



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71220895
Nama Lengkap	Drestanta Dipta Jalu Prakasya
Minggu ke / Materi	11 / Tipe Data Tuples

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%)

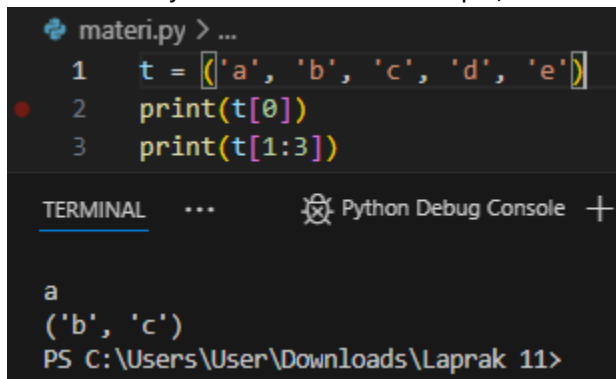
Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

MATERI 1

Meskipun tuple adalah struktur data yang mirip dengan daftar di Python, karakteristiknya berbeda. Berikut ringkasan tuple-nya:

- **Immutable:** Tuple adalah struktur data yang tidak dapat diubah; jika dibuat, elemen tersebut tidak dapat diubah. Namun sebuah tuple dapat "diganti" dengan membuat tuple baru dengan menggabungkan elemen yang diubah dengan elemen lain dari tuple yang sudah ada.
- **Hashable:** Tuple dapat dibandingkan dan dapat di-hash, sehingga dapat dimasukkan dalam daftar sebagai kunci dalam kamus Python.
- **Sintaks Penulisan:** Tuple dapat ditulis dengan atau tanpa tanda kurung. Jika ada satu elemen dalam tuple, maka harus ditambahkan koma setelah elemen tersebut agar tuple tersebut tidak dianggap sebagai tipe data lain.
- **Constructor:** Tuple memiliki konstruktor tuple() yang dapat membuat tuple kosong atau mengubah urutan menjadi tuple, seperti string, daftar, atau tuple.
- **Akses Elemen:** Anda dapat mengakses elemen tuple dengan menggunakan indeks mulai dari 0, dan pemotongan dapat mengakses rentang nilai elemen.

Untuk menunjukkan indeks elemen tuple, tanda kurung kotak [] digunakan:



```
materi.py > ...
1  t = ('a', 'b', 'c', 'd', 'e')
2  print(t[0])
3  print(t[1:3])

TERMINAL  ...  Python Debug Console  +

a
('b', 'c')
PS C:\Users\User\Downloads\Laprak 11>
```

Karena tuple tidak dapat diubah, usaha untuk mengubahnya akan menghasilkan error.

```
materi.py > ...
1  t = ('a', 'b', 'c', 'd', 'e')
2  t[0] = 'A'

TERMINAL  ...  Python Debug Console  + - [ ] [x] ... ^ x

Traceback (most recent call last):
  File "c:\Users\User\Downloads\Laprak 11\mater_i.py", line 3
, in <module>
    t[0] = 'A'
    ~^^^
TypeError: 'tuple' object does not support item assignment
PS C:\Users\User\Downloads\Laprak 11>
```

Element pada tuple tidak dapat dirubah tetapi dapat diganti dengan elemen lain.

```
materi.py > ...
1  t = ('a', 'b', 'c', 'd', 'e')
2  t = ('A',) + t[1:]
3  print(t)

TERMINAL  ...  Python Debug Console  +

('A', 'b', 'c', 'd', 'e')
PS C:\Users\User\Downloads\Laprak 11>
```

MATERI 2

Untuk tuple dan struktur data sekuensial lainnya, seperti daftar, kamus, dan kumpulan, operator perbandingan membandingkan elemen pertama dari setiap struktur data. Apabila terdapat persamaan maka elemen selanjutnya akan dibandingkan. Sampai ada perubahan, proses terus berjalan. Fungsi sort Python juga beroperasi dengan cara yang sama. Pertama, elemen diurutkan berdasarkan elemen pertama; terkadang, elemen kedua juga diurutkan, dan seterusnya. Metode ini dikenal dengan nama DSU (Decorate, Sort, Undecorate):

- Decorate: Urutan dibuat menjadi daftar tuple dengan satu atau lebih kunci pengurutan di depan elemennya.
- Sort: Fungsi sortir, yang merupakan fungsi bawaan Python, digunakan untuk mengurutkan daftar tuple.
- Undecorate: Struktur data sekuensial mengambil elemen yang telah diurutkan.

