



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	71200609
<b>Nama Lengkap</b>	Airell Aristo Subagia
<b>Minggu ke / Materi</b>	11 / Tuple

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

# Tuple

## Apa Itu Tuple

Dalam pemrograman Python, tuple merupakan salah satu struktur data yang digunakan untuk menyimpan sekumpulan item. Berbeda dengan list, tuple bersifat immutable (tidak dapat diubah, ditambah, ataupun dihapus). Untuk membuat sebuah tuple digunakan tanda kurung “()”

Contoh :

```
#Cara Pembuatan List
list = ["a", "b", "c"]

#Cara Pembuatan Tuple
tuple = ("a", "b", "c")

print(list)
print(tuple)
```

## Mengakses Nilai pada Tuple

Untuk mengakses nilai yang ada pada tuple dapat digunakan cara sebagai berikut:

1. Menggunakan indeks secara langsung

```
#Cara Pembuatan Tuple
tuple = ("a", "b", "c")

print(tuple[0])
print(tuple[1])
```

Outputnya :

```
a
b
```

2. Menggunakan perulangan

```
#Cara Pembuatan Tuple
tuple = ("a", "b", "c")

for item in tuple :
    print(item)
```

Outputnya :

```
a
b
c
```

## Membandingkan Pada Tuple

Kita dapat membandingkan tuple menggunakan operator perbandingan (`==`, `!=`, `<`, `>`, `<=`, dan `>=`). Pada dasarnya cara kerja untuk membandingkan nilai pada tuple ialah, membandingkan dimulai dari elemen pertama secara sekuensial. Jika ditemukan ada perbedaan maka akan digunakan elemen tersebut sebagai pembandingnya Contoh :

### 1. Penggunaan (`==` dan `!=`)

```
Tuple1 = (1, 2, 3)
tuple2 = (1, 2, 3)
tuple3 = (1, 2, 4)

print(tuple1 == tuple2)
print(tuple1 != tuple3)
```

Outputnya :

```
True
True
```

### 2. Penggunaan (`>`, `<`, `<=`, `>=`)

```
tuple1 = (1, 2, 3)
tuple2 = (1, 2, 4)
tuple3 = (1, 3, 0)
tuple4 = (0, 2, 3)

print(tuple1 < tuple2)
print(tuple1 > tuple4)
print(tuple1 <= tuple3)
print(tuple1 >= tuple2)
```

Outputnya :

```
True
True
True
False
```

## Function dan Method Yang Tersedia Pada Tuple

Ada beberapa function dan method tuple yang biasa digunakan pada bahasa pemrograman python. Diantaranya :

### 1. len()

Digunakan untuk mengembalikan panjang dari sebuah tuple.

```
tuple1 = (1, 2, 3)
print(len(tuple1))
```

Outputnya :

```
3
```

### 2. max()

Digunakan untuk mengembalikan elemen terbesar pada sebuah tuple.

```
tuple1 = (1, 2, 3)
print(max(tuple1))
```

Outputnya :

```
3
```

### 3. min()

Digunakan untuk mengembalikan elemen terkecil pada sebuah tuple.

```
tuple1 = (1, 2, 3)
print(min(tuple1))
```

Outputnya :

```
1
```

### 4. sum()

Digunakan untuk mengembalikan jumlah semua elemen pada sebuah tuple.

```
tuple1 = (1, 2, 3)
print(sum(tuple1))
```

Outputnya :

```
6
```

5. sorted()

Digunakan untuk mengembalikan sebuah list terurut dari elemen yang ada pada tuple.

```
tuple1 = (3, 1, 2)
print(sorted(tuple1))
```

Outputnya :

```
[1, 2, 3]
```

6. count()

Mengembalikan jumlah kemunculan suatu nilai pada tuple.

```
tuple1 = (1, 2, 3, 1)
print(tuple1.count(1))
```

Outputnya :

```
2
```

7. index()

Mengembalikan indeks pertama dari kemunculan nilai yang dicari pada suatu tuple.

```
tuple1 = (1, 2, 3, 1)
print(tuple1.index(1))
```

Outputnya :

```
0
```

## BAGIAN 2: LATIHAN MANDIRI (60%)

### SOAL 1

```
#Latihan 11.1
tA = (90,90,90,90)

def allEqual(x) :
    count = 0
    simpan = x[0]
    for i in x :
        if simpan == i :
            count += 1
    if count == len(x) :
        return True
    else :
        return False

print(allEqual(tA))
```

Outputnya :

**True**

Penjelasannya :

1. Pertama-tama, dibuat dulu function bernama **allEqual()** yang dimana parameternya akan berupa tuple.
2. Setelah itu, dibuat variable bernama **count** yang digunakan untuk menghitung seberapa banyak nilai yang sama.
3. Kemudian, dibuat juga variable berikutnya bernama **simpan** dilakukan untuk menjadi pembanding antara nilai yang berada pada tuple.
4. Setelah itu, dilakukan perulangan **for loop** untuk mengakses nilai yang ada pada tuple dan dilakukan pengecekan apakah nilai itu sama atau tidak menggunakan **if statement**. Jika sama maka count akan ditambahkan dengan 1.
5. Terakhir, dilakukan **if statement** untuk checking apakah jumlah yang benar itu sama dengan panjang dari tuple. Jika sama maka dapat dipastikan semua nilai itu equal.

