

MODUL PEMBELAJARAN: DASAR-DASAR PEMROGRAMAN & CODING

Panduan Konseptual, Logika Pemrograman, dan Pengenalan Sintaksis untuk Pemula

Tingkat: Dasar / Pemula

Bahasa Acuan: Python & Pseudo-code

Fokus: Logika & Algoritma

1. Pengenalan Pemrograman dan Algoritma

Pemrograman (Programming) adalah proses merancang, menulis, menguji, dan memelihara instruksi-instruksi terstruktur yang dijalankan oleh komputer untuk menyelesaikan masalah atau mencapai tujuan tertentu. Komputer pada dasarnya adalah mesin yang sangat cepat tetapi membutuhkan instruksi yang sangat spesifik dalam bentuk kode.

Coding vs Pemrograman

Coding adalah proses menerjemahkan logika pemikiran manusia ke dalam bahasa pemrograman yang dimengerti mesin.

Pemrograman mencakup spektrum yang lebih luas: analisis masalah, perancangan algoritma, penulisan kode (coding), hingga pengujian dan perawatan sistem.

Konsep Algoritma

Algoritma adalah urutan langkah-langkah logis, terstruktur, dan tidak ambigu untuk menyelesaikan suatu tugas.

Ciri Algoritma Baik: Memiliki input jelas, langkah-langkah terbatas (pasti berakhir), tepat, dan menghasilkan output yang diinginkan.

Mekanisme Eksekusi Kode: Compiler vs Interpreter

Komputer hanya memahami bahasa mesin (bilangan biner 0 dan 1). Oleh karena itu, kode program yang ditulis manusia harus diterjemahkan terlebih dahulu.

Kriteria	Compiler (Penerjemah Sekaligus)	Interpreter (Penerjemah Baris demi Baris)
Cara Kerja	Menerjemahkan seluruh kode program menjadi file biner sebelum dieksekusi.	Menerjemahkan dan menjalankan kode secara langsung, baris demi baris.
Kecepatan Eksekusi	Sangat cepat setelah dikompilasi.	Cenderung lebih lambat karena proses penerjemahan saat running.
Deteksi Kesalahan	Error ditampilkan sekaligus setelah proses kompilasi selesai.	Eksekusi berhenti tepat pada baris yang mengalami error.
Contoh Bahasa	C, C++, Rust, Go, Swift	Python, JavaScript, Ruby, PHP

2. Elemen Fundamental Pemrograman

A. Variabel dan Tipe Data

Variabel adalah nama lokasi penyimpanan di memori komputer yang digunakan untuk menampung data yang dapat berubah selama program berjalan. Setiap data memiliki **Tipe Data** yang menentukan jenis nilai dan operasi yang dapat dilakukan dengannya.

Tipe Data	Keyword	Penjelasan	Contoh Nilai
Integer	int	Bilangan bulat tanpa komponen desimal.	42, -10, 0
Floating Point	float	Bilangan real yang memiliki angka di belakang koma (desimal).	3.14, -0.001, 99.9

Tipe Data	Keyword	Penjelasan	Contoh Nilai
String	str	Kumpulan karakter atau teks yang diapit tanda petik.	"Budi", 'Pemrograman'
Boolean	bool	Nilai logika kebenaran yang hanya bernilai benar atau salah.	True atau False

```
# Contoh Deklarasi Variabel dalam Python
nama_siswa = "Budi Santoso"    # String
umur = 17                      # Integer
nilai_ujian = 88.5             # Float
is_lulus = True                 # Boolean
```

B. Operator Pemrograman

Operator digunakan untuk melakukan manipulasi atau operasi terhadap variabel dan nilai.

- **Operator Aritmatika:** Penjumlahan (+), Pengurangan (-), Perkalian (*), Pembagian (/), Modulus/Sisa Bagi (%), Pangkat (**).
- **Operator Perbandingan:** Sama dengan (==), Tidak sama dengan (!=), Lebih besar (>), Lebih kecil (<), Lebih besar/sama (>=), Lebih kecil/sama (<=).
- **Operator Logika:** AND (bernilai True jika kedua kondisi True), OR (bernilai True jika salah satu kondisi True), NOT (membalikkan nilai Boolean).

C. Percabangan / Kontrol Alur (Control Flow)

Percabangan memungkinkan komputer mengambil keputusan dan memilih blok kode mana yang akan dieksekusi berdasarkan kondisi logika tertentu.

```
# Percabangan menggunakan If-Elif-Else
nilai = 82

if nilai >= 85:
    grade = "A"
elif nilai >= 75:
    grade = "B"
elif nilai >= 65:
    grade = "C"
else:
    grade = "D"

print(f"Grade Anda: {grade}") # Output: Grade Anda: B
```

D. Perulangan (Looping)

Perulangan digunakan untuk menjalankan instruksi yang sama secara berulang tanpa harus menulis ulang baris kode tersebut.

