```
tup = ('Informatika', 'Mesin', 2000, 2021)
print (tup)

del tup

tup = ('Industri', 'Elektro', 2020)
print ("Setelah menghapus tuple :", tup)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_14_ferdy060.py"
('Informatika', 'Mesin', 2000, 2021)
Setelah menghapus tuple : ('Industri', 'Elektro', 2020)
```

Tuple Nested

```
tuple1 = ("ini", "tuple", 1)
tuple2 = ("ini", "tuple", 2)

tuple3 = (tuple1, tuple2)
print (tuple3)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_15_ferdy060.py"
(('ini', 'tuple', 1), ('ini', 'tuple', 2))
```

8. Untuk mengakses elemen Dictionary, Anda dapat menggunakan tanda kurung siku []

```
#Contoh cara membuat Dictionary pada Python
dict = {'Name': 'Andika', 'Age': 18, 'Class': 'First'}
print ("dict['Name'] :", dict['Name'])
print ("dict['Age'] :", dict['Age'])
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_16_ferdy060.py"
dict['Name'] : Andika
dict['Age'] : 18
```

9. Update Nilai Dalam Dictionary Python.

```
dict = {'Name': 'Andika', 'age': 15, 'Class': 'First'}
dict['Age'] = 19
dict['University'] = "Universitas Muhammadiyah Sidoarjo"

print ("dict ['Age']:", dict['Age'])
print ("dict ['University']:", dict['University'])
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_17_ferdy060.py"
dict ['Age']: 19
dict ['University']: Universitas Muhammadiyah Sidoarjo
```

10. Hapus Nilai Dalam Dictionary Python.

```
#Update dictionary python

dict = {'Name': 'Andika', 'age': 15, 'Class': 'First'}
dict['Age'] = 19      # Mengubah enty yang sudah ada
dict['University'] = "Universitas Muhammadiyah Sidoarjo"

print ("dict ['Age']:", dict['Age'])
print ("dict ['University']:", dict['University'])
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_18_ferdy060.py"
dict ['Age']: 19
dict ['University']: Universitas Muhammadiyah Sidoarjo
```

```
#Update dictionary python
```

```
dict = {'Name': 'Andika', 'age': 15, 'Class': 'First'}
```

```
del dict ['Name']    #hapus enty dengan key 'Name'
```

```
dict.clear()         #hapus semua entry di dict
```

```
del dict             #hapus dictionary yang sudah ada
```

```
print ("dict ['Age']:", dict['Age'])
```

```
print ("dict ['University']:", dict['University'])
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_19_ferdy060.py"
```

```
dict ['Age']: dict['Age']
```

```
dict ['University']: dict['University']
```

```
#contoh penggunaan dictionary
```

```
print("\ncontoh 1")
```

```
dic1={1:100, 2:200}
```

```
dic2={3:300, 4:400}
```

```
dic3={5:500, 6:600}
```

```
dic4= {}
```

```
for d in (dic1, dic2, dic3) : dic4.update(d)
```

```
print(dic4)
```

```
print("\ncontoh 2")
```

```
d = {1:100, 2:200, 3:300, 4:400, 5:500, 6:600}
```

```
cek_angka = 3
```

```
if cek_angka in d:
```

```
    print("Angka",cek_angka,'tersedia      dalam  
dictionary.')
```

```
else:
```

```
    print("Angka",cek_angka,'tidak tersedia dalam  
dictionary')
```

```
print("\ncontoh 3")
```

```
d1 = {'a': 1000, 'b':2000}
```

```
d2 = {'x': 3000, 'y':2000}
```

```
d = d1.copy()
```

```
d.update(d2)
```

```
print(d)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_20_ferdy060.py"
```

```
contoh 1
```

```
{1: 100, 2: 200, 3: 300, 4: 400, 5: 500, 6: 600}
```

```
contoh 2
```

```
Angka 3 tersedia dalam dictionary.
```

```
contoh 3
```

```
{'a': 1000, 'b': 2000, 'x': 3000, 'y': 2000}
```

11. Karena set tidak terindeks maka untuk mengaksesnya tidak bisa menggunakan indeks seperti tuple dan list. Untuk mengakses elemen-elemen dalam set bisa dengan menggunakan perulangan (for ...).

```
#Menggunakan set
set_prodi = {"Informatika", "Industry", "Mesin", "Elektro"}
print (set_prodi)
#Set menggunakan tanda kurung kurawa {}
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_21_ferdy060.py"
{'Elektro', 'Informatika', 'Mesin', 'Industry'}
```

