|  |  |
| --- | --- |
| A picture containing text, clipart  Description automatically generated | **Laporan Praktikum Algoritma dan Pemrograman**  Semester Genap 2023/2024 |

|  |  |
| --- | --- |
| **NIM** | **71230992** |
| **Nama Lengkap** | **Andriano Kurniawan Ladjeba** |
| **Minggu ke / Materi** | **02 / Variable, Expression dan 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%)

## MATERI 1

**Values dan type**

Value merupakan komponen utama dari program, seperti huruf atau angka. Value yang sering kita kenal 1,2,’a’,’z’ dan “Hello word”. Value dibagi menjadi beberapa tipe yang berbeda, misalnya 2 untuk sebuah nilai integer dan “Hello World” untuk sebuah nilai string. Interpreter dapat melakukan indentifikasi terhadap string karena pada penulisannya ditutup menggunakan tanda petik (quation mark).  
 untuk mencoba memahami values dan type, silahkan coba beberapa baris kode menggunakan python interactive mode seperti yang ditunjukkan pada potongan kode dibawah ini.

**>>> print(7)**

**7**

**>>> print(17.54)**

**17.54**

**>>> print("lalilulelo")**

**lalilulelo**

**>>> print('True')**

**True**

**>>> print('False')**

**False**

Perintah print juga bekerja untuk mengeksekusi perintah yang sudah dimasukkan, selain untuk value string perintah print juga bekerja untuk value seperti interger (bilangan bulat), float (bilangan pecahan), Character (huruf), *bool* (benar/salah). Untuk mencobanya kita akan mencoba menggunakan tipe-tipe di atas dengan perintah Python yang menjalankan interpreter. Python juga menyediakan fungsi built-in untuk mengubah atau mendefinisikan tipe data pada value dengan menggunakan fungsi type()

**>>> x = 10  
>>> print(x, "tipenya adalah",type(x))  
10 tipenya adalah <class 'int'>  
>>> print(x, "tipenya adalah", type(x))  
10 tipenya adalah <class 'int'>  
>>> x = 5+5j  
>>> print(x, "tipenya adalah", type(x))  
(5+5j) tipenya adalah <class 'complex'>**

Ketika menggunakan bilangan bulat besar, beberapa model penulisan menggunakan tanda koma(,) diantara kelompok tiga digit. Misalnya pada penulisan 1.000.000. Dalam python, akan dianggap sebagai bulat.

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

Hal ini terjadi dikarenakan Python menganggap bahwa 1,000,000,000 merupakan kiriman parameter  
sebanyak 3 parameter pada fungsi print, yaitu 1,0,dan 0.

**Variabel**

Salah satu fitur powerfull dalam

**>>> pesan = 'Selamatt pagii, mari belajar python'**

**>>> n = 17**

**>>> pi = 3.1415926535897931**

Variabel dapat menyimpan berbagai jenis data. Dalam pemrograman Python, variabel memiliki sifat dinamis, artinya variablePython tidak perlu didekralasikan tipe data tertentu dan variable Python dapat diubah saat program dijalankan.

Potongan kode di atas adalah contoh penggunaan variabel. Contoh pertama adalah variabel pesan yang berisi string, contoh kedua adalah variabel n yang berisi nilai integer 17, dan contoh ketiga adalah nilai pi (π).  
Untuk menampilkan nilai suatu variabel, Anda dapat menggunakan perintah print .

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

**Nama Variabel dan Keywords**

Pemberian nama pada variabel mengacu pada panduan berikut ini.  
1. Nama variable boleh diawali menggunakan huruf atau garis bawah (\_),   
contoh: nama, \_nama,namaKu, nama\_variable.

2. Karakter selanjutnya dapat berupa huruf, garis bawah (\_) atau angka,   
contoh: \_nama, n2,nilai1.

3. Karakter pada nama variable bersifat sensitif (case-sensitif). Artinya huruf besar dan kecil dibedakan. Misalnya, variabel\_Ku dan variabel\_ku, keduanya adalah variabel yang berbeda.

4. Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam python seperti if, while, for, dsb.

Python sendiri memiliki 35 keyword yang tidak boleh digunakan untuk memberi nama variabel.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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** |

Berikut ini contoh penggunaan variable dalam bahasa pemrograman Python.  
#proses memasukan data kedalam variabel

Nama = “Agung Sejagat”  
print(nama)  
umur = 20  
print(umur)  
type(umur)  
umur =”dua puluh satu”  
print(umur)  
type(umur)  
namaDepan = “Joko”  
namaBelakang = “Widodo”  
nama = namaDepan + “ ” + namaBelakang  
umur = 22  
hobi “Berenang”  
print(“Biodata\n”,nama,”\n”, umur, “\n”,hobi)

#contoh variabel lainnya

Inivariabel = “Halo”  
Ini\_juga-variabel = “Hai”  
\_inivariabeljuga = “Hi”  
inivariabel1222 = “Bye”  
panjang = 10  
lebar = 5  
luas = panjang \* lebar  
print(luas)

