

# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024



NIM	71200645
Nama Lengkap	Edith Felicia Putri
Minggu ke / Materi	09 / List

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  
2023

## 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

#### PENGANTAR LIST

List pada Python adalah rangkaian nilai-nilai yang dapat diakses menggunakan satu nama tunggal. List dapat berisi karakter, *integer*, *float*, maupun tipe data lainnya. List bersifat *mutable*, artinya nilainya dapat diubah secara langsung.

##### - Operasi Isi List

Menyimpan semua elemen (item) dalam tanda kurung siku '[']' dipisahkan dengan koma.

```
pujian = ["Tuhan", "Yesus", "selamanya"]
```

##### Menampilkan List.

```
list = [1, 2, 3, 4]
print(f>Nama: {list}")
```

##### Pengaksesan List sesuai indeks.

```
akses = ["b", "a", "c", "e"]
print(akses[0]) # b
print(akses[-1]) # e
print(akses[2]) # c
```

##### Pengulangan dengan perkalian di luar List.

```
ulang = [0, 1, 2] * 3
print(ulang)
```

##### - Metode dan Fungsi untuk List

Beberapa metode di dalam Python

1. **Append:** Metode yang digunakan untuk menambahkan elemen baru dan dianggap sebagai kesatuan objek pada bagian akhir list. Tidak memerlukan indeks saat penggunaannya.

##### Menambahkan angka satu di dalam list kosong.

```
satu = []
satu.append(1)
```

- **Insert:** Menambahkan satu elemen ke dalam list pada posisi yang ditentukan oleh indeks. Elemen yang ditambahkan akan memindahkan elemen-elemen yang ada ke posisi berikutnya, sehingga posisi elemen yang ada di bawah indeks yang ditentukan akan bergeser ke kanan.

Pada setiap iterasi *i*, akan dimasukkan ke dalam list1 pada indeks 0. Sehingga elemen nilai 0 sampai 3, dimasukkan ke dalam list secara terbalik: [3, 2, 1, 0].

```
list1 = []
for i in range(4):
    list1.insert(0,i)
print(list1)
```

2. **Extend:** Metode yang digunakan untuk menambahkan elemen pada sebuah list, dan memperlakukan setiap element baru sebagai elemen list sebagai individual.

```
satust = [1,2,3]
duast = [4,5,6]
tigast = [7,8,9]
satust.extend(duast)
satust.extend(tigast)
print(satust)
print(satust+duast+tigast)
```

3. **Sort:** Metode yang digunakan untuk mengurutkan elemen pada sebuah list dimulai dari nilai terkecil hingga terbesar.

**Mengurutkan sekaligus menjumlahkan.**

```
def hitung_jumlah_dan_terbesar(angka_list):
    jumlah = sum(angka_list)
    urutan = sorted(angka_list)
    terbesar = urutan[-1]
    return jumlah, terbesar # kembali dalam bentuk tuple
angka_list = [7, 14, 3, 9, 21, 6, 10, 18]
print(hitung_jumlah_dan_terbesar(angka_list))
```

**Metode-metode untuk menghapus elemen pada list:**

1. **Pop:** Jika indeks elemen sudah diketahui, dan ingin mendapatkan nilai elemen yang dihapus.

```
angka = [0,1,2,3,4,5]
angka.pop(3)
print(angka)
```

2. **Del:** Jika indeks elemen sudah diketahui, dan tidak memerlukan nilai elemen yang dihapus.

```
angka2 = angka.copy()
del angka2[2]
print(angka2)
# pop sama del, fungsinya sama
# del tidak bisa digunakan di dalam variabel sementara pop bisa.

terpop = angka.pop(3)
# terdelete = del angka2
```

3. **Remove:** Jika yang diketahui ialah nilai elemen yang akan dihapus.

```
# remove => hanya menghapus satu elemen yang ditemui lebih dulu.
angka_1 = [1,2,3,4,5,4]
angka_1.remove(4)
print(angka_1) # 4 di akhir masih ada di dalam list.
# How to remove all the same item
# Pakai looping
angka_angka = [1,2,3,4,5,3,7,5,3]
for i in range(angka_angka.count(3)):
    angka_angka.remove(3)

print(angka_angka)
```

