



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230976</b>
<b>Nama Lengkap</b>	<b>Galih Pramana Chandra Prasetya</b>
<b>Minggu ke / Materi</b>	<b>12 / Tipe Data Set</b>

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.

### Pengenalan dan Mendefinisikan Set

Set atau biasa juga dikenal dengan istilah himpunan, adalah salah satu tipe data pada python yang dapat digunakan untuk menyimpan sekumpulan data yang unik.

Set dalam python merupakan tipe data kolektif yang digunakan untuk menyimpan banyak nilai dalam satu variabel dengan ketentuan:

- nilai anggota yang disimpan harus unik (tidak duplikat)
- nilai anggota yang sudah dimasukkan tidak bisa diubah lagi
- set bersifat unordered alias tidak berurut yang artinya tidak bisa diakses dengan index.

Sifat – sifat Set:

- Isi dari Set disebut sebagai anggota
- Set sendiri bersifat mutable (dapat menambah atau mengurangi isi dari sebuah Set)
- Anggota dari Set harus bersifat immutable

Set dapat dibuat dengan beberapa cara, salah satunya menggunakan kurung kurawal ({} ) atau fungsi set(). Berikut beberapa contohnya:

Menggunakan {}:

```
thisset = {"apple", "banana", "cherry"}  
print(thisset)
```

Menggunakan fungsi set():

```
pernah_ke_bulan= set('Neil Armstrong', 'Buzz Aldrin')
```

Apabila ingin mendefinisikan Set kosong harus menggunakan fungsi Set():

```
pernah_ke_mars= set() # menghasilkan set kosong  
print(pernah_ke_mars)
```

### Pengaksesan Set

Set tidak memiliki indeks, oleh karena itu anggota – anggota dari sebuah Set tidak bisa secara langsung diakses menggunakan indeks.

```

nim={'71200120', '71200195', '71200214'}
jumlah_nim= len(nim)
print(jumlah_nim) # akan menghasilkan output 3
# tampilkan isi setsatu-persatu
for n in nim:
    print(n)

#Output : 3
        71200195
        71200120
        71200214

```

Urutan elemen dalam Set yang dihasilkan mungkin berbeda dengan urutan saat deklarasi. Hal ini disebabkan karena Set tidak memiliki indeks dan urutan elemen tidak terdefinisi. Dalam Set, posisi elemen tidak penting.

Set termasuk tipe data yang dapat diubah, di mana elemennya dapat ditambahkan atau dihapus. Berikut contoh program untuk menambahkan elemen ke Set menggunakan fungsi `add()`:

```

# definisikan sebuah set kosong
plat_nomor = set()
# tambahkan plat nomor 'AB 1890 XA'
plat_nomor.add('AB 1890 XA')
# tambahkan plat nomor 'AD 6810 MT'
plat_nomor.add('AD 6810 MT')
# jumlah anggota di dalam Set
print(len(plat_nomor))
# tambahkan plat yang sama sekali lagi
plat_nomor.add('AB 1890 XA')
# tampilkan semua plat nomor
for plat in plat_nomor:
    print(plat)

#Output : 2
        AD 6810 MT
        AB 1890 XA

```

Saat menambahkan elemen baru ke Set, Set secara otomatis akan mengecek apakah elemen tersebut sudah ada sebelumnya (melakukan duplikasi). Jika elemen baru belum ada, maka elemen tersebut akan ditambahkan ke Set.

Namun, jika elemen baru memiliki nilai yang sama dengan elemen yang sudah ada di Set, maka fungsi `add()` tidak akan menambahkan elemen tersebut. Hal ini karena Set tidak mengizinkan duplikasi elemen. Mekanisme cek duplikasi ini sudah tertanam dalam fungsi `add()`, sehingga tidak perlu melakukannya secara manual.

Menghapus elemen dari Set dapat dilakukan dengan beberapa cara, yaitu menggunakan fungsi **`discard()`**, **`remove()`**, **`pop()`**, dan **`clear()`**.

