



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230987</b>
<b>Nama Lengkap</b>	<b>Prastha Pradipta Purusa</b>
<b>Minggu ke / Materi</b>	<b>11 / Tipe Data</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.

### Tuple Immutable

Tuple dan list adalah dua struktur data dalam Python yang mirip; keduanya dapat menyimpan beberapa nilai dengan indeks bilangan bulat. Perbedaan utama adalah tuple bersifat immutable, sehingga elemen-elemennya tidak dapat diubah setelah dibuat, sementara list bersifat mutable. Tuple lebih efisien dalam penggunaan memori dan dapat digunakan sebagai kunci dalam dictionary karena sifatnya yang hashable. Contohnya ada di program berikut:

```
114 t = 1,2,3,4,5,6,7,8
115 type(t)
116 print(t)
```

```
[Running] python -u "c:\Users\HP\prasss020904\bab12\ciba.py"
(1, 2, 3, 4, 5, 6, 7, 8)
```

Jika dalam bentuk string, list atau tuple akan memisahkan semua yang ada di string. Contoh program:

```
tuple = tuple('asik sendiri')
print(tuple)
```

```
[Running] python -u "c:\Users\HP\prasss020904\bab12\ciba.py"
('a', 's', 'i', 'k', ' ', 's', 'e', 'n', 'd', 'i', 'r', 'i')
```

Tuple merupakan nama dari constructor dan tidak bisa menggunakannya sebagai nama variabel. Cara mengambil rentang nilai dari tuple:

```
tuple = tuple('asik sendiri')
print(tuple[2:4])
```

```
[Running] python -u "c:\Users\HP\prasss020904\bab12\ciba.py"
('i', 'k')
```

Tuple bersifat immutable artinya element pada tuple tidak dapat berubah. Elemen pada tuple dapat diganti. Contoh program:

```
isi = ('dlsfpospodfkosp', 'fodshfoh', 'nxbckzbdi')
isi = ('A',) + isi[1:]
print(isi)
```

```
[Running] python -u "c:\Users\HP\prasss  
( 'A', 'dlsfpospodfkosp', 'fodshfoh' )
```

## Membandingkan Tuple

Perbandingan dapat bekerja pada tuple dan model sekuensial seperti list, dictionary, set. Cara kerjanya adalah membandingkan elemen pertama dari setiap sekuensial yang ada. Jika menemukan yang sama maka akan berlanjut ke elemen berikutnya dan proses berlangsung terus menerus hingga menemukan perbedaan. Contoh program:

```
a = (0,1,2)  
b = (42894038,4903984,3294)  
c = a < b  
print(c)
```

```
[Running]  
True
```

Fungsi sort pada python bekerja dengan cara yang sama yaitu melakukan pengurutan berdasarkan elemen pertama dan pada kasus tertentu akan mengurutkan berdasarkan elemen kedua atau lainnya dan disebut sebagai DSU – (Decorate, Sort, Undercorate).

1. Decorate – urutan membangun daftar tuple dengan satu atau lebih key pengurutan sebelum elemen dari urutan
2. Sort – list tuple menggunakan sort.
3. Undercorate – melakukan ekstraksi pada elemen yang telah diurutkan pada satu sekuensial.

Contoh Program:

```
130 kata = 'wisnu gay banget pelangi'  
131 daftarkata = kata.split()  
132 tuple = list()  
133 for i in daftarkata:  
134     tuple.append((len(i),i))  
135 tuple.sort()  
136 urutan = list()  
137 for panjang , i in tuple:  
138     urutan.append(i)  
139 print(urutan)  
140
```

```
[Running] python -u "c:\Users\HP\prasss  
['gay', 'wisnu', 'banget', 'pelangi']
```

Program tersebut ada kata 'wisnu gay banget pelangi'. Untuk daftarkata kita split untuk memisahkan kata. Variable tuple di list. Untuk looping di i dalam daftarkata, tuple di append lalu menggunakan len untuk mengukur panjang dari i. Kemudian tuple di sort. Urutan di list. Untuk looping panjang dan i di tuple lalu urutan di gabungkan dengan append i.

## Penugasan Tuple

Salah satu fitur unik dari Python adalah memiliki tuple disisi kiri dari statement penugasan dan bisa menetapkan lebih dari satu variabel pada sisi sebelah kiri secara berurutan. Contohnya ketika ada dua daftar elemen yang merupakan urutan ke dalam variabel. Contoh program:

```
141 isi = ['i', 'j', 'k']
142 i, j, k = isi
143 print(k)
```

```
141 isi = ['i', 'j', 'k']
142 k = isi[2]
143 print(k)
```

Tuple dapat menukar nilai dalam 1 variabel. Bagian kiri merupakan tuple dari variable dan bagian kanan merupakan tuple dari expressions dan tiap nilai pada bagian kanan diberikan ke masing-masing variable di sebelah kiri. Jumlah variabel antara sisi kiri dan sisi kanan harus sama.

```
141 isi = ['i', 'j', 'k']
142 isi = ['j', 'k', 'i']
143 print(isi)
```

