



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

NIM	71251187
Nama Lengkap	Benedictina Andhika Chinor Jodie Soesila
Minggu ke / Materi	02 / Variable, Expression dan Statements

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2026

BAGIAN 1: MATERI MINGGU INI (40%)

Values dan type

Value adalah elemen dasar dalam program, seperti angka dan huruf, yang terbagi ke dalam beberapa tipe data, misalnya integer dan string. Interpreter mengenali tipe data, terutama string, melalui penggunaan tanda petik. Pemahaman tentang value dan type dapat kita latih menggunakan percobaan kode di Python interactive mode berikut ini.

```
>>> print(4)
4
>>> print(10.876)
10.876

>>> print('Z')
Z
>>> print('True')
True
>>> print('False')
False
```

Gambar 1.1 Tampilan Python Interactive Mode dalam Menguji Value dan Type.

Sumber: <https://shorturl.at/3l4g0>

Perintah **print** dapat digunakan untuk menampilkan berbagai tipe value, seperti integer, float, character dan boolean. Setiap value memiliki tipe data tertentu, dan Python menyediakan fungsi bawaan **type()** untuk mengecek tipe data dari suatu value.

```
>>> x=5
>>> print(x, "tipenya adalah ", type(x))
5 tipenya adalah <class int>
>>> x = 2.0
>>> print(x, "tipenya adalah ", type(x))
2.0 tipenya adalah <class float>
>>> x = 1+2j
>>> print(x, "tipenya adalah ", type(x))
(1+2j) tipenya adalah <class complex>
```

Gambar 1.2 Tampilan Output Berbagai Tipe Data pada Python.

Sumber: <https://shorturl.at/3l4q0>

Penulisan bilangan bulat besar sering menggunakan tanda pemisah setiap tiga digit. Dalam Python, penulisan tersebut tetap dianggap sebagai bilangan bulat.

```
>>> print(1,000,000)
1,0,0
```

Gambar 1.3 Penulisan Bilangan Bulat Besar pada Python. Sumber: <https://shorturl.at/3l4q0>

Python menganggap penulisan 1,000,000,000 sebagai beberapa parameter terpisah pada fungsi **print**, bukan sebagai bilangan utuh.

Variabel

Variabel adalah fitur penting dalam bahasa pemrograman yang berfungsi sebagai lokasi memori untuk menyimpan data. Nilai di dalam variabel dapat berubah selama program dijalankan melalui berbagai operasi.

```
>>> pesan = 'selamat pagi, mari belajar python'
>>> n = 17
>>> pi = 3.1415926535897931
```

Gambar 1.4 Konsep Variabel sebagai Penyimpan Data dalam Program.

Sumber: <https://shorturl.at/3l4g0>

Variabel dalam Python bersifat dinamis dan dapat menyimpan berbagai tipe data tanpa perlu deklarasi tipe terlebih dahulu. Nilai variabel dapat diubah saat program berjalan dan ditampilkan menggunakan perintah **print**.

```
>>> print(n)
17
>>> print(ipk)
3.29
```

Gambar 1.5 Penggunaan Variabel Dinamis pada Python. Sumber: <https://shorturl.at/3l4q0>

Nama Variabel dan Keywords

Penamaan variabel dalam Python harus mengikuti aturan tertentu, seperti diawali huruf atau garis bawah, dapat mengandung angka, bersifat case sensitive, dan harus menggunakan nama yang valid serta konsisten.

```

#proses memasukan data ke dalam variabel
nama = "Agung Sejagat"

#proses mencetak variabel
print(nama)

#nilai dan tipe data dalam variabel dapat diubah
umur = 20
print(umur)
type(umur)
umur = "dua puluh satu" #nilai setelah diubah
print(umur) #mencetak nilai umur
type(umur) #mengecek tipe data umur
namaDepan = "Joko"
namaBelakang = "Widodo"
nama = namaDepan + " " + namaBelakang
umur = 22
hobi = "Berenang"
print("Biodata\n", nama, "\n", umur, "\n", hobi)

#contoh variabel lainya
inivariabel = "Balo"

ini_juga_variabel = "Bali"
inivariabeljuga = "Hi"
inivariabel222 = "Bye"
panjang = 10
lebar = 5
luas = panjang * lebar
print(luas)