Contoh penggunaan pengurutan untuk mengurutkan kata dalam suatu kalimat dari yang terpanjang ke yang terpendek. Hasilnya adalah daftar kata yang diurutkan dari yang terpanjang hingga yang terpendek:

```
materi.py > ...
1  kalimat = 'but soft what light in yonder window breaks'
2  dafkata = kalimat.split()
3  t = list()
4  for kata in dafkata:
5      t.append((len(kata), kata))
6  t.sort(reverse=True)
7  urutan = list()
8  for length, kata in t:
9      urutan.append(kata)
10 print(urutan)
```

PROBLEMS OUTPUT TERMINAL PORTS

```
['yonder', 'window', 'breaks', 'light', 'what', 'soft', 'but', 'in']
PS C:\Users\User\Downloads\Laprak 11>
```

MATERI 3

Penugasan Tuple, fitur unik Python, memungkinkan Anda menetapkan lebih dari satu variabel secara berurutan dalam satu pernyataan. Misalnya, Anda dapat menetapkan elemen ke variabel x dan y dalam satu pernyataan jika Anda memiliki urutan seperti daftar.

```
materi.py > ...
1  m = ['have', 'fun']
2  x, y = m
3  print(x) # Output: 'have'
4  print(y) # Output: 'fun'

TERMINAL  ...  Python Debug Console

have
fun
PS C:\Users\User\Downloads\Laprak 11>
```

Misalnya, Python akan menafsirkan tugas tuple:

```
materi.py > ...
1  m = ['have', 'fun']
2  x = m[0]
3  y = m[1]
4  print(x)
5  print(y)

TERMINAL  ...  Python Debug Console

have
fun
PS C:\Users\User\Downloads\Laprak 11>
```

Untuk menunjukkan tuple, Anda juga dapat menggunakan tanda kurung:

```
materi.py > ...
1  m = ['have', 'fun']
2  (x, y) = m
3  print(x)
4  print(y)

TERMINAL  ...  Python Debug Console

have
fun
PS C:\Users\User\Downloads\Laprak 11>
```

Misalnya, program dapat mengelompokkan alamat email menjadi domain dan nama pengguna:

```
materi.py > ...
1  email = 'didanendya@ti.ukdw.ac.id'
2  username, domain = email.split('@')
3  print(username)
4  print(domain)
```

TERMINAL ... Python Debug Console + v □

```
didanendya
ti.ukdw.ac.id
PS C:\Users\User\Downloads\Laprak 11>
```

Fitur ini membuat penanganan data terstruktur dengan Python lebih mudah.

MATERI 4

Untuk mengembalikan daftar nilai tuple yang terdiri dari pasangan nilai kunci, kamus menggunakan metode yang disebut "item":

```
materi.py > ...
1 d = {'a':10, 'b':1, 'c':22}
2 t = list(d.items())
3 print(t)

TERMINAL ... Python Debug Console

[('a', 10), ('b', 1), ('c', 22)]
PS C:\Users\User\Downloads\Laprak 11>
```

Daftar tuple adalah urutan yang dapat diurutkan meskipun nilai dalam kamus tidak diurutkan. Hasilnya, kita dapat mengurutkannya:

```
materi.py > ...
1 d = {'a':10, 'b':1, 'c':22}
2 t = list(d.items())
3 t.sort()
4 print(t)

TERMINAL ... Python Debug Console

[('a', 10), ('b', 1), ('c', 22)]
PS C:\Users\User\Downloads\Laprak 11>
```

Dalam hal ini, daftar diurutkan menaik berdasarkan kunci dan nilai.

MATERI 5

Multi-penugasan dengan kamus menggabungkan item, penetapan tuple, dan loop untuk mengulangi kunci dan nilai dari kamus dalam satu loop. Dalam loop ini, dua variabel diiterasi: key dan val. `items()` mengembalikan daftar tuple yang berisi pasangan nilai kunci, dan kunci serta nilai maju ke pasangan nilai kunci berikutnya dalam kamus (masih dalam urutan hash). Dengan menggabungkan metode-metode ini, kita dapat mencetak isi kamus yang diurutkan berdasarkan nilai yang disimpan di setiap pasangan nilai kunci. Pertama, kita membuat daftar tuple yang setiap tupelnya terdiri dari (nilai, kunci). Metode `items()` menyediakan daftar tuple (nilai, kunci) dan mengurutkannya berdasarkan nilainya, bukan kuncinya. Setelah daftar tuple kunci nilai dibuat, kami mengurutkan daftar dalam urutan terbalik dan mencetak daftar yang baru diurutkan. Berikut ini contohnya:

```
materi.py > ...
1  d = {'a':10, 'b':1, 'c':22}
2  l = list()
3  for key, val in d.items():
4      l.append((val, key))
5
6  print(l)
7  |
8  l.sort(reverse=True)
9  print(l)
```