#### Pengaturan isi-isi di dalam list.

- **len():** Untuk mendapatkan banyaknya elemen pada *list*.

```
buah = ['apel', 'pisang', 'jeruk', 'mangga', 'anggur']
jumlah_buah = len(buah)
print("Jumlah buah dalam list:", jumlah_buah)
```

- **max():** Untuk mendapatkan nilai maksimum elemen pada *list*.

```
nilai_ujian = [85, 92, 78, 90, 88]
nilai_tertinggi = max(nilai_ujian)
print("Nilai tertinggi:", nilai_tertinggi)
```

- **min():** Untuk mendapatkan nilai minimum elemen pada *list*.

```
nilai_terendah = min(nilai_ujian)
print("Nilai terendah:", nilai_terendah)
```

- **sum():** Untuk mendapatkan total nilai elemen pada *list*. Dapat digunakan hanya pada *list* dengan tipe elemen angka.

```
total_pembelian = [150000, 220000, 180000, 250000, 175000]
total_nilai_pembelian = sum(total_pembelian)
print("Total nilai pembelian:", total_nilai_pembelian)
```

### Menggabungkan List Elemen Unik.

```
def gabung(list1, list2):
    unik1 = set(list1)
    unik2 = set(list2)
    hasil = sorted(list(unik1 | unik2))
    return hasil

list1 = [3, 6, 2, 6, 3, 2, 5, 5]
list2 = [7, 4, 9, 2, 5, 8, 7]
print(gabung(list1, list2))
```

### OUTPUT:

```
PS D:\71200645_PrakAlpro_9> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe d:/71200645_PrakAlpro_9/Modul.py
[2, 3, 4, 5, 6, 7, 8, 9]
PS D:\71200645_PrakAlpro_9> █
```

### Perulangan For Loop dengan List.

```
list = ["Edith", 102.4, True, 4]
for i in list:
    print(f>Nama: {i}")
```

### OUTPUT:

```
PS D:\71200645_PrakAlpro_9> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe d:/71200645_PrakAlpro_9/Modul.py
Nama: Edith
Nama: 102.4
Nama: True
Nama: 4
PS D:\71200645_PrakAlpro_9> █
```

### Membuat Isi List Terbalik.

Objek terakhir menjadi objek pertama.

```
list_1 = [1,2,3]
list_1.reverse() # method yang pakai . (kalau tidak ada return, nilainya void)
list_2 = reversed(list_1) # mengembalikan dalam bentuk objek
print(list_1)
print(list_2)
```

```
listtt = [1,2,3]
listttt = listtt[::-1]
print(listttt)
```

Mengambil id dari variable String dan List.

```
a = "ini ada14h string"
b = ["ini", "ada", 1, 4, "string"]
print(id(b))
b[0] = 100
print(id(b))
```

Menghapus item yang ada pada indeks pertama.

Dengan cara mengambil item dimulai dari indeks 1.

```
def hapus_index_pertama(list):
    return list[1:]

list1 = [1, 2, 3, 4]
print(hapus_index_pertama(list1))
```

Mengecek elemen yang sama pada dua list yang berbeda.

```
def elemen_sama(list1, list2):
    for x in list1:
        for y in list2:
            if x == y:
                return True
    return False

print(elemen_sama([1,2,3], [3,4,5]))
```

```
def apakahsublist(A, B):
    sub_set = False
    if B == []:
        sub_set = True
    elif B == A:
        sub_set = True
    elif len(B) > len(A):
        sub_set = False
    else:
        for n in B:
            if n not in A:
                break
            else:
                sub_set = True
    return sub_set
```

```
a = [2,4,3,5,7]
b = [4,7]
```

```
c = [3, 7, 5, 2, 4]
print(apakahsublist(a, b))
print(apakahsublist(a, c))
```

## MATERI 2

### PERBEDAAN LIST DAN STRING

String merupakan deretan karakter yang mengisi setiap indeks, sedangkan list merupakan deretan nilai yang mengisi setiap indeks. Nilai dari setiap elemen list bisa diisi dengan berbagai tipe data, sedangkan string hanya diisi oleh sebuah karakter. Tetapi list dari sebuah karakter bukan berarti sebuah string.

```
a = "Hello World"
b = list(a)
print(b)
```