```

Gambar 1.6 Contoh Penulisan Nama Variabel. Sumber: <https://shorturl.at/3l4q0>

Statements

Statements adalah bagian dari kode Python, contohnya adalah **print**, **expression statement**, dan **assignment**. Dalam mode interaktif, hasil dieksekusi dan langsung ditampilkan, sedangkan pada script mode interaktif, hasil dieksekusi dan langsung ditampilkan, sedangkan pada script modde, statements dijalankan secara berurutan dalam sebuah file program.

```

1 print(1)
2 x=2
3 print(x)

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

192:~ macintosh$ /Library/Frameworks/Python.framework/Versions/3.7/bin/python3 "
python/source-code/bab 02/statemet.py"
1
2
192:~ macintosh$ █

```

Gambar 1.7 Contoh Statement dan Outputnya. Sumber: <https://shorturl.at/3l4q0>

Operator dan Operand

Operator adalah simbol yang digunakan untuk melakukan operasi aritmatika atau logika, sedangkan nilai yang dikenai operasi disebut operand. Contohnya pada $2 + 3$, tanda “+” adalah operator sedangka “2” dan “3” adalah operand. Pada bagian ini dibahas operator aritmatika dalam Python, yaitu operator yang digunakan untuk operasi matematika seperti penjumlahan, pengurangan, perkalian, dan pembagian.

```

>>>32 + 30
62

>>> hour = 5
>>> print (hour-1)
4
>>> minute=60
>>> print (hour*6+minute)
90
>>> print (minute/60)
1.0
>>> 5**2
25
>>> (5+9)*(15-7)
112

```

Gambar 1.8 Contoh Penggunaan Operator Aritmatika pada Python.

Sumber:<https://shorturl.at/3l4g0>

Expressions

Expression adalah representasi nilai yang dapat berupa gabungan antara value, variabel, dan operator. Value dan variabel juga termasuk ke dalam expression. Ketika menggunakan expression dalam model interactive, interpreter akan melakukan evaluasi dan menampilkan hasilnya.

```

17
x
x+17

```

Gambar 1.9 Konsep Expression dalam Python. Sumber:<https://shorturl.at/3l4q0>

```

>>> 1 + 1
2
>>> 3 + 2
5

```

Gambar 1.10 Evaluasi Expression pada Python Interactive Mode.

Sumber:<https://shorturl.at/3l4q0>

Urutan Operasi

Jika dalam sebuah expression terdapat lebih dari satu operator, maka urutan pengerjaannya mengikuti aturan prioritas (order of operations). Python mengikuti aturan matematika yang dikenal dengan PEMDAS: Parentheses, Exponentiation, Multiplication and Division, serta Addition and Subtraction. Operasi dengan prioritas yang sama dikerjakan dari kiri ke kanan.

Operator Modulus dan String Modulus

Operator modulus digunakan untuk menghilangkan sisa hasil pembagian antara dua bilangan. Dalam Python, operator ini dilambangkan dengan tanda persen (%) dan umumnya digunakan pada tipe data integer.

String

Operator + ketika bekerja dengan string tidak berarti penjumlahan secara matematika, melainkan penggabungan antar string.

Menangani Input dari Pengguna

Program umumnya mengikuti alur Input-Proses-Output. Input adalah data yang dibutuhkan program, proses adalah langkah pengolahan data, dan output adalah hasil akhir yang diperoleh. Contohnya dapat dilihat pada proses pengambilan uang di ATM. Proses pengambilan uang di ATM dapat dijelaskan dengan konsep Input-Proses-Output. Memasukkan kartu, PIN, memilih menu, dan nominal uang termasuk Input. Pengecekan saldo dan validasi kartu merupakan Proses. Uang yang keluar serta berkurangnya saldo merupakan Output.

Dalam Python, input dari pengguna dapat diterima menggunakan fungsi bawaan **input()**. Saat fungsi ini dijalankan, program akan menunggu pengguna mengetik dan menekan Enter, lalu mengembalikan nilai tersebut dalam bentuk string. Sebelum menerima input dari pengguna, sebaiknya ditampilkan prompt untuk memberi petunjuk tentang apa yang harus dimasukkan. Prompt tersebut dapat langsung ditambahkan ke dalam fungsi **input()** agar ditampilkan sebelum program menunggu masukan.

