



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71220876
Nama Lengkap	Devina Elisse Putri
Minggu ke / Materi	02 / Variable, expression, statements

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.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2024

BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

MATERI 1

Value dalam program adalah informasi yang direpresentasikan oleh huruf, angka, atau karakter lainnya. Value dapat berupa integer seperti 1 atau 2, karakter tunggal seperti 'a' atau 'z', dan string seperti "Hello World". Value memiliki beberapa tipe yang berbeda, seperti tipe integer untuk nilai numerik dan tipe string untuk urutan karakter. Interpreter dapat mengidentifikasi string karena ditutup dalam tanda kutip.

Perintah print dapat menampilkan nilai selain string, seperti integer, float, karakter, atau boolean. Kita dapat mencoba fungsinya dengan menjalankan interpreter Python. Setiap nilai memiliki tipe yang bisa diketahui dengan menggunakan fungsi built-in type() untuk melakukan pemeriksaan tipe data.

Dalam beberapa model penulisan, tanda koma (,) digunakan di antara kelompok tiga digit untuk bilangan bulat besar, seperti 1.000.000. Namun, dalam Python, penulisan tersebut akan dianggap sebagai beberapa argumen terpisah dalam fungsi print(), bukan sebagai satu bilangan bulat.

MATERI 2

Salah satu fitur kuat dalam pemrograman adalah kemampuan untuk memanipulasi variabel. Variabel adalah lokasi memori yang digunakan untuk menyimpan nilai-nilai. Saat Anda membuat variabel, Anda mengalokasikan ruang di memori. Variabel menyimpan data yang dapat diubah selama program dieksekusi, yang mana isi variabel tersebut dapat dimodifikasi oleh operasi pada program yang menggunakan variabel.

Variabel dalam Python dapat menampung berbagai jenis data dan memiliki sifat dinamis, yang berarti tidak perlu dideklarasikan tipe data tertentu dan dapat diubah nilainya saat program berjalan. Potongan kode di atas memberikan contoh penggunaan variabel, seperti variabel "pesan" yang berisi string, variabel "n" yang berisi nilai integer 17, dan variabel "pi" yang mewakili nilai π . Untuk menampilkan nilai variabel, perintah print dapat digunakan.

MATERI 3

Pemberian nama pada variabel harus mengikuti panduan berikut ini:

1. Nama variabel boleh dimulai dengan huruf atau garis bawah (_), seperti: nama, _nama, namaKu, nama_variable.
2. Karakter berikutnya bisa berupa huruf, garis bawah (_), atau angka, seperti: _nama, n2, nilai1.

3. Sensitif terhadap kasus (case-sensitive), artinya huruf besar dan kecil berbeda, misalnya: variabel_Ku dan variabel_ku adalah variabel yang berbeda.
4. Hindari menggunakan kata kunci yang sudah ada dalam Python seperti if, while, for, dll.m

and	del	from	None	True
as	elif	global	nonlocaly	try
assert	else	if	not	while
break	except	import	or	width
class	False	in	pass	yield
continue	finally	is	raise	async
def	for	lamda	return	wait

```
> 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)
```

```
Agung Sejagat
20
dua puluh satu
Biodata
Joko Widodo
22
Berenang
```

MATERI 4

Statements dalam Python adalah bagian kode yang dapat dijalankan oleh interpreter. Contohnya adalah pernyataan print yang dapat berisi ekspresi dan penugasan. Saat menggunakan Python secara interaktif, interpreter langsung mengeksekusi pernyataan dan menampilkan hasilnya. Namun, dalam mode skrip, kode biasanya terdiri dari pernyataan yang berhubungan secara sekuensial.

MATERI 5

Operator adalah simbol untuk operasi aritmatika/logika, operand adalah nilai yang dioperasikan. Contohnya, dalam $2 + 3$, $+$ adalah operator penjumlahan dan 2 serta 3 adalah operand. Dalam Python, operator aritmatika digunakan untuk operasi matematika dasar.