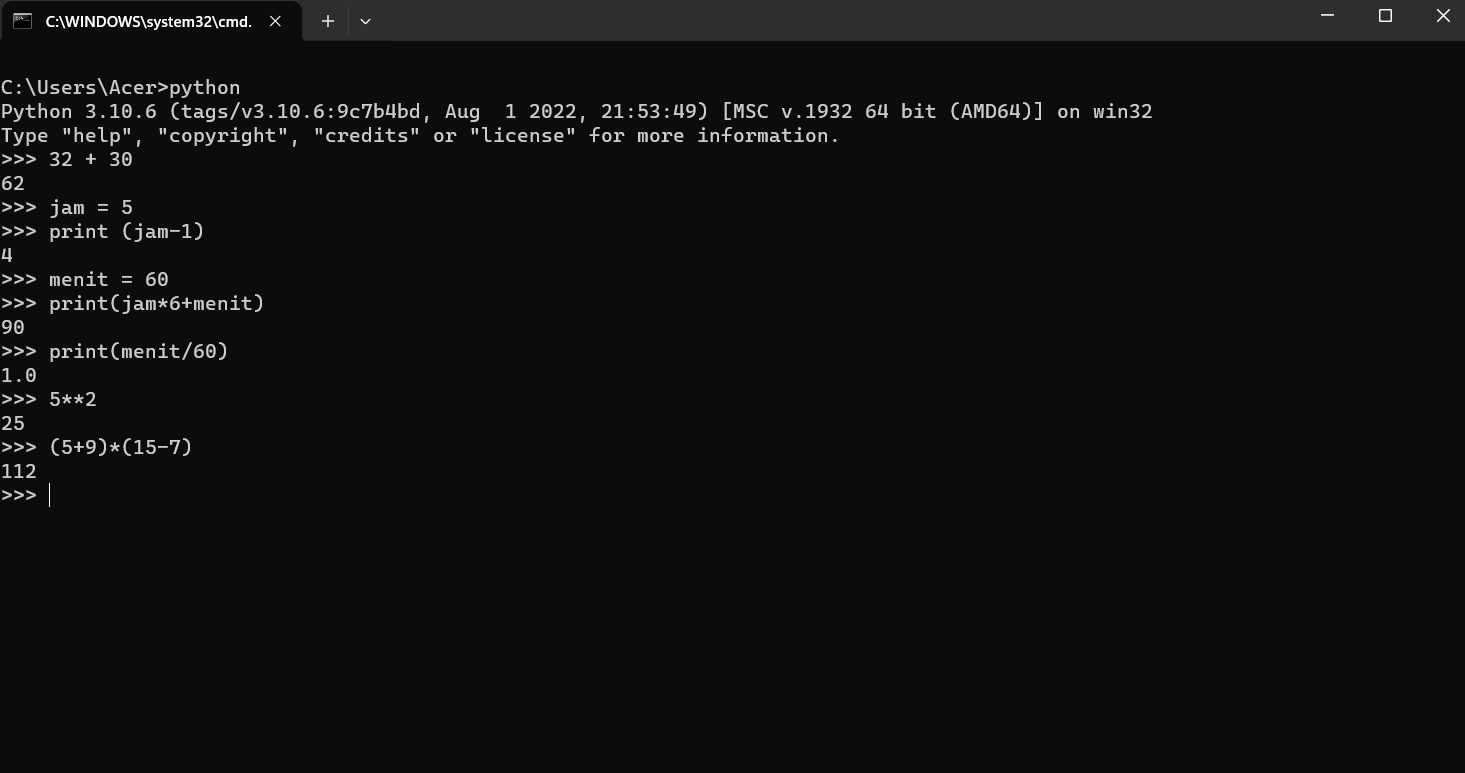
**Statements**Ini adalah suatu instruksi dari komputer untuk memberi tahu apa yang dilakukan. Statements juga bagian dari code interpreter Python yang dapat dieksekusi. Jika pada mode interaktif akan langsung memunculkan hasilnya. Berbeda halnya dengan script mode.   
Contoh Statements dan hasilnya :

## 

**Operator dan Operand**

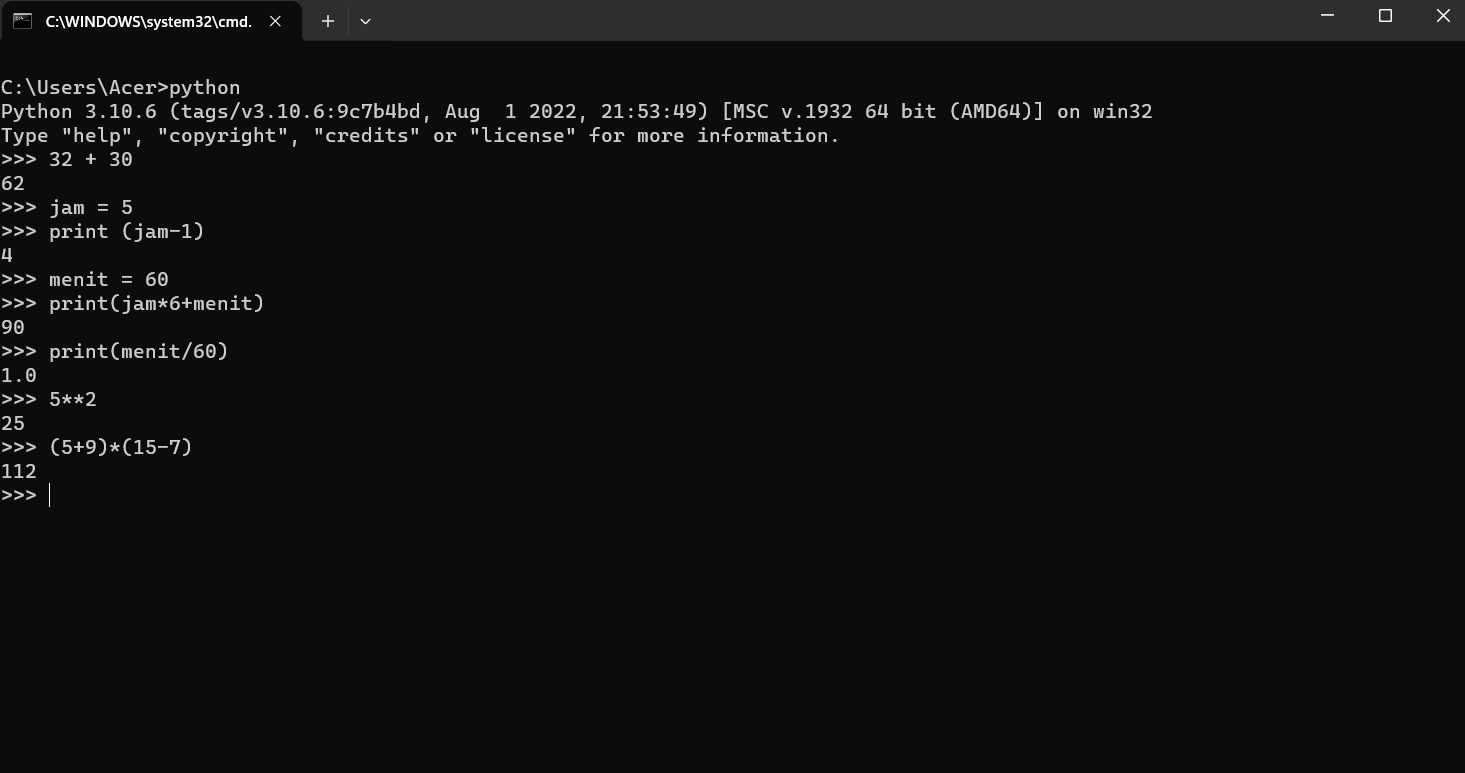
Operator adalah simbol tertentu yang digunakan untuk melakukan operasi aritmatika maupun  
logika. Nilai yang padanya dilakukan operasi disebut operand. Misalnya adalah 2 + 3. Di sini  
tanda + adalah operator penjumlahan. 2 dan 3 adalah operand.  
 Pada bagian ini secara khusus akan membahas operator aritmatika pada Python. Operator aritmatika adalah operator yang digunakan untuk melakukan operasi matematika, seperti penjumlahan,pengurangan, perkalian, pembagian, dan sebagainya. Tabel berikut menunjukkan jenis operator aritmatika.