Komentar

Tanda pagar **#** digunakan untuk menandai komentar. Bagian yang ditulis sebagai komentar ini tidak akan dieksekusi oleh interpreter. Komentar dibuat semata-mata untuk membantu programmer menjelaskan atau memperjelas maksud dari kode yang ditulis.

BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

SOAL 1

Link Github: https://github.com/chinorjodie/71251187_chinor.git

Source Code:

```
# Input
tinggi = float(input("Masukkan tinggi badan (meter): "))
bmi = float(input("Masukkan BMI yang diharapkan: "))

# Proses
berat = bmi * (tinggi ** 2)

# Output
print("\nBerat badan yang diperlukan adalah:", berat, "kg")
```

Penjelasan:

Program tersebut digunakan untuk menghitung berat badan yang diperlukan berdasarkan tinggi badan dan nilai BMI yang diharapkan. Pada bagian awal, program meminta pengguna memasukkan tinggi badan dalam satuan meter dan nilai BMI yang diinginkan melalui fungsi **input()**. Karena data yang dimasukkan melalui **input()** berbentuk teks (string), maka digunakan fungsi **float()** untuk mengubahnya menjadi angka desimal agar dapat digunakan dalam perhitungan matematika. Selanjutnya, program melakukan proses perhitungan dengan rumus $\text{berat} = \text{BMI} \times \text{tinggi}^2$, di mana tanda ****** digunakan untuk menyatakan pangkat dua (kuadrat). Hasil perhitungan tersebut disimpan dalam variabel **berat**. Terakhir, program menampilkan hasilnya ke layar menggunakan fungsi **print**.

Output:



```
PROBLEMS OUTPUT PORTS DEBUG CONSOLE TERMINAL
PS D:\Praktikum-Alpro\week 2> python .\soal1.py
Masukkan tinggi badan (meter): 1.6
Masukkan BMI yang diharapkan: 25

Berat badan yang diperlukan adalah: 64.00000000000001 kg
PS D:\Praktikum-Alpro\week 2> |
```

Gambar 2.1 Output Program Menghitung Berat Badan

SOAL 2

Link Github: https://github.com/chinorjodie/71251187_chinor.git

Source Code:

```
#input
x = int(input("Masukkan nilai x:"))

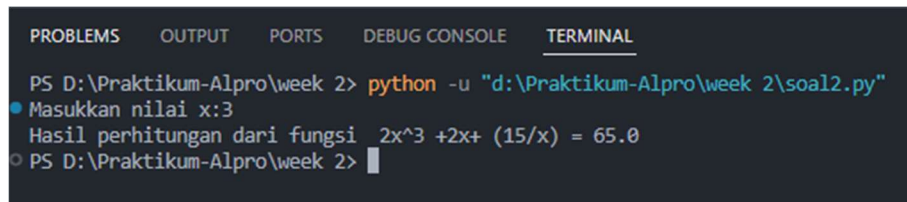
#procces
fx = 2* (x**3) + 2*x + (15/x)

#output
print(f"Hasil perhitungan dari fungsi 2x^3 +2x+ (15/x) = {fx}")
```

Penjelasan:

Program tersebut digunakan untuk menghitung nilai suatu fungsi matematika berdasarkan nilai x yang dimasukkan oleh pengguna. Pada bagian awal, program meminta pengguna memasukkan nilai x melalui fungsi `input()`. Karena data dari `input()` berbentuk teks (string), maka digunakan fungsi `int()` untuk mengubahnya menjadi bilangan bulat agar dapat diproses secara matematis. Selanjutnya, pada bagian proses, program menghitung nilai fungsi $f(x)=2x^3+2x+15/3$. Operasi $x**3$ berarti x , lalu ditambahkan dengan hasil pembagian 15 oleh x . Hasil perhitungan tersebut disimpan dalam variabel `fx`. Pada bagian akhir, program menampilkan hasil perhitungan menggunakan `print()` dengan format f-string, sehingga nilai hasil perhitungan dapat langsung dimasukkan ke dalam kalimat output secara rapi dan mudah dibaca.