Operator	Nama dan Fungsi	Contoh
$+$	Penjumlahan, menjumlahkan 2 buah operand	$x + y$
$-$	Pengurangan, mengurangi 2 buah operand	$x - y$
$*$	Perkalian, mengalikan 2 buah operand	$x * y$
$/$	Pembagian, membagi 2 buah operand	x / y
$**$	Pemangkatan, memangkatkan bilangan	$x ** y$

MATERI 6

Expression adalah representasi nilai yang terdiri dari values, variable, dan operator. Baik values maupun variable dapat dianggap sebagai expression, dan secara umum, gabungan ini dapat disebut sebagai expression.

MATERI 7

Urutan operasi dalam Python mengikuti aturan PEMDAS adalah Parantheses (Tanda kurung), Exponentiation (Eksponensial), Multiplication and Division (Perkalian dan Pembagian), lalu Operator. Tanda kurung memiliki prioritas tertinggi dan digunakan untuk mengatur urutan ekspresi. Kemudian, eksponensial, perkalian, dan pembagian memiliki prioritas yang sama, diikuti oleh penjumlahan dan pengurangan. Operator dibaca dari kiri ke kanan. Misalnya, dalam $5-3-1$, operasi pengurangan $5-3$ dilakukan terlebih dahulu, lalu hasilnya dikurangi dengan 1.

MATERI 8

Operator Modulus dalam Python memberikan sisa hasil bagi dari bilangan pertama dengan bilangan kedua, hanya berlaku untuk tipe data integer, dan dilambangkan dengan tanda persen (%).

```
>>> quotient = 7 // 3
>>> print(quotient)
2
>>> oprmomulus = 7 % 3
>>> print(oprmomulus)
```

Operator modulus digunakan untuk memeriksa pembagian, misalnya $x \% y == 0$ menunjukkan x dapat dibagi oleh y , dan untuk mengambil digit paling kanan dari suatu angka, contohnya $x \% 10$ menghasilkan digit paling kanan dalam basis 10, serta $x \% 100$ menghasilkan dua digit terakhir.

```
>>> first = 10
>>> second = 15
>>> print(first+second)
25
>>> first = '100'
>>> second = '150'
>>> print(first + second)
100150
```

Operator `*` juga bekerja dengan string dengan melakukan perkalian antara content string dan integer.

```
>>> first = 'Test '
>>> second = 3
>>> print(first * second)
Test Test Test
```

MATERI 9

Program melibatkan tiga tahap utama: Input, Proses, dan Output. Input adalah data yang dibutuhkan program, Proses adalah langkah-langkah untuk memecahkan masalah, dan Output adalah hasil dari proses tersebut. Misalnya, pengambilan uang melalui ATM dapat dipahami sebagai aliran Input-Proses-Output.

Python dapat menangani input dari pengguna dengan fungsi bawaannya, `input()`. Ketika dipanggil, program akan berhenti dan menunggu pengguna untuk memasukkan input teks dari keyboard. Setelah pengguna menekan tombol Enter, program akan melanjutkan eksekusi dan `input()` akan mengembalikan apa yang diketik pengguna sebagai string.

MATERI 10

Tanda pagar (#) di Python digunakan untuk menandai komentar. Komentar ini tidak diproses oleh interpreter Python dan hanya berguna untuk membantu programmer memahami maksud dari kode.

```
# Ini komentar  
# Ini juga adalah komentar  
# Ini juga masih komentar
```