|  |  |  |
| --- | --- | --- |
| **Operator** | **Nama dan Fungsi** | **Contoh** |
| + | Penjumlahan untuk 2 buah operand | x + y |
| - | Pengurangan untuk 2 buah operand | x - y |
| \* | Perkalian untuk 2 buah operand | x \* y |
| / | Pembagian untuk 2 buah operand | x / y |
| \*\* | Pangkat untuk pemangkatan bilangan | x \*\* y |

Beberapa contoh penggunaan operator aritmatika pada python dapat dilihat pada potongan kode berikut ini.

**Expressions**

Expression merupakan representasi dari nilai dan dapat terdiri dari gabungan antara values,  
variable dan operator. Values dengan sendirinya dapat dianggap sebagai expression dan juga  
variabel. Secara umum, semuanya dapat disebut dengan expression.

17  
x  
x + 17  
Ketika menggunakan expression dalam model intercative, interpreter akan melakukan evaluasi  
dan menampilkan hasilnya.

**Urutan Operasi**

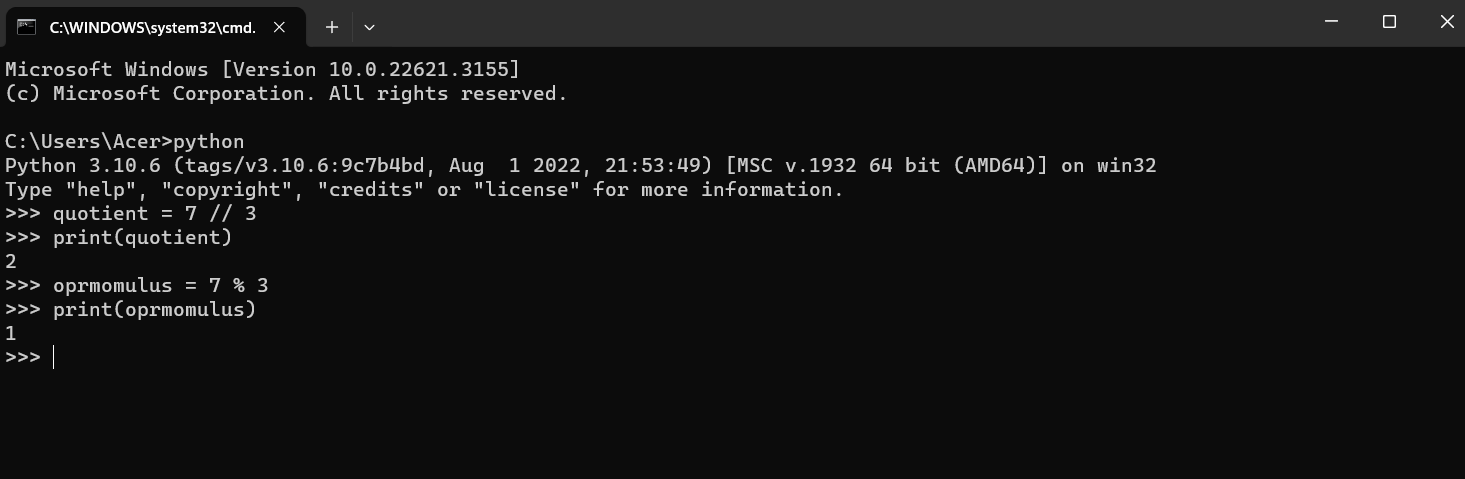
Urutan operasi berlaku ketika ekspresi berisi beberapa operator. Urutan operasi bergantung pada aturan prioritas. Untuk operasi matematika, Python mengikuti transformasi matematika.  
Urutan operasi sering disingkat dengan **PEMDAS - Parantheses, Exponentiation, Multiplication and Division, Operator**.