Pada sisi sebelah kanan biasanya terdapat data sekuensial string, list atau tuple. Contoh program:

```
145 email = 'prastha.pradipta@ti.ukdw.ac.id'
146 user, domain = email.split('@')
147 print(user, '\n', domain)
```

```
[Running] python -u "c:\Users\HP\prasss020904\bab12\ciba.py"
prastha.pradipta
ti.ukdw.ac.id
```

## Dictionary dan Tuple

Dictionaries mempunyai metode yang disebut items untuk memunculkan nilai list dari tuple dimana tiap tuple-nya merupakan key-value pair. Item nilainya tidak berurutan dan harus diurutkan dengan sort pada tuple. Contoh program:

```
149 dict = {'x':143243, 'z':1324, 'c':22654}
150 t = list(dict.items())
151 t.sort()
152 print(t)
```

```
[Running] python -u "c:\Users\HP\prasss020904\bab12\ciba.py"
[('x', 143243), ('z', 1324), ('c', 22654)]
```

List yang muncul diurutkan secara ascending berdasarkan alfabet dan key value.

## Multipenugasan dengan Dictionary

Gabungan items, tuple assignment dan for akan menghasilkan kode tertentu pada keys dan values dari dictionary. Cara penggunaan:

```
154 d = {'x':143243, 'z':1324, 'c':22654}
155 for key, val in list(d.items()):
156     print(key, val)
```

x	143243
z	1324
c	22654

Terdapat 2 looping variabel karena item memunculkan list dari tuple dan key. Val merupakan penugasan tuple yang melakukan looping melalui pasangan key-value pada dictionary.

```
dict = {'x':143243, 'z':1324, 'c':22654}
l = list()
for key, val in dict.items():
    l.append((val, key))
r = sorted(l, reverse=True)
print(l)
print(r)
```

```
[(143243, 'x'), (1324, 'z'), (22654, 'c')]
[(143243, 'x'), (22654, 'c'), (1324, 'z')]
```

Program tersebut menciptakan kamus dict dengan tiga kunci-nilai dan kemudian membentuk sebuah daftar kosong l. Selanjutnya, program melakukan perulangan melalui setiap pasangan kunci-nilai dalam kamus dict dan menambahkan pasangan nilai-kunci ke dalam daftar l sebagai tuple dan r menggunakan sorted l untuk membalikan nilai dari yang terbesar. Dengan melakukan penyusunan list dari tuple yang memiliki value sebagai elemen pertama akan memudahkan mengurutkan list dari tuple dan mendapatkan isi dictionary yang diurutkan berdasarkan value.

## Kata yang sering muncul

Cara memunculkan kata-kata yang sering muncul dalam text dengan contoh 5 kata terbanyak:

```
192 import string
193 file = open('txt.txt', 'r')
194 hitung_kata = dict()
195 for baris in file:
196     baris = baris.lower()
197     kata = baris.split()
198     for kata in kata:
199         # Hapus tanda baca dari kata
200         kata = kata.strip(string.punctuation)
201         if kata:
202             if kata not in hitung_kata:
203                 hitung_kata[kata] = 1
204             else:
205                 hitung_kata[kata] += 1
206 lst = list()
207 for kunci, nilai in list(hitung_kata.items()):
208     lst.append((nilai, kunci))
209 lst.sort(reverse=True)
210 for kunci, nilai in lst[:5]:
211     print(kunci, nilai)
```

```
[Running] python  
5 saya  
3 untuk  
2 pukul  
2 konferensi  
2 di
```

Program digunakan untuk membaca file dan menghitung kemunculan setiap kata dalam dokumen. Dengan menambahkan print out counts, list tuple (val, key) dibuat dan diurutkan dalam list dengan urutan terbalik. Nilai pertama digunakan dalam perbandingan. Jika ada lebih dari satu tuple dengan nilai yang sama, program lanjut elemen kedua (key) sehingga tuple jika nilai sama akan diurutkan abjad sesuai key. Pada bagian akhir, terdapat looping pada list dengan Batasan 5, kemudian mencetak 5 kata yang paling sering muncul dengan melakukan perulangan pada list .

### Tuple sebagai kunci Dictionaries

Tuple merupakan hashable dan bukan list dan dapat menggunakan tuple sebagai key dalam dictionary. Contoh program:

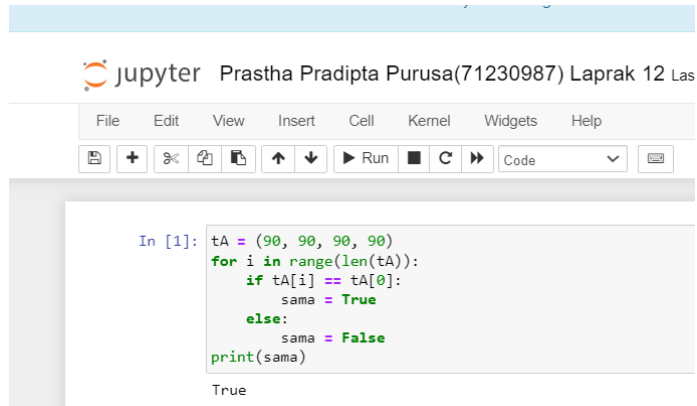
```
213 a = 'ajskdajds'  
214 b = 'fkdsjfpjds'  
215 c = 'uu382u4ho'  
216 d = dict()  
217 d['a','b'] = c  
218 for key in d:  
219     print(a, b, d[key])  
220
```

```
[Running] python -u "c:\Users\HP\pras  
ajskdajds fkdsjfpjds uu382u4ho
```