Python Debug Console

```
[(10, 'a'), (1, 'b'), (22, 'c')]
[(22, 'c'), (10, 'a'), (1, 'b')]
PS C:\Users\User\Downloads\Laparak 11>
```

Dengan mengkompilasi daftar tuple dengan nilai sebagai elemen pertama, mengurutkan daftar tuple menjadi mudah dan mengurutkan konten kamus berdasarkan nilainya.

MATERI 6

Di bawah ini adalah program yang menghitung dan menampilkan sepuluh kata yang paling sering muncul dalam teks dari Adegan 2 Romeo dan Juliet.

```
materi.py > ...
1  import string
2  fhand = open('romeo-full.txt')
3  counts = dict()
4  for line in fhand:
5      line = line.translate(str.maketrans('', '', string.punctuation))
6      line = line.lower()
7      words = line.split()
8      for word in words:
9          if word not in counts:
10             counts[word] = 1
11          else:
12             counts[word] += 1
13  lst = list()
14  for key, val in list(counts.items()):
15      lst.append((val, key))
16  lst.sort(reverse=True)
17  for key, val in lst[:10]:
18      print(key, val)

PROBLEMS  OUTPUT  TERMINAL  PORTS

61 i
42 and
40 romeo
34 to
34 the
32 thou
32 juliet
30 that
29 my
24 thee
PS C:\Users\User\Downloads\Laprak 11>
```

Penjelasan:

- Alat ini dapat membuka file teks, membersihkan tanda baca, mengubah huruf menjadi huruf kecil, dan memecah teks menjadi kata.
- Counts menghitung berapa kali setiap kata muncul di kamus.
- Selanjutnya, kamus diubah menjadi daftar tuple yang diurutkan berdasarkan jumlah kata yang muncul dalam urutan terbalik.
- Program ini mencetak sepuluh kata yang paling sering muncul, serta berapa kali kata tersebut muncul.

Output program menunjukkan sepuluh kata yang paling sering muncul, disertai dengan berapa kali kata tersebut muncul. Program ini menggunakan 19 baris kode Python untuk memproses data yang kompleks, menunjukkan keunggulan Python dalam pemrosesan dan analisis data.

MATERI 7

Dalam kamus, string digunakan sebagai kunci:

```
materi.py > ...
1 last = 'nendya'
2 first = 'dida'
3 number = '088112266'
4 directory = dict()
5 directory[last, first] = number
6
7 for last, first in directory:
8     print(first, last, directory[last,first])
```

PROBLEMS OUTPUT TERMINAL PORTS

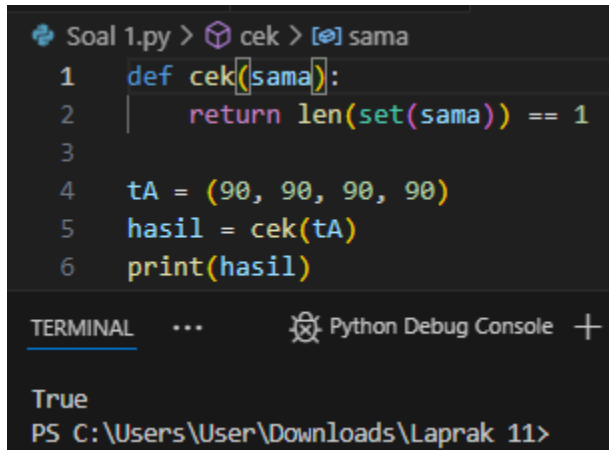
dida nendya 088112266
PS C:\Users\User\Downloads\Laprak 11>

Dalam contoh ini, kamus direktori menggunakan tupel (last, first) untuk memetakan pasangan nama belakang dan nama depan ke nomor telepon. Kemudian, kami menggunakan perulangan for untuk mencetak nama dan nomor telepon yang sesuai.