#### OUTPUT:

```
PS D:\71200645_PrakAlpro_9> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe d:/71200645_PrakAlpro_9/M
odul.py
['H', 'e', 'l', 'l', 'o', ' ', 'W', 'o', 'r', 'l', 'd']
```

Pada list fungsi split dapat digunakan untuk memecah sebuah kalimat menjadi beberapa kata dengan pemisah default berupa spasi.

```
kalimat = "Terima kasih, ya Tuhan"
kata = kalimat.split()
print(kata)
```

#### OUTPUT:

```
PS D:\71200645_PrakAlpro_9> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe d:/71200645_PrakAlpro_9/M
odul.py
['Terima', 'kasih,', 'ya', 'Tuhan']
PS D:\71200645_PrakAlpro_9> 
```

#### - List sebagai Parameter Fungsi

Tipe data list juga dapat digunakan pada parameter sebuah fungsi. Fungsi *append* akan memodifikasi sebuah list dan operasi *+* akan membuat variabel list baru.

```
hasil = [1, 2]
tambah = hasil.append(6)
print(hasil)
print(tambah)
lagi = hasil + [10]
print(lagi)
```

## 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 9.1** Buatlah sebuah program untuk mencari 3 nilai terbaik pada sebuah list. Program akan menampilkan 3 nilai terbaik dimulai dari yang paling tinggi. ■

Pengerjaan dilakukan dengan mengurutkan daftar nilai yang disimpan dalam sebuah *list* dari angka terkecil sampai terbesar. Lalu melakukan fungsi *reverse()* untuk mengubah urutan dari yang terbesar sampai terkecil, kemudian dilakukan print untuk mengambil tiga nilai tertinggi berdasarkan *index* dari *list*.

#### SOURCE CODE:

```
def tiga_terbaik(nilai):  
    urutan = sorted(nilai) # Merapikan dari rendah ke tinggi  
    urutan.reverse() # Urutan dari tinggi ke rendah  
    print(urutan[:3]) # Mengambil 3 nilai tertinggi  
  
tiga_terbaik([100, 35, 76, 93, 81, 22, 40, 55])
```

#### OUTPUT:

```
PS D:\71200645_PrakAlpro_9> &  
akAlpro_9/Latihan 9.1.py"  
[100, 93, 81]  
PS D:\71200645_PrakAlpro_9>
```

### SOAL 2

**Latihan 9.2** Buatlah sebuah program yang meminta masukan dari user dan menyimpannya dalam bentuk tipe data list dalam bentuk angka. Saat user memasukkan masukan 'done', maka program akan menampilkan nilai maksimum dan minimum dari deretan angka yang sudah user masukkan. Gunakan fungsi list *max()* dan *min()* pada program ini. ■

Pengerjaan dilakukan dengan membuat variabel untuk menyimpan list kosong untuk penyimpanan angka dari pengguna. Selama pengguna belum mengetikkan 'done', maka program terus meminta pengguna untuk memasukkan angka menggunakan *while True*. Lalu inputan angka bisa juga dalam bentuk *float* untuk mencegah *error*, dan sewaktu ditampilkan diubah ke bentuk *integer*. Menggunakan fungsi *min()* dan *max()* untuk mengetahui angka terbesar dan angka terkecil.

#### SOURCE CODE:

```
def simpan_user():  
    nilai = []  
    while True:  
        angka = input("Masukkan angka atau ketik 'done' untuk selesai: ")
```



```

if angka.lower() == "done":
    break

try:
    daftar_nilai = float(angka) # Coba konversi masukan ke float
    nilai.append(daftar_nilai) # Tambahkan angka ke dalam list
except ValueError:
    print("Angka tidak valid, coba lagi")

if nilai: # Periksa apakah list tidak kosong
    nilai_max = int(max(nilai)) # Cari nilai maksimum
    nilai_min = int(min(nilai)) # Cari nilai minimum
    print("Nilai maksimum adalah:", nilai_max)
    print("Nilai minimum adalah:", nilai_min)
else:
    print("Tidak ada angka yang dimasukkan.")

simpan_user()