MATERI 11

Contoh input/output tipe data string

```
Nama = input("Nama Anda : ")  
Alamat = input("Alamat Tinggal Anda : " )  
Umur = input("Umur Anda : ")  
TL = input("Tempat Lahir Anda : ")  
Tgl = input("Tanggal Lahir Anda : ")  
IPK = input("IPK Anda : ")  
print("=====  
print("DATA AKADEMIK")  
print("Nama Anda : ", Nama)  
print("Alamat Tinggal Anda : ",Alamat)  
print("Umur Anda : ", Umur)  
print("Tempat Lahir Anda : ", TL)  
print("Tanggal Lahir Anda : ", Tgl)  
print("IPK Anda : ", IPK)  
  
Nama Anda : gebi  
Alamat Tinggal Anda : sarkem  
Umur Anda : 25  
Tempat Lahir Anda : suarkem  
Tanggal Lahir Anda : 69  
IPK Anda : 0,0  
=====  
DATA AKADEMIK  
Nama Anda : gebi  
Alamat Tinggal Anda : sarkem  
Umur Anda : 25  
Tempat Lahir Anda : suarkem  
Tanggal Lahir Anda : 69  
IPK Anda : 0,0
```

Berikut adalah contoh menggunakan tipe data bilangan baik integer (bilangan bulat) maupun float (bilangan berkoma).

```
# CONTOH PROGRAM
# MENGHITUNG BILANGAN
#=====

x1 = eval(input("X1 = "))
x2 = eval(input("X2 = "))
x3 = eval(input("X3 = "))
x4 = eval(input("X4 = "))
jumlah = x1+x2+x3+x4
kali = x1*x2*x3*x4
print('Hasil Penjumlahan semua bilangan = ', jumlah)
print('Hasil Perkalian semua bilangan = ', kali)
jumlah = jumlah + 0.5
print('Jika ditambah 0.5 hasilnya = ', jumlah)
kali = kali * 0.5
print('Jika dikali 0.5 hasilnya = ', kali)

X1 = 69
X2 = 77
X3 = 17
X4 = 99
Hasil Penjumlahan semua bilangan = 262
Hasil Perkalian semua bilangan = 8941779
Jika ditambah 0.5 hasilnya = 262.5
Jika dikali 0.5 hasilnya = 4470889.5
```

MATERI 12

Inisiasi variabel dan konstanta untuk masalah penjumlahan total harga pembelian beberapa barang.

- Langkah 1: Inisiasi Variabel input: kode_barang, nama_barang, harga_satuan_barang, jumlah_per_barang_beli, total_harga_per_transaksi = 0 Proses: harga_beli_per_barang = harga_satuan_barang * jumlah_per_barang_beli
total_harga_per_transaksi=harga_beli_per_barang + total_harga_per_transaksi Output: total_harga_per_transaksi.
- Langkah 2: Tipe Data: kode_barang dan nama_barang string, jumlah_barang integer, harga_satuan, harga_beli, total_harga_barang float.\
- kode program :

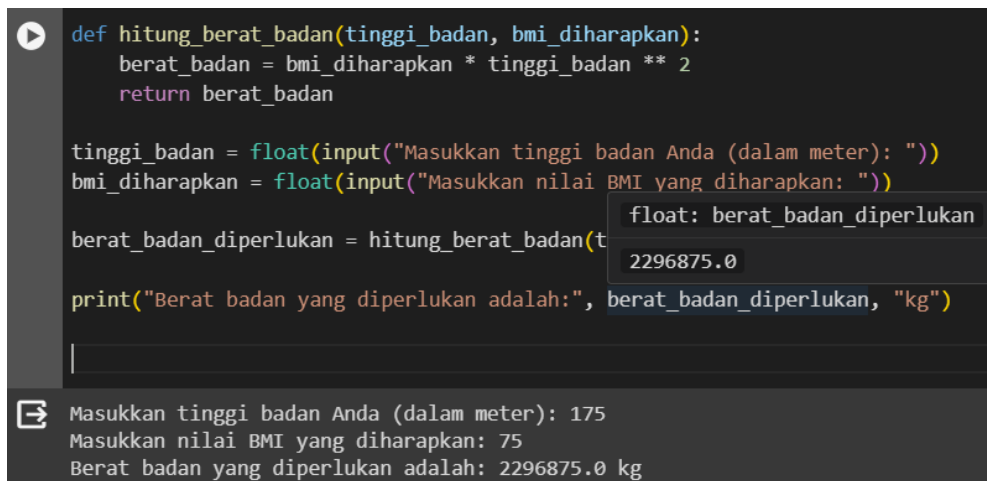
```
total_hrg_brg = 0.0
kd_brg = input("Kode barang = ")
nama_brg = input("Nama barang = ")
harga_satuan = eval(input("Harga satuan barang =Rp. "))
jum_brg = eval(input("Jumlah barang yang dibeli = "))
harga_beli = harga_satuan * jum_brg
total_hrg_brg = harga_beli + total_hrg_brg
print("Total harga yang dibayar Rp",total_hrg_brg)

Kode barang = 11111
Nama barang = tas rembo
Harga satuan barang =Rp. 20000000000
Jumlah barang yang dibeli = 50
Total harga yang dibayar Rp 100000000000.0
```