## SOAL 2

```
#Latihan 11.2
Data= ('Matahari Bhakti Nendya', '22064091', 'Bantul, DI Yogyakarta')

def dataDiri(x) :
    print("NIM :",x[1])
    print("NAMA :",x[0])
    print("ALAMAT :",x[2])
    print()
    nama = x[0].split()
    namaTerbalik = [nama[len(nama)-1-i] for i in range(len(nama))]
    print("NIM :",tuple(x[1]))
    print("NAMA DEPAN :",tuple(nama[0]))
    print('NAMA TERBALIK :', tuple(namaTerbalik))

dataDiri(Data)
```

Outputnya :

```
NIM : 22064091
NAMA : Matahari Bhakti Nendya
ALAMAT : Bantul, DI Yogyakarta

NIM : ('2', '2', '0', '6', '4', '0', '9', '1')
NAMA DEPAN : ('M', 'a', 't', 'a', 'h', 'a', 'r', 'i')
NAMA TERBALIK : ('Nendya', 'Bhakti', 'Matahari')
```

Penjelasannya :

1. Pertama-tama, dibuat function bernama **dataDiri()** dengan parameternya berupa tuple.
2. Setelah itu, dilakukan printing sederhana untuk mendapatkan NIM, nama, dan alamat.  
NIM indeks 1, nama indeks 0, dan alamat indeks 2
3. Kemudian, dibuat variable **nama**. Yang pertama-tama digunakan untuk split.
4. Setelah itu, dibuat varibale **namaTerbalik**. Dengan digunakan cara list comprehension yang dimana akan mengembalikan semua nama secara terbalik karena indeksing dimulai dari panjang dari variable **nama** lalu akan dikurangi sebanyak berapa kali perulangan dilakukan.
5. Terakhir, tinggal dilakukan printing untuk mendapatkan nim cukup memasukkan string nim yang sudah dislicing kedalam tuple, sama dengan nama depan memasukkan

variable nama. Dan terakhir sisa print nama terbalik dengan memanggil variable **namaTerbalik**.

### SOAL 3

```
#Latihan 11.3
fileName = input("Enter A Name File : " )

text = open("c:\\Users\\Airell\\Documents\\Tugas Kampus\\Semester 8\\Praktikum AlPro\\Pertemuan11\\"+fileName,"r")
dict = {}

for baris in text :
    x = baris.split()
    if len(x) >= 1 :
        if x[0] == "From" :
            jam = x[5].split(":")[0]
            if jam not in dict.keys() :
                dict[jam] = 1
            else :
                dict[jam] += 1

sorted_data = {key: dict[key] for key in sorted(dict.keys())}
for key,value in sorted_data.items() :
    print(key, value)
```

Outputnya :

```
Enter A Name File : mbox-short.txt
04 3
06 1
07 1
09 2
10 3
11 6
14 1
15 2
16 4
17 2
18 1
19 1
```

Penjelasannya :

1. Pertama-tama, dibuat variable bernama **fileName** untuk menyimpan nama file yang akan dibuka.
2. Setelah itu, dibuka di varibale bernama **text** dengan mode **read**.



3. Kemudian, dibuat **dictionary** bernama **dict** untuk menyimpan jam dari surat nantinya.
4. Setelah itu , dilakukan **for loop** yang dimana digunakan untuk mendapatkan baris pada artikel lalu **split()** menggunakan spasi.
5. Kemudian, dilakukan **filtering menggunakan if statement** kembali untuk menghilangkan semua baris kosong dengan check panjang dari listnya.
6. Terakhir, dilakukan checking lagi apakah pada baris tersebut kata pertama mengandung huruf **"From"**. Jika iya maka akan dilanjutkan ke **if statement berikutnya** untuk check apakah **jam sudah ada** didalam dictionary atau belum, jika sudah tinggal ditambahkan 1 jika belum dibuat key dan value yang baru.
7. Kemudian, dilakukan sorting yang dimana akan membuat sebuah dictionary baru dengan cara dimana nilainya sudah disorting.
8. Terakhir, tinggal dilakukan printing saja.

Link Github :

[https://github.com/AirellAristo/TugasPrakAlpro/tree/main/71200609\\_Pertemuan11\(Tuple\)](https://github.com/AirellAristo/TugasPrakAlpro/tree/main/71200609_Pertemuan11(Tuple))