* **Paranthese** (Tanda kurung) - merupakan prioritas tertinggi dan digunakan untuk memaksa  
  expression dalam urutan yang sesuai. Contohnya 2\*(3-1) hasilnya 4, dan (1+1) \*\* (5-2)  
  hasilnya 8. Penggunaan tanda kurung dapat digunakan untuk membuat expression menjadi  
  lebih mudah untuk dibaca, misalny (minute \* 100) / 60.
* **Exponentiation** (Eksponensial/Pemangkatan) - merupakan prioritas tertinggi berikutnya,  
  contoh 2\*\*1+1 hasilnya 3, bukan 4, dan 3\*1\*\*3 hasilnya 3 bukan 27
* **Multiplication and Divison** (Perkalian dan Pembagian) - memiliki prioritas yang sama tetapi  
  lebih tinggi dari penjumlahan dan pengurangan. Penjumlahan dan pengurangan juga memiliki  
  prioritas yang sama pula. Contoh 2\*3-1 hasilnya 5 bukan 4, dan 6+4/2 hasilnya 8, bukan 5.
* **Operators** - operator memiliki prioritas yang sama, dibaca dari kiri ke kanan. Contoh 5-3-1  
  hasilnya 1 bukan 3 karena operasi pengurangan 5-3 terlebih dahulu baru kemudian hasilnya  
  dikurangi dengan 1.

Jika terjadi keraguan, silakan letakkan tanda kurung di dalam ekspresi untuk memastikan bahwa komposisinya sesuai dengan yang diinginkan.

**Operator Modulus dan String**

**Modulus**Operator Modulus merupakan sisa hasil bagi dari bilangan pertama dengan bilangan kedua.  
Operator ini hanya berlaku pada tipe data integer. Dalam python, operator modulus dilambangakan  
dengan tanda persen (%).

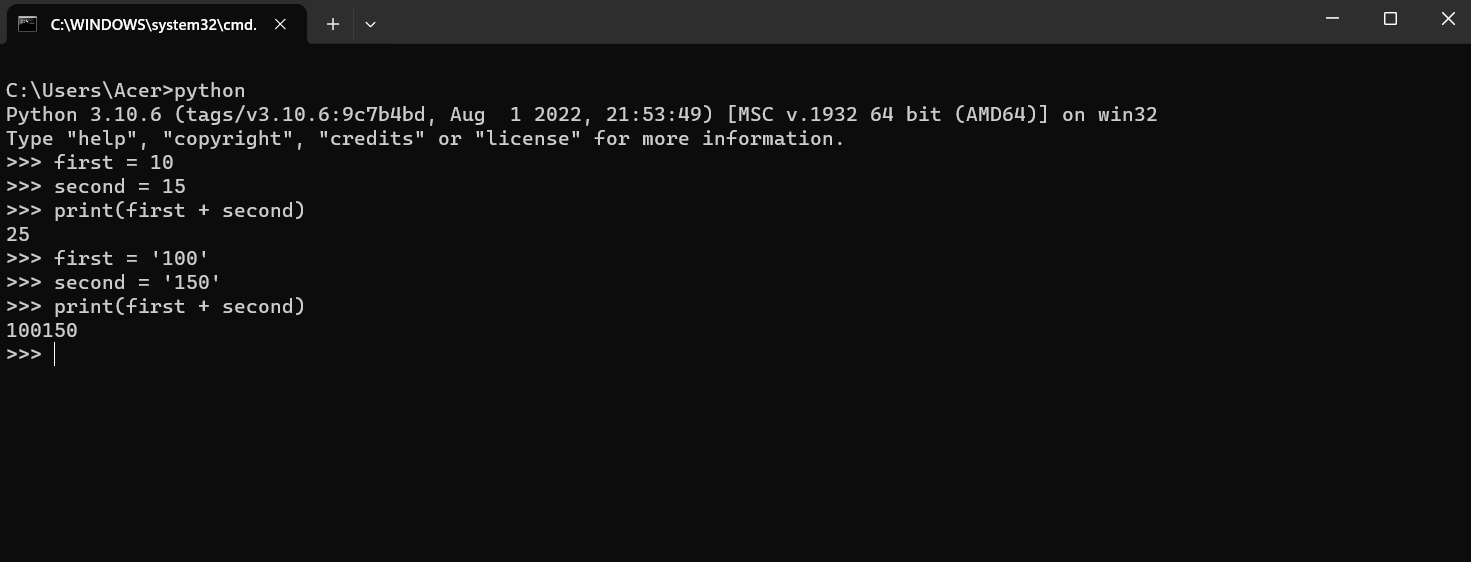


7 dibagi dengan 3 menghasilkan 2 dnegan sisa hasil bagi 1.

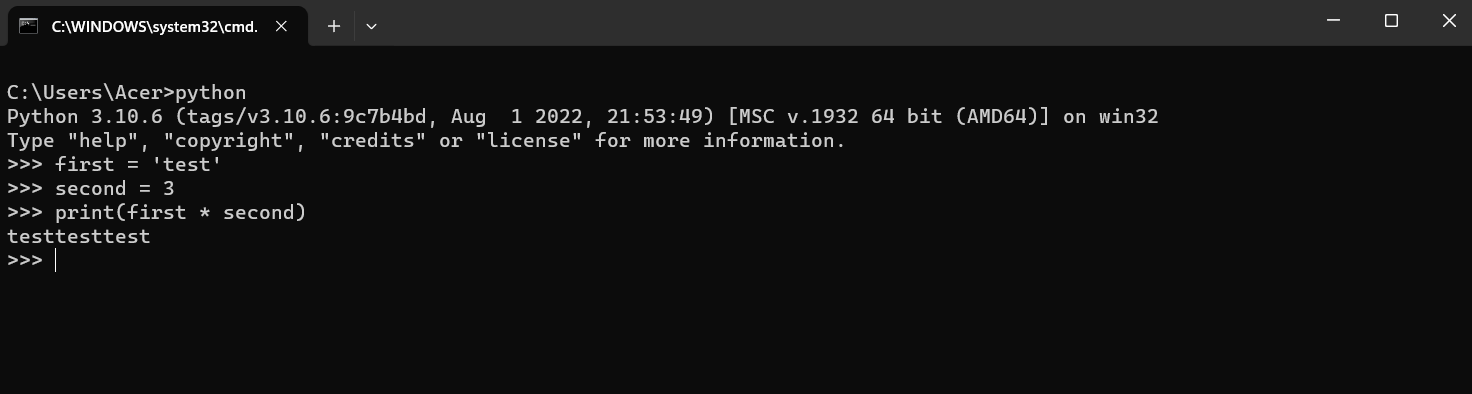
Contoh penggunaan operator modulus.