12. Mengecek elemen set

```
set_prodi = {"Informatika", "Industry", "Mesin", "Elektro"}
if "Informatika" in set_prodi:
    print ("informatika ada di dalam set prodi")
#Memasukkan fungsi kondisi untuk mengecek keberadaan item
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_22_ferdy060.py"
informatika ada di dalam set prodi
```

Pada perintah diatas diberikan masukan berupa set_prodi. Kemudian, diberikan perintah if 'informatika' in set_prodi untuk memeriksa elemen set menggunakan fungsi if yang diikuti perintah print. Perintah tersebut berarti apabila terdapat elemen 'informatika', akan menghasilkan luaran 'informatika' ada dalam set_prodi.

13. Menambahkan item kedalam set

```
set_prodi = {"Informatika", "Industry", "Mesin", "Elektro"}
set_prodi.add ("teknik")
print (set_prodi)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_23_ferdy060.py"
{'Industry', 'Informatika', 'Mesin', 'teknik', 'Elektro'}
```

14. Menghapus set dengan fungsi clear().

```
set_prodi = {"Informatika", "Industry", "Mesin", "Elektro"}
set_prodi.clear()
print (set_prodi)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_24_ferdy060.py"
set()
```

15. Menggabungkan 2 set

```
prodi = {"Informatika", "Industry", "Mesin", "Elektro"}
fakultas = {"fst", "fpip", "fbhis", "fikes"}

gabungan1 = prodi.union(fakultas)
print(gabungan1)

prodi.update(fakultas)
print(gabungan1)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_25_ferdy060.py"
{'Industry', 'Elektro', 'fst', 'fikes', 'Informatika', 'fbhis', 'fpip', 'Mesin'}
{'Industry', 'Elektro', 'fst', 'fikes', 'Informatika', 'fbhis', 'fpip', 'Mesin'}
```

TUGAS

Menggunakan Daftar Lists sebagai Antrian Queues

jamal, samsul dan adnan mengantri untuk berobat di rumah sakit X, tidak lama kemudian udin dan baha masuk ke daftar antrian tersebut. setelah dipanggil, jamal meninggalkan antrian untuk diperiksa, begitupun dengan antrian setelahnya. **Implementasikan list yang terdapat pada cerita tersebut kedalam antrian queue dan gunakan collections deque untuk mengimplementasikannya!**

```
from collections import deque # import module
antrian = deque(['jamal', 'samsul', 'adnan']);
print("antrian pertama yang berobat di rumah sakit X
adaalah :", antrian)
print("antrian bertambah 2 orang")
antrian.append("udin")
antrian.append("baha")
print("antrian saat ini menjadi:", antrian)
print("setelah itu jamal meninggalkan antrian dan
diperiksa!")
antrian.popleft()
print("jumlah antrian saat ini adalah:", antrian)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul5_tugas_ferdy060.py"
antrian pertama yang berobat di rumah sakit X adaalah : deque(['jamal', 'samsul', 'adnan'])
antrian bertambah 2 orang
antrian saat ini menjadi: deque(['jamal', 'samsul', 'adnan', 'udin', 'baha'])
setelah itu jamal meninggalkan antrian dan diperiksa!
jumlah antrian saat ini adalah: deque(['samsul', 'adnan', 'udin', 'baha'])
```



**LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 - 2026**

Lembar Asistensi

Praktikum Algoritma dan Pemrograman

Pokok Bahasan 6

Judul : Struktur Pemograman : Function
Nama : Mochammad Ferdyansyah
NIM : 251080200060
Kelompok : 8
Dilaksanakan : 08 Januari 2026

Mengetahui,

Dosen Praktikum

Asisten Praktikum

(Uce Indahyanti, S.Kom. M.Kom.)

(Muhamad Maulana Musafir)

POKOK BAHASAN 6

STRUKTUR PEMROGRAMAN : FUNCTION

6.1 FUNCTION

Fungsi pada python adalah kumpulan perintah atau baris kode yang dikelompokkan menjadi satu kesatuan untuk kemudian bisa dipanggil atau digunakan berkali-kali. Sebuah fungsi bisa menerima parameter, bisa mengembalikan suatu nilai, dan bisa dipanggil berkali-kali secara independen. Dengan fungsi kita bisa memecah program besar yang kita tulis, menjadi bagian-bagian kecil dengan tugasnya masing-masing.