## 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



The screenshot shows a Jupyter Notebook window titled "Prastha Pradipta Purusa(71230987) Laprak 12 Las". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and other functions. A code cell is active, containing the following Python code:

```
In [1]: tA = (90, 90, 90, 90)
for i in range(len(tA)):
    if tA[i] == tA[0]:
        sama = True
    else:
        sama = False
print(sama)
```

The output of the code cell is "True".

Line1: membuat tuple tA

Line2: Perulangan for berjalan sebanyak jumlah elemen dalam tuple tA. Fungsi len(tA) untuk mengecek jumlah panjang tuple

Line3: mengecek apakah i sama dengan nilai pertama tA

Line4: Jika sama maka akan menghasilkan True

Line5: Kondisi lain jika i tidak sama dengan nilai pertama tA

Line6: sama maka akan menghasilkan False

Line7: mengeluarkan output dari variabel sama dan jika ya maka akan muncul output True jika tidak maka akan muncul output False

## SOAL 2

```
In [2]: nama = 'Matahari Bhakti Nendya'
nim = '22064091'
alamat = 'Bantul, DI Yogyakarta'
print("Data: ", (nama, nim, alamat))
print("NIM :", nim)
print("NAMA :", nama)
print("ALAMAT :", alamat)
nim = tuple(nim)
namadepan = tuple(nama.split()[0])
namaterbalik = tuple(nama.split()[::-1])
print("NIM:", nim)
print("NAMA DEPAN:", namadepan[1:])
print("NAMA TERBALIK:", namaterbalik)
```

Data: ('Matahari Bhakti Nendya', '22064091', 'Bantul, DI Yogyakarta')

NIM : 22064091

NAMA : Matahari Bhakti Nendya

ALAMAT : Bantul, DI Yogyakarta

NIM: ('2', '2', '0', '6', '4', '0', '9', '1')

NAMA DEPAN: ('a', 't', 'a', 'h', 'a', 'r', 'i')

NAMA TERBALIK: ('Nendya', 'Bhakti', 'Matahari')

Line1: variabel nama dan isi nama

Line2: nim dan string nim

Line3: Alamat dan isi string Alamat

Line4: memunculkan output tulisan Data: dan isi nama,nim,Alamat

Line5: memunculkan output tulisan NIM: dan isi nim

Line6: memunculkan output tulisan NAMA: dan isi nama

Line7: memunculkan output tulisan ALAMAT: dan isi Alamat

Line8: mengubah menjadi tuple untuk nim

Line9: mengubah menjadi tuple dan memisahkan lagi dengan split mengambil index 0 untuk nama depan

Line10: diubah menjadi tuple dan di pisah tiap nama dengan split dan mengambil index[::-1] untuk membuat nama terbalik

Line11: memunculkan output nim setelah di tuple

Line12: memunculkan output namadepan tetapi huruf paling depan tidak di ambil sehingga menggunakan [1:] untuk mengambil mulai dari index 1

Line13: memunculkan output namaterbalik

### SOAL 3

```
In [ ]: namafile= input("Masukkan nama file: ")
file = open(namafile, 'r')
baca_baris = file.readlines()
file.close()
dict = {}
for line in baca_baris:
    if line.startswith("From") and line.endswith("2008\n"):
        jam = line.split()[5].split(":")[0]
        dict[jam] = dict.get(jam, 0) + 1
for key, val in sorted(dict.items()):
    print(key, val)
```

```
Masukkan nama 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
```

Line1: membuat input untuk file yang mau dipakai



Line2: membuka file lalu membaca file

Line3: membaca semua baris dengan fungsi readlines

Line4: menutup file setelah dibaca

Line5: membuat dictionary kosong

Line6: for line in baca\_baris:: Melakukan looping setiap baris dalam daftar baca\_baris.

Line7: Memeriksa apakah dimulai dengan string "From" dan diakhiri dengan "2008\n"

Line8: Mengambil bagian jam dari baris dan dipecah menjadi kata-kata menggunakan spasi sebagai pemisah (split()), kemudian waktu dipecah lagi menggunakan titik dua sebagai pemisah untuk mendapatkan jam dan disimoan di variabel jam

Line9: Menghitung kemunculan jam. Jika jam sudah ada dimasukan ke kamus

Line10: Melakukan looping melalui item dalam kamus yang diurutkan berdasarkan kuncinya.

Line11: memunculkan output kunci (jam) dan nilai (kemunculan) yang telah diurutkan.

Link github:

<https://github.com/prasss020904/Laprak12-71230987.git>