* Memeriksa satu angka dapat dibagi dengan yang lain, misal jika x % y adalah 0, maka x  
  dapat dibagi oleh y.
* Dapat mengekstrak digit paling kanan atau digit dari suatu angka. Misalnya, x% 10 menghasilkan digit x paling kanan (dalam basis 10). Demikian pula, x% 100 menghasilkan dua digit terakhir.

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



Operator \* juga bekerja dengan string dengan melakukan perkalian antara content string dan interger.

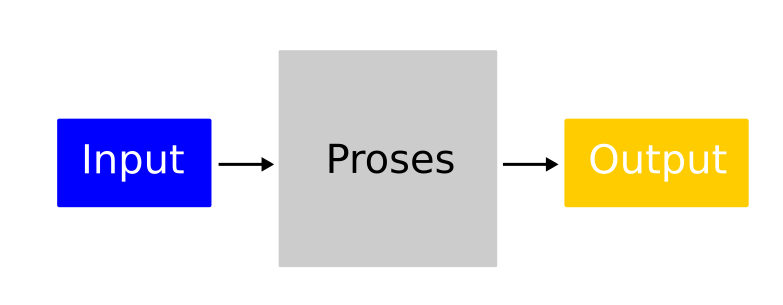


**Menangani Input dari Pengguna**

Sebuah program biasanya memiliki alur kerja Input - Proses - Output yang alurnya ditunjukkan

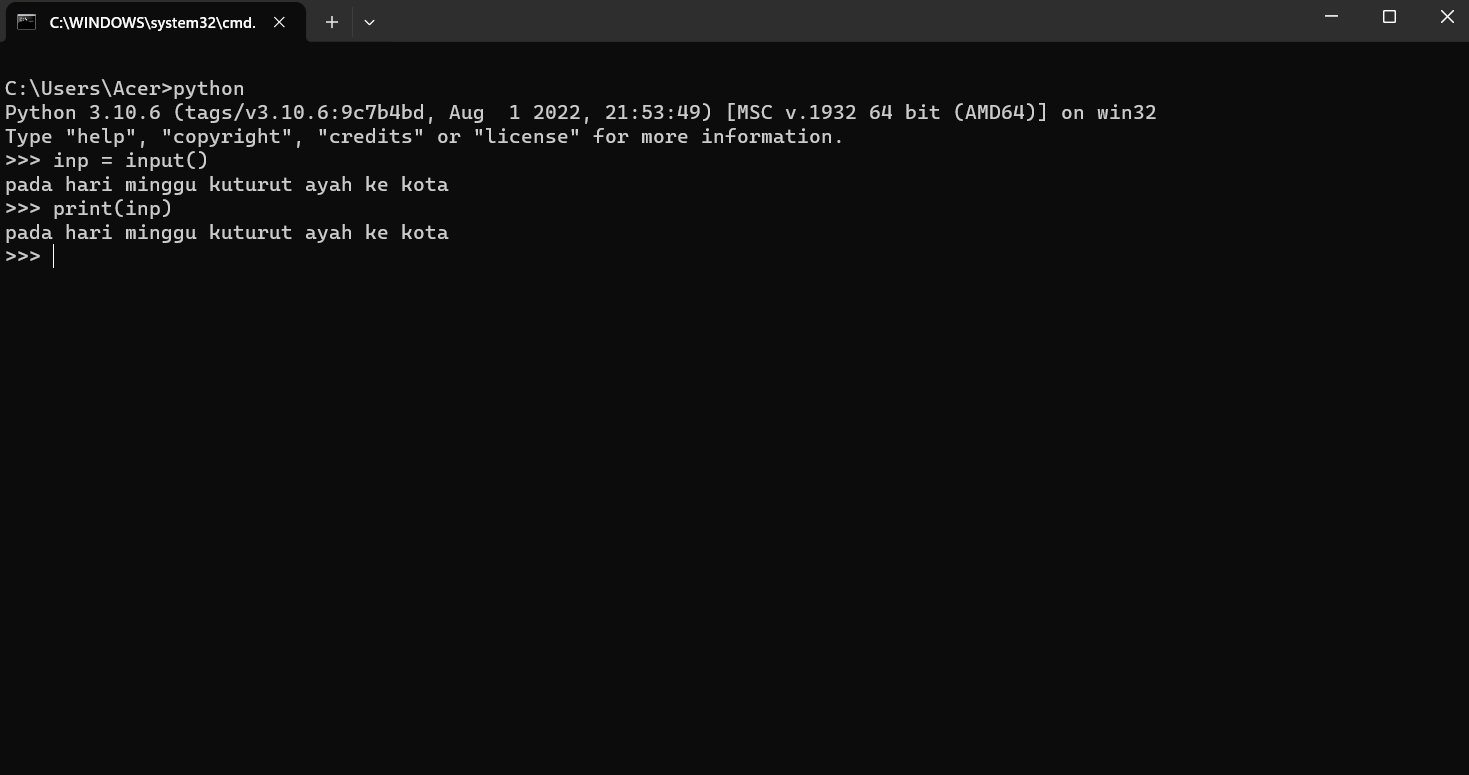
pada Gambar dibawah.