Fungsi dipakai untuk mengumpulkan beberapa perintah yang sering dipakai dalam sebuah program. Dengan memakai fungsi, program yang dibuat menjadi lebih terstruktur. Lebih mudah diikuti oleh orang lain yang membaca program dibuat. Paling penting adalah mempersingkat waktu yang diperlukan untuk mengembangkan suatu perangkat lunak. Karena perangkat lunak yang dibuat, bisajadi memakai komponen-komponen yang sama.

Seperti layaknya sebuah bahasa pemrograman, Python juga memberikan fasilitas pembuatan fungsi yang sangat bagus. Konsep fungsi dalam Python sama dengan bahasa pemrograman C/C++. Python menganggap fungsi dan prosedur adalah sesuatu yang sama, dalam artian cara mendeklarasikan fungsi dan prosedur adalah sama. Hanya bedanya, kalau fungsi mengembalikan suatu nilai setelah proses sedangkan prosedur tidak.

6.2 Struktur Function

Di dalam python, sintaks pembuatan fungsi terlihat seperti berikut:

```
def namaFungsi (parameter)
• statemens
```

- Sebuah fungsi diawali dengan statemen defkemudian diikuti oleh sebuah nama_fungsi nya. Pernyataan def dipakai untuk mendeklarasikan fungsi.
- Sebuah fungsi dapat memiliki daftar argumen (parameter) ataupun tidak.

- Tanda titik dua (:) menandakan awal pendefinisian tubuh dari fungsi yang terdiri dari statemen-statemen

6.3 Memanggil Function

Di dalam python, sintaks memanggil fungsi sangat sederhana seperti berikut:

```
namaFungsi()
```

Dalam deklarasi fungsi, juga bisa menambahkan komentar-komentar yang memberi penjelasan mengenai fungsi yang dibuat. Secara umum memang bisa menambahkan komentar-komentar di sembarang tempat dalam program yang dibuat. Baris-baris komentar diawali dengan karakter pagar (#). Semua karakter yang mengikuti tanda ini sampai akhir baris dianggap sebagai komentar dan tidak akan mempengaruhi jalannya program. Akan tetapi terdapat satu gaya pemberian komentar dalam Python yang disebut dengan docstring. Biasanya dipakai untuk memberi penjelasan mengenai fungsi atau objek. Docstring diapit dengan tanda petik ganda, komentar jenis ini hanya boleh diberikan tepat satu baris dibawah deklarasi fungsi atau objek yang akan ditunjukkan pada pembahasan selanjutnya. Docstring sangat bermanfaat ketika kita ingin mendokumentasikan semua fungsi dan kelas yang telah kita buat. Karena ada beberapa perangkat lunak yang mampu membuat dokumentasi berdasarkan docstring yang ada dalam source code.

6.4 Function dengan Parameter dan Argumen

Sebuah fungsi juga bisa menerima parameter atau pun argumen. Ia merupakan suatu nilai/variabel yang dilemparkan ke dalam fungsi untuk diproses lebih lanjut. Parameter adalah sebutan untuk nilai inputan fungsi pada saat fungsi itu di definisikan, sedangkan argumen adalah sebutan untuk nilai inputan fungsi pada saat fungsi itu dipanggil.

Tergantung kebutuhan, sebuah fungsi bisa menerima 1, 2, atau lebih dari 5 parameter atau argumen, namun bisa juga tidak memerlukan sama sekali. Sebenarnya parameter dan argumen sangat mirip dan sering di pertukarkan. Perbedaan dari keduanya hanya di posisi pembuatan saja.

Parameter merujuk kepada inputan fungsi pada saat pendefinisian, sedangkan argumen merujuk ke nilai input fungsi pada saat pemanggilan.

Berikut format dasar fungsi Python dengan parameter dan argumen:

```
def namaFungsi (param1, param2):  
    ...  
    return nilai  
namaFungsi(arg1, arg2)
```

Di akhir baris 1, yakni dalam tanda kurung setelah nama_function, adalah tempat untuk penulisan parameter. Dalam contoh di atas, terdapat 2 buah parameter bernama param1 dan param 2. Sepanjang isi function, param1 dan param2 bisa di akses sebagaimana variabel biasa. Ketika memanggil fungsi, kita harus isi kedua nilai seperti baris 6, yakni dari perintah nama_function(arg1, arg2). Disini arg1, arg2 adalah argumen. Nilai argumen ini nantinya akan mengisi nilai parameter.