```

## OUTPUT:

```

PS D:\71200645_PrakAlpro_9> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe "d:/71200645_PrakAlpro_9/Latihan 9.2.py"
Masukkan angka atau ketik 'done' untuk selesai: 10.4
Masukkan angka atau ketik 'done' untuk selesai: 11
Masukkan angka atau ketik 'done' untuk selesai: 54
Masukkan angka atau ketik 'done' untuk selesai: 67
Masukkan angka atau ketik 'done' untuk selesai: 34
Masukkan angka atau ketik 'done' untuk selesai: done
Nilai maksimum adalah: 67
Nilai minimum adalah: 10
PS D:\71200645_PrakAlpro_9> 

```

## SOAL 3

**Latihan 9.3** Carilah artikel berita pada media massa online. Simpan dalam bentuk file \*.txt. Buatlah program untuk membaca file \*.txt tersebut dan membaca file baris demi baris kalimat. Untuk setiap kalimat, pecahlah menjadi setiap kata dengan fungsi `split()`. Simpan kata pada variabel **kata** dengan tipe data **list**. Hasil akhir program ialah variabel **kata** akan menyimpan kata unik pada file berita tersebut. Hasil luaran seperti pada Gambar 9.1 . ■

Pengerjaan dilakukan dengan mencari berita, lalu melakukan *copy-paste* ke file \*.txt. Membuka file dengan *with open* dalam mode *read*. Membuat variabel kata berisi list kosong sebagai penyimpanan kata-kata unik, lalu menggunakan variabel hasil untuk menyimpan semua baris-baris dari fungsi *read()*, dan memisahkan setiap kata menggunakan fungsi *split()*. Melakukan *for loop* setiap kata, jika *i* tidak ada di variabel kata, maka *i* dimasukkan ke variabel kata.

## SOURCE CODE:

```
def split_kata():
    with open ("Berita.txt", "r") as file:
        kata = []
        hasil = file.read()
        pisah = hasil.split()
        for i in pisah:
            if i not in kata:
                kata.append(i)
        print(kata)
    file.close()

split_kata()
```

## OUTPUT:

```
PS D:\71200645_PrakAlpro_9> & C:/Users/HP/AppData/Local/Programs/Python/Python311/python.exe "d:/71200645_PrakAlpro_9/Latihan 9.3.py"
['KOMPAS.com', '-', 'Kepala', 'Dinas', 'Pariwisata', 'Bali', 'Tjok', 'Bagus', 'Pemayun', 'mengatakan,', 'padatnya', 'wisatawan', 'di', 'Bali', 'terjadi', 'bukan', 'karena', 'overtourism', 'atau', 'ledakan', 'wisatawan', 'melainkan', 'sebaran', 'belum', 'merata.', 'Bukan', 'overtourism', 'ada', 'beberapa', 'terkonsentrasinya', 'kunjungan', 'mancanegara', 'bagian', 'selatan,', 'kata', 'dalam', 'program', 'Weekly', 'Press', 'Briefing', 'secara', 'dari', 'mungkin', 'faktor', 'penyebaran', 'konsentrasinya', '(Bali', 'bagian)', 'sama.', 'Ia', 'memaparkan,', 'merujuk', 'data', '(wisman)', 'ke', 'Indonesia', 'jumlahnya', 'puluh', 'jika', 'dibandingkan', 'jumlah', 'wisman', 'sebelum', 'pandemi.', 'Jumlah', 'Indonesia', 'tahun', '2019', '16,11', 'juta', 'kunjungan', 'pada', '2023', '11,68', 'kunjungan.', 'Artinya', 'nasional', 'kembali', 'masa', 'pra', 'pandemi,', 'terangnya.']
PS D:\71200645_PrakAlpro_9> █
```

## Link Github:

[https://github.com/EdithFelicia/71200645\\_Guided/tree/main/71200645\\_PrakAlpro\\_9](https://github.com/EdithFelicia/71200645_Guided/tree/main/71200645_PrakAlpro_9)