Input adalah data/masukan yang dibutuhkan supaya program bisa berjalan. Proses adalah  
langkah-langkah yang dilakukan oleh program untuk memecahkan masalah. Sedangkan Output  
adalah hasil yang didapatkan setelah menjalankan langkah-langkah tersebut. Sebagai contoh  
misalnya mengambil uang lewat ATM. Pengambilan uang melalui ATM dapat dibagi menjadi 3  
bagian (Input-Proses-Output) tersebut, yaitu:

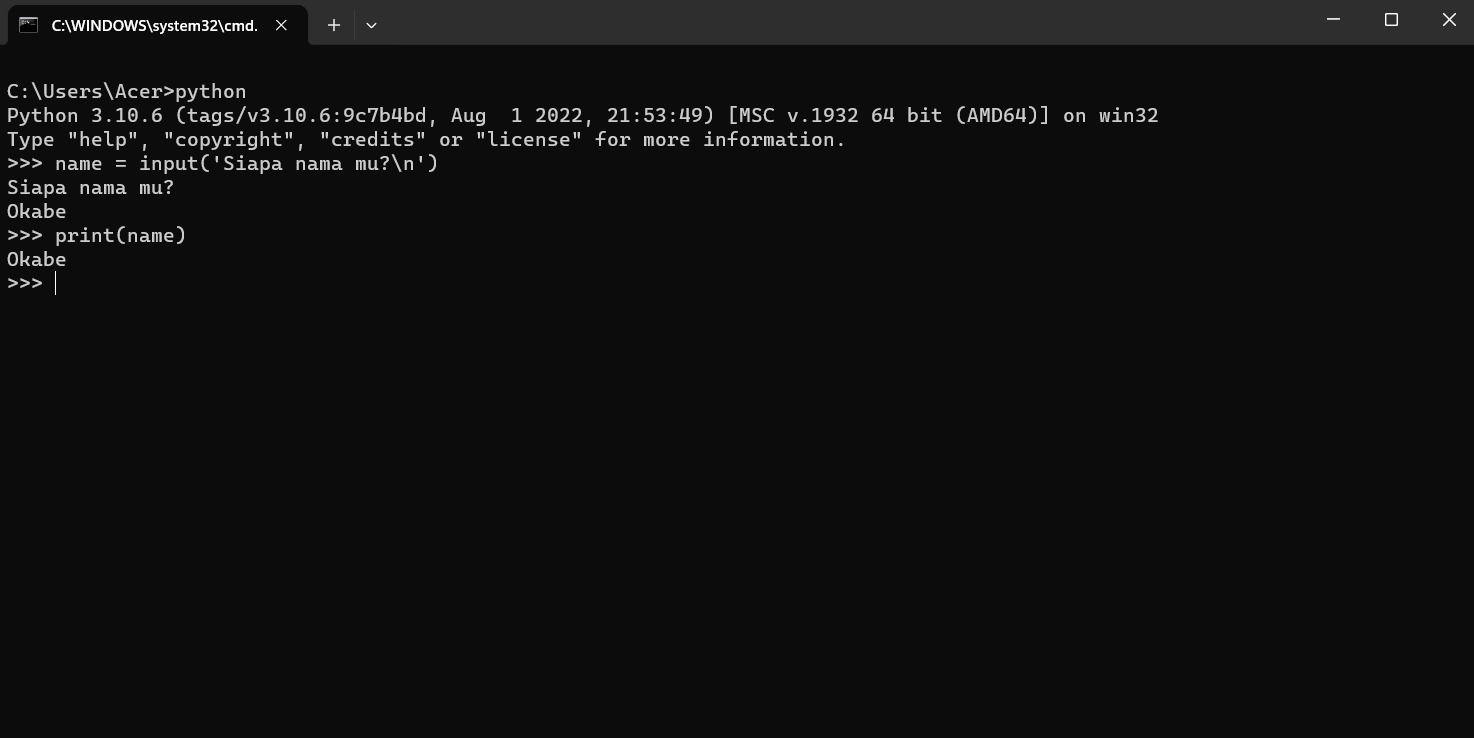


1. Masukkan kartu ATM. Masukkan PIN anda. Kartu ATM dan PIN adalah Input yang diperlukan supaya anda bisa mengambil uang.
2. Anda memilih menu Pengambilan Uang. Kemudian anda memasukkan nominal yang diinginkan. Bagian ini juga merupakan Input.
3. Mesin ATM akan memproses transaksi anda dengan menghubungi server Bank yang bersangkutan. Dilakukan berbagai macam pengecekan (misal: apakah saldonya cukup? apakah kartunya masih berlaku? apakah ada blokir?). Bagian ini disebut Proses.
4. Mesin ATM mengeluarkan uang, bukti pengambilan uang dan kartu anda. Uang yang keluar merupakan Output dari kegiatan ini. Selain itu, saldo anda juga berkurang. Pengurangan saldo juga merupakan hasil dari kegiatan ini.

Python juga dapat memproses masukan pengguna. Input dalam hal ini adalah input teks yang dimasukkan oleh pengguna. Oleh karena itu, Python menyediakan fungsi bawaan yang disebut "input" untuk mendapatkan masukan dari keyboard. Ketika fungsi ini dipanggil, program akan berhenti dan menunggu pengguna mengetikkan sesuatu. Ketika pengguna menekan tombol Enter, program melanjutkan dan mengembalikan apa yang diketik pengguna sebagai string.