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



```
def hitung_berat_badan(tinggi_badan, bmi_diharapkan):  
    berat_badan = bmi_diharapkan * tinggi_badan ** 2  
    return berat_badan  
  
tinggi_badan = float(input("Masukkan tinggi badan Anda (dalam meter): "))  
bmi_diharapkan = float(input("Masukkan nilai BMI yang diharapkan: "))  
  
berat_badan_diperlukan = hitung_berat_badan(tinggi_badan, bmi_diharapkan)  
  
print("Berat badan yang diperlukan adalah:", berat_badan_diperlukan, "kg")  
|  
  
Masukkan tinggi badan Anda (dalam meter): 175  
Masukkan nilai BMI yang diharapkan: 75  
Berat badan yang diperlukan adalah: 2296875.0 kg
```

1. `def hitung_berat_badan(tinggi_badan, bmi_diharapkan):`: Ini adalah deklarasi fungsi `hitung_berat_badan` yang menerima dua parameter, yaitu `tinggi_badan` (tinggi badan dalam meter) dan `bmi_diharapkan` (nilai BMI yang diharapkan).
2. `berat_badan = bmi_diharapkan * tinggi_badan ** 2`: Di dalam fungsi `hitung_berat_badan`, berat badan dihitung dengan rumus BMI, yaitu `bmi_diharapkan * tinggi_badan ** 2`.
3. `return berat_badan`: Fungsi mengembalikan nilai `berat_badan` yang dihitung.
4. `tinggi_badan = float(input("Masukkan tinggi badan Anda (dalam meter): "))`: Program meminta pengguna untuk memasukkan tinggi badan dalam meter dan menyimpannya dalam variabel `tinggi_badan`.
5. `bmi_diharapkan = float(input("Masukkan nilai BMI yang diharapkan: "))`: Program juga meminta pengguna untuk memasukkan nilai BMI yang diharapkan dan menyimpannya dalam variabel `bmi_diharapkan`.
6. `berat_badan_diperlukan = hitung_berat_badan(tinggi_badan, bmi_diharapkan)`: Variabel `berat_badan_diperlukan` diinisialisasi dengan memanggil fungsi `hitung_berat_badan` dengan parameter `tinggi_badan` dan `bmi_diharapkan`.
7. `print("Berat badan yang diperlukan adalah:", berat_badan_diperlukan, "kg")`: Program mencetak hasil `berat_badan_diperlukan` dalam kilogram.

SOAL 2

```
def hitung_fungsi(x):
    hasil = (2 * x ** 3 + 2 * x + 15) / x
    return hasil

x = int(input("Masukkan bilangan bulat x: "))

hasil_fungsi = hitung_fungsi(x)

print("Hasil dari fungsi f(x) adalah:", hasil_fungsi)
```

Masukkan bilangan bulat x: 7
Hasil dari fungsi f(x) adalah: 102.14285714285714

1. `def hitung_fungsi(x):`: Mendefinisikan fungsi `hitung_fungsi` yang menerima parameter `x`.
2. Di dalam fungsi `hitung_fungsi`, menghitung nilai fungsi $f(x)$ menggunakan rumus yang diberikan: $\frac{2x^3 + 2x + 15}{x}$.
3. `return hasil`: Mengembalikan nilai hasil perhitungan fungsi.
4. Program meminta pengguna untuk memasukkan bilangan bulat x dan menyimpannya dalam variabel `x`.
5. Memanggil fungsi `hitung_fungsi` dengan parameter `x` untuk menghitung nilai fungsi $f(x)$ berdasarkan input pengguna.
6. Mencetak hasil nilai fungsi $f(x)$.