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



```
Soal 1.py > cek > sama
1 def cek(sama):
2     return len(set(sama)) == 1
3
4 tA = (90, 90, 90, 90)
5 hasil = cek(tA)
6 print(hasil)

TERMINAL  ...  Python Debug Console  +
True
PS C:\Users\User\Downloads\Laprak 11>
```

Kode ini memastikan apakah semua anggota tuple identic:

1. Def cek(sama): definisi fungsi cek yang memiliki satu parameter, yaitu sama.
2. Fungsi pemeriksaan berisi return len(set(sama)) == 1. Fungsi ini menggunakan set() untuk mengubah tuple yang sama menjadi satu set; ketika sebuah tuple diubah menjadi satu set, semua duplikat akan dihapus. Selanjutnya, kita dapat menghitung panjang himpunan dengan len(). Jika panjang himpunan adalah 1, berarti semua anggota dalam tuple asli adalah sama, sehingga fungsi akan mengembalikan True jika tidak. Jika panjang himpunan tidak sama, maka fungsi akan mengembalikan False.
3. Contoh tuple dengan tA = (90, 90, 90, 90) digunakan untuk menentukan apakah semua anggotanya identic.
4. Fungsi pemeriksaan dengan tA disebut result = check(tA). Fungsi ini mengembalikan Benar atau Salah tergantung pada apakah semua anggota tuple tA sama.
5. Print(result) mencetak hasil pemeriksaan, yang bisa benar jika semua anggota tuple sama atau salah jika tidak.

SOAL 2

```
Soal 2.py > [E] nama_depan
1 data = ('Matahari Bhakti Nendya', '22064091', 'Bantul, DI Yogyakarta')
2
3 print("Data:", data)
4 print("NIM :", data[1])
5 print("NAMA :", data[0])
6 print("ALAMAT :", data[2])
7 print("NIM:", tuple(data[1]))
8
9 nama_depan = tuple(data[0].split()[0].lower())
10 print("NAMA DEPAN:", nama_depan)
11 print("NAMA TERBALIK:", tuple(data[0].split()[::-1]))

PROBLEMS OUTPUT TERMINAL PORTS

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: ('m', 'a', 't', 'a', 'h', 'a', 'r', 'i')
NAMA TERBALIK: ('Nendya', 'Bhakti', 'Matahari')
PS C:\Users\User\Downloads\Laparak 11> |
```

Program ini menggunakan tuple dengan nama data yang berisi informasi data pribadi. Ini mencetak NIM, Nama, dan Alamat dari tuple data, mengubahnya menjadi tuple(data[1]), lalu mengubah nama depan menjadi tuple dengan memisahkan nama depan dari data[0], mengubahnya menjadi huruf kecil, dan lalu mengubahnya menjadi Tuple. Terakhir, program ini menggunakan tuple dengan nama data yang berisi informasi data pribadi. Setiap output dicetak dengan format yang ditunjukkan pada contoh output.

SOAL 3

```
Soal 3.py > ...
1  try:
2      with open(input("Enter a file name: "), 'r') as file:
3          hour_count = {}
4          for line in file:
5              if line.startswith('From '):
6                  hour = line.split()[5].split(':')[0]
7                  hour_count[hour] = hour_count.get(hour, 0) + 1
8          for hour, count in sorted(hour_count.items()):
9              print(hour, count)
10 except FileNotFoundError:
11     print("File not found.")
```

PROBLEMS OUTPUT TERMINAL PORTS

```
Enter a file name: 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
PS C:\Users\User\Downloads\Laparak 11>
```

Pengguna akan diminta memasukkan nama file mbox-short.txt dalam program. Setelah itu, program akan membuka file tersebut dan membaca setiap barisnya. Setiap baris yang dimulai dengan kata "Dari" akan diproses. Program akan membagi baris menjadi kata-kata dan menghitung waktu (jam) kata keenam. Selanjutnya, program akan mengubah jumlah pesan email yang diterima selama jam tersebut di kamus jumlah_jam. Jika jam belum ada dalam kamus, nilai awalnya akan ditambah dengan nilai 0 sebelum ditambahkan.

Setelah membaca file, kamus hour_count akan diurutkan berdasarkan jam. Kemudian, program akan mencetak setiap jam dan jumlah pesan email yang diterima selama jam tersebut. Harap pastikan file mbox-short.txt berada di direktori yang sama dengan program Python yang sedang berjalan, atau berikan path lengkap ke file tersebut saat dimintai nama file. Distribusi jam pesan email yang diterima dalam sehari akan ditampilkan di output.