Lembar kerja dan tugas

1. Membuat dan Memanggil Function

```
def hello() :  
    print ("Hello World")  
hello()
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_1_ferdy060.py"  
Hello World
```

Sekarang buat function dengan namaKalian dan print dengan nama, nim kalian masing-masing.

```
def namaKalian() :  
    print ("Mochammad Ferdiansyah_251080200060")  
namaKalian()
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_2_ferdy060.py"  
Mochammad Ferdiansyah_251080200060
```

Pada kode di bawah ini terdapat error, perbaiki kode tersebut sehingga dapat dijalankan.

```
def coba () :  
    print ("Hai nama saya ..., saya kuliah di umsida")  
  
coba()
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_3_ferdy060.py"  
Hai nama saya ..., saya kuliah di umsida
```

Dapat juga menggunakan sebuah fungsi dengan mengembalikan sebuah nilai berikut ontho fungsi untuk menghitung penjumlahan.

```
def hitung () :  
    return 10+10  
tampil = hitung()  
print ("Hasil Perhitungan : ", tampil)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_4_ferdy060.py"  
Hasil Perhitungan : 20
```

Pada perintah diatas setelah proses penjumlahan maka hasilnya akan dikembalikan ke fungsi memanggil dengan menggunakan pernyataan return, sehubungan ada nilai yang dikembalikan maka diperlukan variabel penampung yang dalam hal ini menggunakan variabel tampil baru selanjutnya hasilnya ditampilkan.

2. Parameter Function

Cara menggunakan fungsi dengan parameter dapat di contohkan dengan kode berikut:

```
def praktikum(bahasa) :  
    print ("Saya sedang praktikum dengan menggunakan  
    bahasa pemrograman", bahasa)  
  
    praktikum("Python")
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_5_ferdy060.py"  
Saya sedang praktikum dengan menggunakan bahasa pemrograman Python
```

Pada kode diatas, bahasa merupakan parameter untuk menampung nilai dari fungsi dengan nama praktikum yang isinya Python. Sekarang pada kode dibawah mencoba menggunakan 2 parameter untuk menampung sebuah nilai .

```
def bio (nama, hobi) :  
    print ("Hai namaku", nama, "Saya mempunyai hobi",  
    hobi)  
  
    bio ("Mochammad Ferdiansyah", "Berenang")
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_6_ferdy060.py"  
Hai namaku Mochammad Ferdiansyah Saya mempunyai hobi Berenang
```

Pada kode di atas nama merupakan parameter 1 dan hobi parameter ke 2, ketika memberikan nilai pada parameter tersebut harus sesuai dengan urutannya. Dan menggabungkan string dengan parameter menggunakan koma (,).

3. Menghitung dengan menggunakan Function

kita juga bisa menghitung sebuah nilai dengan menggunakan function, sebagai contoh menghitung luas segitiga.

```
def luas_segitiga(alas, tinggi) :  
    luas = (alas * tinggi) / 2  
    print ("Luas Segitiga adalah : ", luas)  
  
    luas_segitiga (5, 15)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_7_ferdy060.py"  
Luas Segitiga adalah : 37.5
```

Dapat menggunakan cara dengan mengembalikan nilai dari hasil pemrosesannya. Cara mengembalikan nilai dengan kata kunci **return**.

```
def luas_segitiga(alas, tinggi) :  
    luas = (alas * tinggi) / 2  
    print ("Luas Segitiga adalah : ", luas)  
  
print ("Luas Segitiga adalah : ", luas_segitiga (5,  
15))
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/moduol6_8_ferdy060.py"  
Luas Segitiga adalah : 37.5  
Luas Segitiga adalah : None
```

pada fungsi luas_segitiga() kita melakukan print dari hasil pemrosesan secara langsung di dalam fungsinya. Sedangkan fungsi luas_persegi(), kita melakukan print pada saat pemanggilanya. Jadi, fungsi luas_persegi() akan bernilai sesuai dengan hasil yang dikembalikan. Sehingga kita dapat memanfaatkannya untuk pemrosesan berikutnya.