SOAL 3

```
f hitung_pendapatan(gaji_per_jam, jam_kerja_per_minggu):
    pendapatan_sebelum_pajak = gaji_per_jam * jam_kerja_per_minggu * 5
    pajak = pendapatan_sebelum_pajak * 0.14
    pendapatan_setelah_pajak = pendapatan_sebelum_pajak - pajak

    belanja_pakaian_aksesoris = pendapatan_setelah_pajak * 0.10
    belanja_alat_tulis = pendapatan_setelah_pajak * 0.01
    sedekah = (pendapatan_setelah_pajak - belanja_pakaian_aksesoris - belanja_alat_tulis) * 0.25
    sedekah_anak_yatim = sedekah * 0.30
    sedekah_kaum_dhuafa = sedekah * 0.70

    return pendapatan_sebelum_pajak, pendapatan_setelah_pajak, belanja_pakaian_aksesoris, belanja_alat_tulis, sedekah, sedekah_anak_yatim, sedekah_kaum_dhuafa

ji_per_jam = float(input("Masukkan gaji per jam yang diinginkan: "))
m_kerja_per_minggu = float(input("Masukkan jumlah jam kerja per minggu: "))

pendapatan_sebelum_pajak, pendapatan_setelah_pajak, belanja_pakaian_aksesoris, belanja_alat_tulis, sedekah, sedekah_anak_yatim, sedekah_kaum_dhuafa = hitung_pendapatan(gaji_per_jam, jam_kerja_per_minggu)

int("1. Pendapatan Budi sebelum pembayaran pajak:", pendapatan_sebelum_pajak)
int("2. Pendapatan Budi setelah pembayaran pajak:", pendapatan_setelah_pajak)
int("3. Jumlah uang yang akan Budi habiskan untuk membeli pakaian dan aksesoris:", belanja_pakaian_aksesoris)
int("4. Jumlah uang yang akan Budi habiskan untuk membeli alat tulis:", belanja_alat_tulis)
int("5. Jumlah uang yang akan Budi sedekahkan:", sedekah)
int("6. Jumlah uang yang akan diterima anak yatim:", sedekah_anak_yatim)
int("7. Jumlah uang yang akan diterima kaum dhuafa:", sedekah_kaum_dhuafa)
```

Source code tersebut adalah sebuah fungsi bernama `hitung_pendapatan` yang menerima dua parameter, yaitu `gaji_per_jam` (gaji per jam yang diinginkan) dan `jam_kerja_per_minggu` (jumlah jam kerja per minggu). Fungsi ini menghitung pendapatan Budi selama libur musim panas serta alokasi penggunaan uangnya.

Pada baris-baris selanjutnya, variabel-variabel seperti ``pendapatan_sebelum_pajak``, ``pendapatan_setelah_pajak``, ``belanja_pakaian_aksesoris``, ``belanja_alat_tulis``, ``sedekah``, ``sedekah_anak_yatim``, dan ``sedekah_kaum_dhuafa`` dihitung berdasarkan rumus-rumus yang diberikan dalam permasalahan.

Setelah itu, program meminta pengguna untuk memasukkan ``gaji_per_jam`` dan ``jam_kerja_per_minggu`` menggunakan fungsi ``input()`` dan menghitung pendapatan Budi dengan memanggil fungsi ``hitung_pendapatan()``.

Terakhir, program mencetak hasil-hasil perhitungan sesuai dengan format yang diminta dalam permasalahan.