Cara untuk menghapus anggota dari sebuah Set:

- Fungsi **`discard()`**, menghapus satu elemen yang disebutkan Fungsi `discard()` tidak akan menghasilkan error jika anggota yang ingin di hapus tidak ada di dalam Set.

```
a = {"1", "2", "3", "4",}
a.discard("1")
print(a)
# {'2', '4', '3'}
```

- **`Remove()`**, menghapus satu elemen yang disebutkan.

```
a = {"1", "2", "3", "4",}
a.remove("2")
print(a)
# {'3', '4', '1'}
```

- **`Pop()`**, mengambil salah satu dan menghapusnya dari Set (tidak tentu) Fungsi `pop()` akan berguna jika kita ingin memproses isi dari Set satu persatu tanpa memperdulikan urutan dari Setiap anggota di dalam Set.

```
a = {"1", "2", "3", "4",}
b = a.pop()
print(b) # 1
print(a) # {'2', '4', '3'}
```

- **`Clear()`**, menghapus seluruh elemen di dalam Set.

```
a = {"1", "2", "3", "4",}
a.clear()
print(a)
# set()
```

Berbeda dengan tipe data lain, Set tidak memungkinkan kita untuk mengubah nilai elemennya secara langsung. Oleh karena itu, untuk mengubah nilai elemen dalam Set, kita perlu melakukan operasi penggantian. Caranya adalah dengan menghapus elemen yang ingin diubah terlebih dahulu, kemudian memasukkan elemen baru dengan nilai yang diinginkan.

```
Hewan = set(['ayam', 'dinosaurus', 'sapi', 'kambing', 'kucing'])
print(Hewan) # {'dinosaurus', 'ayam', 'kambing', 'kucing', 'sapi'}

Hewan.remove('ayam')
Hewan.add('kuda')
print(Hewan) # {'dinosaurus', 'kambing', 'kucing', 'sapi', 'kuda'}
```

Pada program diatas, dapat disimpulkan bahwa Set tidak mengenal posisi atau urutan data. Sehingga output dari program tersebut hanya menampilkan data yang baru namun urutan tidak sama. Setiap kali ada operasi penambahan dan penghapusan maka urutan anggota di dalam Set biasanya akan berubah.

## Operasi – Operasi Pada Set

Operasi Set merupakan operasi yang dilakukan pada kumpulan data. Berikut adalah beberapa operasi Set yang tersedia dalam Python:

- **Operator Union (Union):** Menggabungkan dua Set menjadi satu Set baru yang berisi semua elemen dari kedua Set awal. Operasi ini dapat dilakukan dengan menggunakan operator '|' atau fungsi `union()`.
- **Operator Intersection (Intersection):** Menghasilkan Set baru yang berisi elemen-elemen yang sama-sama terdapat di kedua Set awal. Operasi ini dapat dilakukan dengan menggunakan operator '&' atau fungsi `intersection()`.
- **Operator Difference (Difference):** Menghasilkan Set baru yang berisi elemen-elemen yang terdapat di Set pertama, **kecuali** elemen-elemen yang juga terdapat di Set kedua. Operasi ini dapat dilakukan dengan menggunakan operator '-' atau fungsi `difference()`.
- **Operator Symmetric Difference (Symmetric Difference):** Menghasilkan Set baru yang berisi elemen-elemen yang **tidak** sama-sama terdapat di kedua Set awal. Operasi ini dapat dilakukan dengan menggunakan operator '^' atau fungsi `symmetric_difference()`.

**Operator Union:** menggabungkan dua Set menjadi satu. Menggunakan operator | atau menggunakan fungsi `union()`

```
merek_hp = {'Samsung', 'Apple', 'Xiaomi', 'Sony'}
merek_ac = {'LG', 'Samsung', 'Panasonic', 'Daikin', 'Sony'}
# bisa juga menggunakan 'gabungan = merek_hp | merek_ac'
gabungan = merek_hp.union(merek_ac)
print(gabungan)
#{'LG', 'Panasonic', 'Xiaomi', 'Samsung', 'Sony', 'Apple', 'Daikin'}
```

**Operator Intersection:** menghasilkan irisan dari dua Set. Menggunakan operator & atau fungsi intersection().

```
renang = {'siti', 'mail', 'ikhsan', 'upin', 'ipin'}
tenis = {'joko', 'mail', 'ipin', 'upin', 'tejo'}
# suka renang dan tenis
renang_tenis = renang & tenis
print(renang_tenis)
print()
#{'upin', 'ipin', 'mail'}
```

**Operator Difference:** menghasilkan selisih dari dua Set dan membuatnya menjadi Set baru yang sudah dibandingkan.

```
english = {'desi', 'tono', 'evan', 'miko', 'takashi', 'chaewon'}
korean = {'chaewon', 'yeona', 'erika', 'miko'}

only_korean = korean - english
print(only_korean) # {'yeona', 'erika'}

only_english = english - korean
print(only_english) # {'desi', 'evan', 'tono', 'takashi'}
```