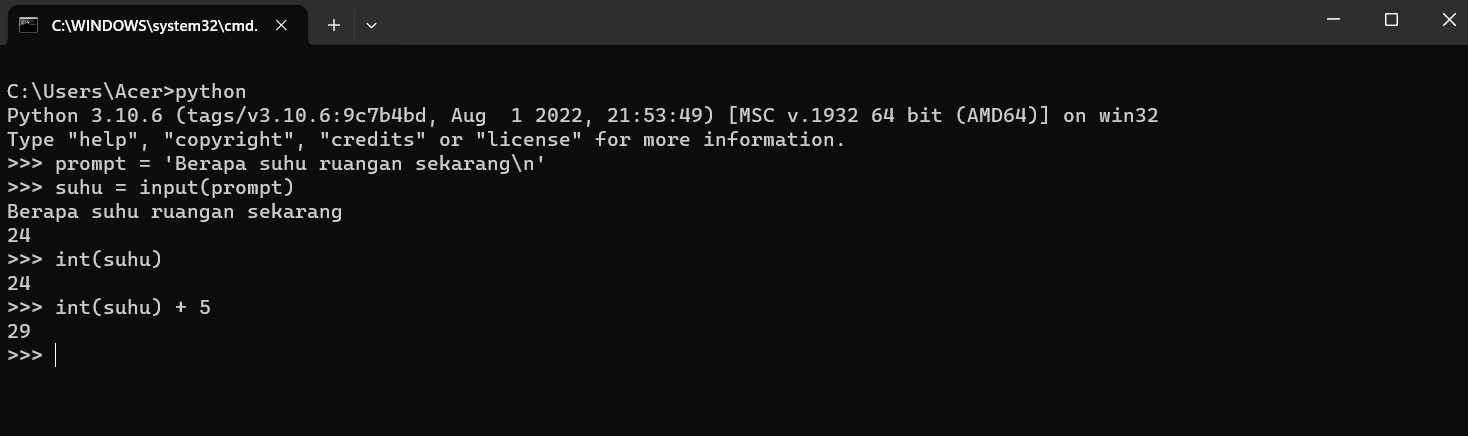


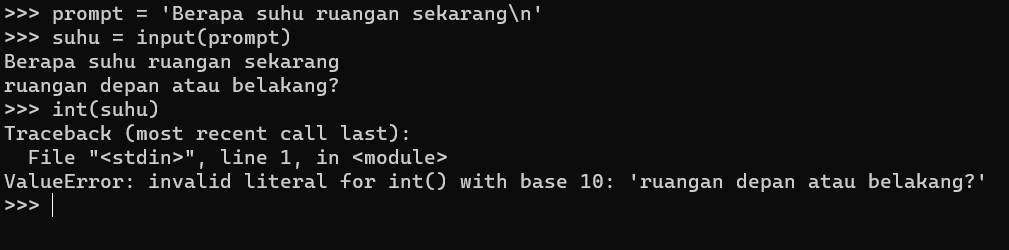
Sebelum mendapatkan input dari pengguna, lebih baik untuk mencetak prompt yang memberitahu pengguna apa yang harus diinput. String tersebut dapat diteruskan ke input untuk ditampilkan  
kepada pengguna sebelum berhenti untuk input.



Tanda \n pada akhir perintah mewakili baris baru atau ganti baris sehingga hasil input pengguna muncul dibawah perintah.

Ketika mengharapkan pengguna untuk mengetik bilangan bulat, dapat dilakukan dengan mengonversi nilai kembali ke int menggunakan fungsi int():



Akan menjadi error jika pengguna memasukkan data selain angka.  


**Komentar**

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

# komentar pertama

print("hai dunia!") # komentar kedua

Outputnya  
  
**hai dunia!**  
Python tidak memiliki fitur komentar multibaris. Kita harus mengomentari satu persatu baris  
seperti berikut:

# Ini komentar

# Ini juga adalah komentar

# Ini juga masih komentar

## MATERI 2

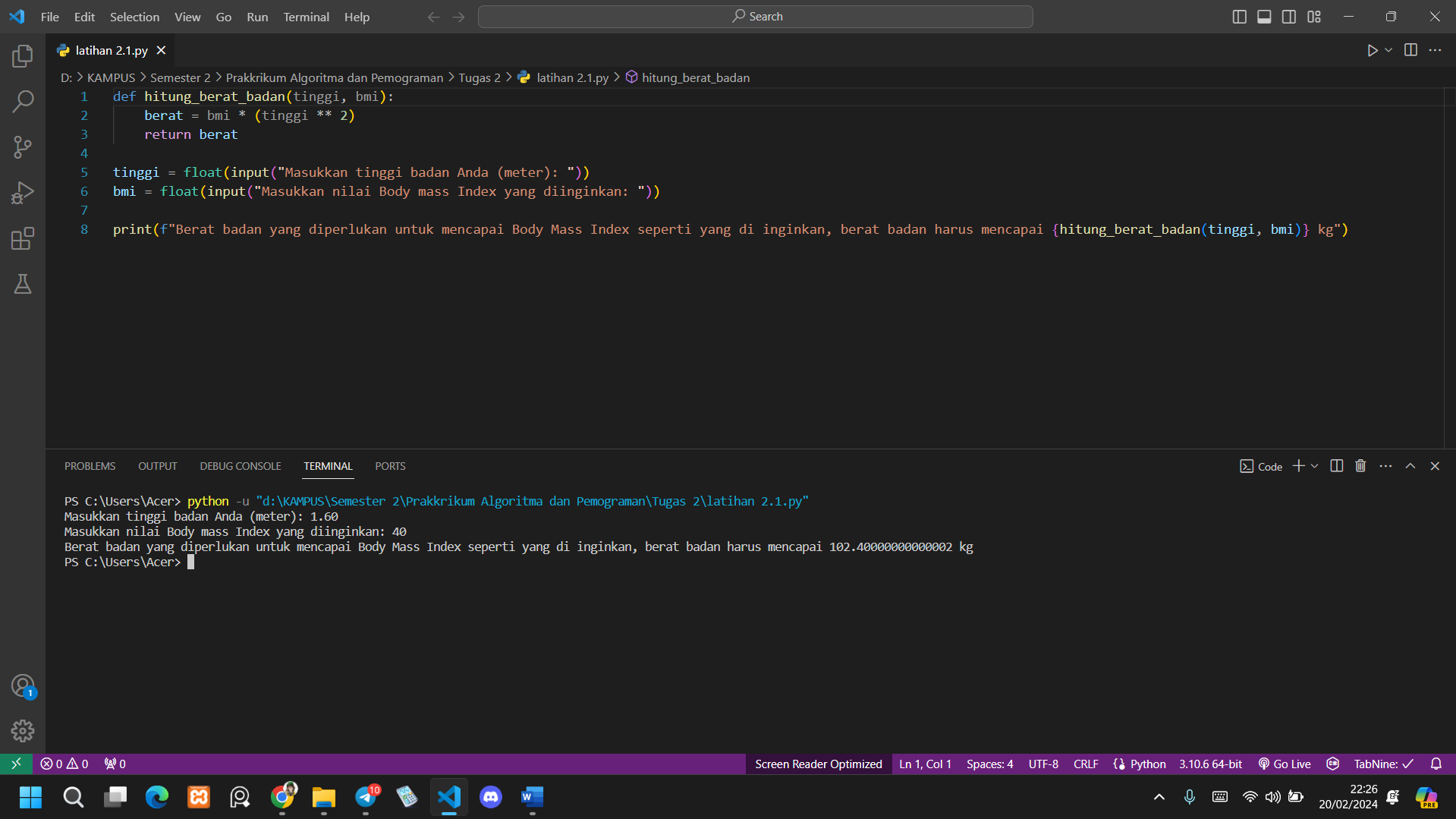
Penjelasan materi 2, dst… sesuai format ini.