1. For Loop (Jumlah Terukur)

Digunakan ketika jumlah iterasi/pengulangan sudah diketahui sejak awal.

2. While Loop (Berdasarkan Kondisi)

Diulang terus-menerus selama kondisi bernilai True.

```
for i in range(1, 4):
    print(f"Iterasi ke-{i}")
```

```
counter = 1
while counter <= 3:
    print(counter)
    counter += 1
```

E. Fungsi (Function)

Fungsi adalah blok kode terorganisir dan reusable (dapat digunakan kembali) yang dirancang untuk melakukan satu tugas spesifik. Fungsi membantu membuat kode lebih modular dan mudah dibaca.

```
# Definisi Fungsi dengan Parameter dan Return Value
def hitung_luas_persegi(sisi):
    luas = sisi * sisi
    return luas

# Memanggil fungsi
hasil = hitung_luas_persegi(5)
print(f"Luas Persegi: {hasil}") # Output: Luas Persegi: 25
```

3. Struktur Data Dasar

Struktur data adalah cara mengorganisir, mengelola, dan menyimpan data di komputer agar dapat diakses dan dimanipulasi secara efisien.

- **List / Array:** Kumpulan elemen terurut yang diakses menggunakan indeks (dimulai dari angka 0).

```
buah = ["Apel", "Jeruk", "Mangga"]
print(buah[0]) # Output: Apel
```

- **Dictionary / Hash Map:** Kumpulan data berbasis pasangan Kunci-Nilai (*Key-Value pair*).

```
siswa = {"nama": "Siti", "jurusan": "Informatika"}
print(siswa["nama"]) # Output: Siti
```

4. Contoh Studi Kasus Program Lengkap

Berikut adalah contoh program lengkap dalam bahasa Python yang mengintegrasikan variabel, fungsi, percabangan, dan perulangan untuk menghitung status kelulusan siswa.

```
# Program Penentuan Kelulusan Siswa

def evaluasi_siswa(nama, nilai_tugas, nilai_uts, nilai_uas):
    # Bobot: Tugas 20%, UTS 30%, UAS 50%
    nilai_akhir = (nilai_tugas * 0.2) + (nilai_uts * 0.3) + (nilai_uas * 0.5)

    if nilai_akhir >= 70:
        status = "LULUS"
    else:
        status = "REMIDIAL"

    return nilai_akhir, status

# Data beberapa siswa
daftar_siswa = [
    {"nama": "Ahmad", "tugas": 80, "uts": 75, "uas": 85},
    {"nama": "Rina", "tugas": 60, "uts": 55, "uas": 65}
]

print("=== LAPORAN HASIL BELAJAR ===")
for s in daftar_siswa:
    n_akhir, stat = evaluasi_siswa(s["nama"], s["tugas"], s["uts"], s["uas"])
    print(f"Siswa: {s['nama']} | Nilai Akhir: {n_akhir:.1f} | Status: {stat}")
```

5. Latihan Soal & Evaluasi Mandiri

Soal 1 (Pilihan Ganda): Manakah dari opsi berikut yang merupakan contoh tipe data Integer?

- A. "123"
- B. 123.0
- C. 123
- D. True

Kunci Jawaban: C. 123 adalah bilangan bulat tanpa tanda petik dan tanpa koma desimal. Option A adalah String, B adalah Float, dan D adalah Boolean.

Soal 2 (Pilihan Ganda): Beda utama antara Compiler dan Interpreter adalah...

- A. Compiler hanya untuk bahasa Python.
- B. Interpreter menerjemahkan kode sekaligus ke bentuk file executable.
- C. Compiler menerjemahkan kode sekaligus, sedangkan Interpreter baris demi baris.
- D. Interpreter tidak dapat menemukan kesalahan kode.

Kunci Jawaban: C. Compiler memproses seluruh dokumen sekaligus, sedangkan Interpreter memproses kode baris demi baris saat dijalankan.

Soal 3 (Esai Singkat): Jelaskan kegunaan fungsi (function) dalam penulisan program!

Kunci Jawaban: Fungsi digunakan untuk mengelompokkan baris kode yang menjalankan tugas spesifik agar modul kode bersifat *reusable* (dapat dipanggil berulang kali), mengurangi redundansi penulisan kode, dan mempermudah perawatan serta pembacaan program.

Rekomendasi Langkah Pembelajaran Selanjutnya:

1. Praktikkan langsung penulisan sintaksis Python menggunakan text editor (VS Code, Google Colab, atau IDLE).
2. Pelajari paradigma Pemrograman Berbasis Objek (Object-Oriented Programming / OOP).
3. Pelajari Algoritma Pencarian (Search) dan Pengurutan (Sorting).