**Operator Symmetric Difference:** menghasilkan Set baru yang berasal dari dua Set yang dijumlahkan kecuali irisannya.

```
english = {'desi', 'tono', 'evan', 'miko', 'takashi', 'chaewon'}
korean = {'chaewon', 'yeona', 'erika', 'miko'}
one_language = english ^ korean
print(one_language)
#{'evan', 'tono', 'takashi', 'erika', 'desi', 'yeona'}
```

## 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

Sc:

```
#latman 12.1
inkategori = int(input("Masukkan Jumlah Kategori: "))
data_apk = {}

for i in range(inkategori):
    nama_kategori = input("Masukkan Nama Kategori: ")
    print("Masukkan 5 nama aplikasi di kategori", nama_kategori)
    apk = list()

    for j in range(5):
        nama_apk = input("Masukkan Nama Aplikasi: ")
        apk.append(nama_apk)
    data_apk[nama_kategori] = apk
print(data_apk)
print()

daftar_apk_lst = list()
for apk in data_apk.values():
    daftar_apk_lst.append(set(apk))
print(daftar_apk_lst)
print()

hasil = daftar_apk_lst
for x in range(1,inkategori):
    unik = hasil[x-1] ^ hasil[x]
    print("Aplikasi Yang Hanya Muncul Di 1 Kategori Saja: ", unik)

aplikasi_di_dua_kategori = set()
for x in range(inkategori-1):
    for y in range(x+1, inkategori):
        aplikasi_di_dua_kategori |= hasil[x] & hasil[y]

print("Aplikasi Yang Muncul Di Tepat Dua Kategori Saja: ",
aplikasi_di_dua_kategori)
```

Output:

```
PS D:\PRA Alpro sem 2\pra alpro\Pertemuan 12> & C:/Users/Asus/Appdata/Local/Programs/Python/Python311/python.exe d:/PRA A
Masukkan Jumlah Kategori: 2
Masukkan Nama Kategori: game
Masukkan 5 nama aplikasi di kategori game
Masukkan Nama Aplikasi: ML
Masukkan Nama Aplikasi: PUBG
Masukkan Nama Aplikasi: COC
Masukkan Nama Aplikasi: CODM
Masukkan Nama Aplikasi: Genshin
Masukkan Nama Kategori: Santai
Masukkan 5 nama aplikasi di kategori Santai
Masukkan Nama Aplikasi: Tiktok
Masukkan Nama Aplikasi: WA
Masukkan Nama Aplikasi: Genshin
Masukkan Nama Aplikasi: IG
Masukkan Nama Aplikasi: FB
{'game': ['ML', 'PUBG', 'COC', 'CODM', 'Genshin'], 'Santai': ['Tiktok', 'WA', 'Genshin', 'IG', 'FB']}

[{'PUBG', 'CODM', 'ML', 'COC', 'Genshin'}, {'WA', 'Tiktok', 'FB', 'Genshin', 'IG'}]

Aplikasi Yang Hanya Muncul Di 1 Kategori Saja: {'WA', 'Tiktok', 'PUBG', 'CODM', 'ML', 'FB', 'COC', 'IG'}
Aplikasi Yang Muncul Di Tepat Dua Kategori Saja: {'Genshin'}
```

Penjelasan:

Kode tersebut meminta pengguna untuk memasukkan jumlah kategori aplikasi, kemudian untuk setiap kategori, memasukkan nama kategori dan lima nama aplikasi yang termasuk dalam kategori tersebut, yang kemudian disimpan dalam dict 'data\_apk'. Data aplikasi dari setiap kategori dikonversi ke dalam set untuk menghilangkan duplikat dan disimpan dalam 'daftar\_apk\_lst'. Kode ini kemudian menampilkan aplikasi yang hanya muncul di satu kategori dengan membandingkan set aplikasi antar kategori menggunakan operasi XOR. Selain itu, kode juga mencari dan menampilkan aplikasi yang muncul di tepat dua kategori dengan melakukan operasi irisan (intersection) pada set aplikasi dari kategori yang berbeda.