Kita juga dapat menghitung umur dengan menggunakan function

```
def hitung_umur(thn_sekarang, thn_lahir) :  
    umur = thn_sekarang - thn_lahir  
    print ("Umurku sekarang : " , umur , "tahun")  
  
hitung_umur (2026, 2006)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_9_ferdy060.py"  
Umurku sekarang : 20 tahun
```

4. Menggunakan input dengan Function

```
def makanan(makan, minum) :  
    print ("Saya sedang makan : ",makan)  
    print ("Saya sedang minum : ",minum)  
    makan = input ("Masukkan nama makanan : ")  
    minum = input ("Masukkan nama minuman : ")  
    makanan (makan, minum)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul9_10_ferdy060.py"  
Masukkan nama makanan : Nasi Rendang  
Masukkan nama minuman : Air Mineral  
Saya sedang makan : Nasi Rendang  
Saya sedang minum : Air Mineral
```

pada perintah diatas parameter-parameter yang akan dilewatkan ke dalam fungsi didaftarkan dalam tanda kurung yaitu parameter nama dan kota. Masing-masing parameter dipisahkan oleh koma (,) . Setelah dilewatkan selanjutnya di dalam fungsi akan ditampilkan berdasarkan nilai yang dikirim. Sekarang kalian coba **buat program untuk menginput nama, nim dan kelompok praktikum?** Konsepnya sama seperti kode diatas

```
def kelompok (nama, nim, kelompok_praktikum) :  
    print("Nama :", nama)  
    print("NIM :", nim)  
    print("Kelompok Praktikum :", kelompok_praktikum)  
nama = input ("Masukkan Nama Anda : ")  
nim = input ("Masukkan NIM anda :")  
kelompok_praktikum = input ("Masukkan Kelompok  
Praktikum : ")  
  
kelompok(nama, nim, kelompok_praktikum)
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_11_ferdy060.py"  
Masukkan Nama Anda : Mochammad Ferdyansyah  
Masukkan NIM anda :251080200060  
Masukkan Kelompok Praktikum : Kelompok 8  
Nama : Mochammad Ferdyansyah  
NIM : 251080200060  
Kelompok Praktikum : Kelompok 8
```

Selain itu kita juga bisa menghitung nilai dari sebuah inputan

```
def penjumlahan(bil1, bil2) :  
    hasil = bil1 + bil2  
    return hasil  
bil1 = int (input("Masukkan bilangan pertama : "))  
bil2 = int (input("Masukkan bilangan kedua : "))  
print ("Hasil Penjumlahan adalah  
:",penjumlahan(bil1,bil2))
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_12_ferdy060.py"  
Masukkan bilangan pertama : 10  
Masukkan bilangan kedua : 6  
Hasil Penjumlahan adalah : 16
```

Kalian coba **buat program untuk pengurangan, perkalian, pembagian dengan menginputkan bilangan sendiri?** Konsepnya sama seperti kode di atas

```
def pengurangan(bil1, bil2) :  
    hasil = bil1 - bil2  
    return hasil  
bil1 = int (input("Masukkan bilangan pertama : "))  
bil2 = int (input("Masukkan bilangan kedua : "))  
print      ("Hasil      Penjumlahan      adalah  
:",pengurangan(bil1,bil2))
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_13_ferdy060.py"  
Masukkan bilangan pertama : 20  
Masukkan bilangan kedua : 5  
Hasil Penjumlahan adalah : 15
```

```
def perkalian(bil1, bil2) :  
    hasil = bil1 * bil2  
    return hasil  
bil1 = int (input("Masukkan bilangan pertama : "))  
bil2 = int (input("Masukkan bilangan kedua : "))  
print      ("Hasil      Perkalian      adalah  
:",perkalian(bil1,bil2))
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_14_ferdy060.py"  
Masukkan bilangan pertama : 50  
Masukkan bilangan kedua : 25  
Hasil Perkalian adalah : 1250
```

```
def pembagian(bil1, bil2) :  
    hasil = bil1 / bil2  
    return hasil  
bil1 = float (input("Masukkan bilangan pertama : "))  
bil2 = float (input("Masukkan bilangan kedua : "))  
print      ("Hasil      Pembagian      adalah  
:",pembagian(bil1,bil2))
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_15_ferdy060.py"  
Masukkan bilangan pertama : 100  
Masukkan bilangan kedua : 50  
Hasil Pembagian adalah : 2.0
```

Tugas

1. Buatlah program dengan menngisikan biodata yang isisnya terdapat: Nama, Nim, TTL, Jenis Kelamin, Agama, Alamat, No Telp, Email.
2. Buatlah program untuk mengkonversi suhu Celcius, Fanrenheit, Kelvin, Reamur
3. Buatlah program untuk menentukan bilangan ganjil atau genap

Jawab:

1. Biodata

```
# Input biodata
nama = input("Masukkan nama: ")
nim = input("Masukkan NIM: ")
ttl = input("Masukkan TTL: ")
jenis_kelamin = input("Masukkan jenis kelamin: ")
agama = input("Masukkan agama: ")
alamat = input("Masukkan alamat: ")
no_telp = input("Masukkan nomor telepon: ")
email = input("Masukkan email: ")

# Menampilkan biodata
print(f"\nBiodata:\n{'='*20}")
print(f>Nama: {nama}")
print(f"NIM: {nim}")
print(f"TTL: {ttl}")
print(f>Jenis Kelamin: {jenis_kelamin}")
print(f>Agama: {agama}")
print(f>Alamat: {alamat}")
print(f>No Telp: {no_telp}")
print(f>Email: {email}")
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_tugas1biodata_ferdy060.py"
Masukkan NIM: 251080200060
Masukkan TTL: Pasuruan, 25 Agustus 2006
Masukkan jenis kelamin: Laki-Laki
Masukkan agama: Islam
Masukkan alamat: Jl. Cempakah putih No.31
Masukkan nomor telepon: +6281234932830
Masukkan email: policeferdy@gmail.com

Biodata:
=====
Nama: Mochammad Ferdyansyah
NIM: 251080200060
TTL: Pasuruan, 25 Agustus 2006
Jenis Kelamin: Laki-Laki
Agama: Islam
Alamat: Jl. Cempakah putih No.31
No Telp: +6281234932830
Email: policeferdy@gmail.com
```

2. Suhu

```
# Fungsi konversi suhu dari Celcius ke Fahrenheit
def celcius_to_fahrenheit(celcius):
    fahrenheit = (celcius * 9/5) + 32
    return fahrenheit

# Fungsi konversi suhu dari Celcius ke Kelvin
def celcius_to_kelvin(celcius):
    kelvin = celcius + 273.15
    return kelvin

# Fungsi konversi suhu dari Celcius ke Reamur
def celcius_to_reamur(celcius):
    reamur = celcius * 4/5
    return reamur

# Input suhu dalam skala Celcius
celcius = float(input("Masukkan suhu dalam Celcius:"))

# Konversi suhu ke skala Fahrenheit, Kelvin, dan Reamur
fahrenheit = celcius_to_fahrenheit(celcius)
kelvin = celcius_to_kelvin(celcius)
reamur = celcius_to_reamur(celcius)

# Menampilkan hasil konversi suhu
print(f"\n{celcius} derajat Celcius setara dengan:")
print(f"{fahrenheit} derajat Fahrenheit")
print(f"{kelvin} derajat Kelvin")
print(f"{reamur} derajat Reamur")
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_tugas2suhu_ferdy060.py"
Masukkan suhu dalam Celcius:16

16.0 derajat Celcius setara dengan:
60.8 derajat Fahrenheit
289.15 derajat Kelvin
12.8 derajat Reamur
```

```
# Input bilangan
bilangan = int(input("Masukkan bilangan: "))

# Menentukan apakah bilangan ganjil atau genap
if bilangan % 2 == 0:
    print(f"{bilangan} adalah bilangan genap.")
else:
    print(f"{bilangan} adalah bilangan ganjil.")
```

```
/SEMESTER 1/TUGAS ALGORITMA/PRAKTIKUM 2/modul6_tugas3bilangan_ferdy060.py"
Masukkan bilangan: 25
25 adalah bilangan ganjil.
```

Referensi :

1. Universitas pancasila. *Modul Praktikum Dasar Pemrograman Dengan Bahasa Python*
2. Petani Kode. *Python Fungsi*



KARTU ASISTENSI
LABORATORIUM INFORMATIKA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MUHAMMADIYAH SIDOARJO
2025 – 2026

Nama : Mochammad Ferdiansyah

NIM : 251080200060

Kelompok : 8

No	Judul Praktikum	Tanggal Asistensi	Tanggal Asistensi	Catatan Asisten
1.	Flowchart dan Instalasi	13 November 2025		
2.	Tipe Data dan Operator Aritmatika	13 November 2025		
3.	Struktur Pemograman : Berkondisi	13 November 2025		
4.	Struktur Pemrograman : Perulangan	08 Januari 2026		
5.	Tipe Data Koleksi	08 Januari 2026		
6.	Struktur Pemrograman : Function	08 Januari 2026		

Mengetahui,

Dosen Praktikum

Asisten Praktikum

(Uce Indahyanti, S.Kom., M.Kom.)

(Muhamad Maulana Musafir)