## 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.

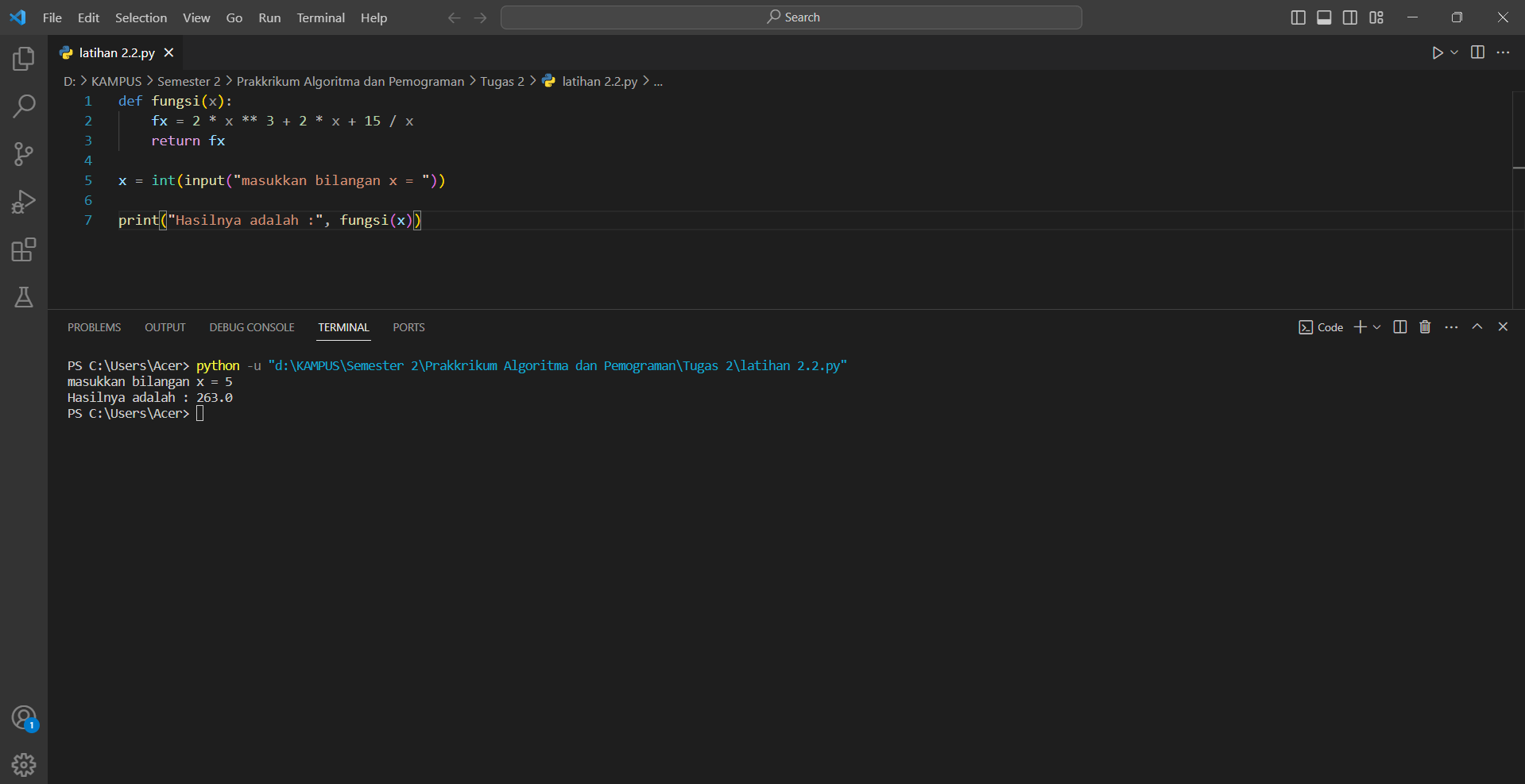
## SOAL 1

**Latihan 2.1**



## SOAL 2

**Latihan 2.2**



## SOAL 3

**Latihan 2.3**