## SOAL 2

Sc:

```
#latman 12.2
data = input("Masukkan kalimat: ")
data_list = list(data)
print("List ke Set")
print("sebelum ->",data_list)
print("sesudah ->",set(data_list))
print()
data_set = set(data)
print("Set ke List")
print("sebelum ->",data_set)
print("sesudah ->",list(data_set))
print()
data_tuple = tuple(data)
print("Tuple ke Set")
print("Sebelum ->",data_tuple)
print("Sesudah ->",set(data_tuple))
print()
print("Set ke Tuple")
print("sebelum ->",data_set)
print("sesudah ->",tuple(data_set))
```

Output:

```
PS D:\Pra Alpro sem 2\pra alpro\Pertemuan 12> & C:/Us
Masukkan kalimat: halo
List ke Set
sebelum -> ['h', 'a', 'l', 'o']
sesudah -> {'o', 'h', 'a', 'l'}

Set ke List
sebelum -> {'o', 'h', 'a', 'l'}
sesudah -> ['o', 'h', 'a', 'l']

Tuple ke Set
Sebelum -> ('h', 'a', 'l', 'o')
Sesudah -> {'o', 'h', 'a', 'l'}

Set ke Tuple
sebelum -> {'o', 'h', 'a', 'l'}
sesudah -> ('o', 'h', 'a', 'l')
```

Penjelasan:

1. Input Data:
  - Program Meminta pengguna memasukkan sebuah kalimat dan menyimpannya dalam variabel 'data'.
2. List ke Set:
  - Mengubah kalimat yang diinput menjadi list '(data\_list)'.
  - Mencetak list tersebut sebelum dan sesudah dikonversi ke set. Set akan menghilangkan elemen-elemen duplikat.
3. Set ke List:
  - Mencetak set tersebut sebelum dan sesudah dikonversi ke list. Konversi ini mengembalikan elemen-elemen set ke dalam bentuk list.
4. Tuple ke Set:
  - Mengubah kalimat yang diinput menjadi tuple '(data\_tuple)'.
  - Mencetak tuple tersebut sebelum dan sesudah dikonversi ke set.

### SOAL 3

Sc:

```
#latman 12.3
teks1 = input("Masukkan file teks 1: ")
teks2 = input("Masukkan file teks 2: ")
try:
    handle = open(teks1, 'r')
    handle2 = open(teks2, 'r')
except:
    print("File cannot be opened")
    exit()
a = set()

for teks in handle:
    tekslow = teks.lower()
    tekspl = tekslow.split()
    for b in tekspl:
        a.add(b)
for teks in handle2:
    tekslow = teks.lower()
    tekspl = tekslow.split()
    for b in tekspl:
        a.add(b)

print(a)
```

Output:

```
PS D:\Pra Alpro sem 2\pra alpro\Pertemuan 12> & C:/Users/Asus/AppData/Local/Programs/Python/Python311/python.exe "d:/Pra Alpro sem 2/pra alpro/Pertemuan 12/laprak3.py"
Masukkan file teks 1: teks1.txt
Masukkan file teks 2: teks2.txt
{'makanan', 'mata', 'buah', 'gajah', 'kesukaan', 'nama', 'bermain', 'hati', 'geli', 'minuman', 'hobi', 'putih-putih', 'tak', 'melati', 'bagaimana', 'adalah', 'saya', 'merah-merah', 'delima', 'melihat', 'bunga'}
```

Penjelasan:

1. Membaca Nama File:
  - Program meminta pengguna memasukkan dua nama file teks.
2. Membuka File:
  - Program mencoba membuka kedua file dalam mode baca ('r').
  - jika file tidak dapat dibuka, program mencetak pesan error dan keluar.
3. Inisialisasi Set:
  - Program membuat sebuah set kosong 'a' untuk menyimpan kata-kata unik.
4. Memproses File Pertama:
  - Program akan membaca setiap baris dalam file pertama.
  - Mengubah teks menjadi huruf kecil.
  - Memecah baris menjadi kata-kata.
  - Menambahkan setiap kata ke dalam set 'a'.

5. Memproses File Kedua:

- Membaca setiap baris dalam file kedua.
- Mengubah teks menjadi huruf kecil.
- Memecah baris menjadi kata-kata.
- Menambahkan setiap kata ke dalam set 'a'.

6. Mencetak Set:

- Menampilkan semua kata unik yang ditemukan dalam kedua file.

Link github: <https://github.com/GalihPramana/Praktikum-Alpro-71230976.git>