Output:



```
PROBLEMS OUTPUT PORTS DEBUG CONSOLE TERMINAL
PS D:\Praktikum-Alpro\week 2> python -u "d:\Praktikum-Alpro\week 2\soal2.py"
• Masukkan nilai x:3
Hasil perhitungan dari fungsi 2x^3 + 2x + (15/x) = 65.0
PS D:\Praktikum-Alpro\week 2> █
```

Gambar 2.2 Output Program Menghitung Fungsi

SOAL 3

Link Github: https://github.com/chinorjodie/71251187_chinor.git

Source Code:

```
# Input
gaji_per_jam = float(input("Masukkan gaji per jam: "))
jam_per_minggu = float(input("Masukkan jumlah jam kerja per minggu: "))

# Proses
minggu = 5

# 1. Pendapatan sebelum pajak
pendapatan_kotor = gaji_per_jam * jam_per_minggu * minggu

# 2. Pajak 14%
pajak = 0.14 * pendapatan_kotor
pendapatan_bersih = pendapatan_kotor - pajak

# 3. Pengeluaran setelah pajak
baju_aksesoris = 0.10 * pendapatan_bersih
alat_tulis = 0.01 * pendapatan_bersih

# Sisa setelah belanja
sisa_uang = pendapatan_bersih - baju_aksesoris - alat_tulis

# 4. Sedekah 25% dari sisa
sedekah = 0.25 * sisa_uang
```

```

# 5. Pembagian sedekah
anak_yatim = 0.30 * sedekah
kaum_dhuaafa = 0.70 * sedekah

# Output
print("\n--- HASIL PERHITUNGAN ---")
print("1. Pendapatan sebelum pajak: Rp", pendapatan_kotor)
print("2. Pendapatan setelah pajak: Rp", pendapatan_bersih)
print("3. Uang untuk baju & aksesoris: Rp", baju_aksesoris)
print("4. Uang untuk alat tulis: Rp", alat_tulis)
print("5. Jumlah uang yang disedekahkan: Rp", sedekah)
print("6. Untuk anak yatim: Rp", anak_yatim)
print("7. Untuk kaum dhuafa: Rp", kaum_dhuaafa)

```

Penjelasan:

Program tersebut digunakan untuk menghitung penghasilan Budi selama bekereja dalam lima minggu serta pembagian pengeluarannya. Pada bagian input, program meminta pengguna memasukkan gaji per jam dan jumlah jam kerja per minggu. Karena data dari **input()** berbentuk string, maka digunakan **float()** agar nilainya dapat dihitung sebagai angka desimal.

Pada proses, pertama-tama ditentukan bahwa lama berkeja adalah 5 minggu. Pendapatan kotor dihitung dengan mengalikan gaji per jam, jumlah jam kerja per minggu, dan jumlah minggu kerja. Setelah itu, pajak sebesar 14% dari pendapatan kotor dihitung, lalu dikurangkan untuk mendapatkan pendapatan bersih. Dari pendapat bersih tersebut, Budi mengalokasikan 10% untuk membeli baju dan aksesoris serta 1% untuk membeli alat tulis. Sisa uang setelah pengeluaran tersebut kemudian dihitung, dan 25% dari sisa tersebut digunakan untuk sedekah. Uang sedekah tersebut dibagi lagi, yaitu 30% diberikan kepada anak yatim dan 70% kepada kaum dhuafa.

Pada bagian output, program menampilkan seluruh hasil perhitungan secara terstruktur, mulai dari pendapat sebelum pajak, pendapatan setelah pajak, jumlah pengeluaran untuk baju dan alat tulis, jumlah sedekah, hingga rincian pembagian sedekah kepada anak yatim dan kaum dhuafa.

Output:

```
PS D:\Praktikum-Alpro\week 2> python .\soal3.py
Masukkan gaji per jam: 25000
Masukkan jumlah jam kerja per minggu: 40

--- HASIL PERHITUNGAN ---
1. Pendapatan sebelum pajak: Rp 5000000.0
2. Pendapatan setelah pajak: Rp 4300000.0
3. Uang untuk baju & aksesoris: Rp 430000.0
4. Uang untuk alat tulis: Rp 43000.0
5. Jumlah uang yang disedekahkan: Rp 956750.0
6. Untuk anak yatim: Rp 287025.0
7. Untuk kaum dhuafa: Rp 669725.0
PS D:\Praktikum-Alpro\week 2> █
```

Gambar 2.3 Output Program Penghasilan Budi