



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	<71230981>
Nama Lengkap	<Deo Dewanto >
Minggu ke / Materi	10/Tipe Data Dictionary

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

Dictionary merupakan pemetaan antara sekumpulan indeks(yang disebut kunci) dan sekumpulan nilai. Setiap kunci dari dictionary memetakan sebuah nilai. Setiap kunci dan nilai pada dictionary disebut juga sebagai pasangan nilai kunci(key-value pair) atau ada juga beberapa yang menyebutnya item.

Dictionary memiliki kesamaan dengan list, tetapi pada dictionary ini bersifat umum. Dalam list, indeks harus berupa integer sedangkan pada dictionary indeks dapat berupa apapun.

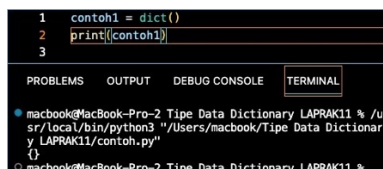
Dictionary selalu memiliki anggota yang terdiri dari pasangan, yaitu kunci:nilai. Pada bagian kunci ini, harus bersifat unik, karena tidak boleh sama dengan kunci lainnya.

Sebagai contoh, kita ingin mengembangkan kamus yang memetakan antara bahasa Indonesia ke bahasa inggris. Jadi bi akita katakana bahwa yang menjadi kunci dan nilai adalah data string .Diasumsikan setiap kata bahasa indonesia memiliki arti kata dalam bahasa inggris.

Pada dictionary, python memiliki fungsi bawaan yang biasa disebut built-in. Fungsi tersebut yaitu dict(). Fungsi dict() ini digunakan untuk membuat dictionary baru yang kosong dengan variable yang bebas. Tetapi perlu diingat bahwa ada fungsi built-in, maka penggunaanya perlu menghindari variable bernama dict. Seperti contoh dibawah ini.

Source Code dan Output:

```
1  contoh1 = dict()
2  print(contoh1)
3
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"

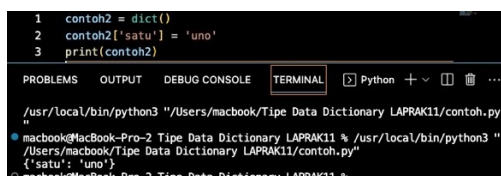
{}

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %

Tanda kurung kurawal seperti contoh diatas ini, merupakan contoh mempresentasikan sebuah dictionary yang kosong. Untuk menambahkan item pada dictionary, dapat menggunakan kurung kotak [ ].

Source Code dan Output:

```
1  contoh2 = dict()
2  contoh2['satu'] = 'uno'
3  print(contoh2)
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python + v [] ...

/usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"

{'satu': 'uno'}

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %

Pada contoh diatas ini, merupakan contoh dari Dictionary yaitu membuat item yang yang memetakan dari kunci 'satu' ke 'dua'. Dan akan dihasilkan output pasangan antara kunci dan nilai.

Format output seperti diatas juga digunakan untuk format input. Misalkan kita ingin membuat dictionary baru dengan tiga item dan melakukan panggilan print, yang akan didapat berupa keseluruhan data dari dictionary tersebut.

Source Code dan Output:

```
1  contoh3 = {'chuck' : 1, 'annie' : 42, 'jan' : 100}
2  print(contoh3['chuck'])
3
```

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 /Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py

Terlihat pada contoh diatas, penggunaan kata kunci 'chuck' selalu dipasangkan dengan '1'. Jadi urutan pada dictionary tidak menjadi masalah. Tetapi jika kita menggunakan kunci yang tidak ada, maka hasilnya akan seperti contoh dibawah ini.

Source Code dan Output:

```
1  contoh3 = {'chuck' : 1, 'annie' : 42, 'jan' : 100}
2  print(contoh3['chucky'])
3
```

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 /Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py

Traceback (most recent call last):  
File "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py", line 2, in <module>  
print(contoh3['chucky'])  
KeyError: 'chucky'

Pada dictionary, fungsi len bisa berfungsi. Fungsi len digunakan untuk mengembalikan jumlah pasangan nilai kunci.

Source Code dan Output:

```
1  contoh4 = {'chuck' : 1, 'annie' : 42, 'jan' : 100}
2  print(len(contoh4))
3
```

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 /Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py

Selain itu, ada juga operator **in** yang digunakan untuk mengembalikan nilai true atau false sesuai dengan kunci yang ada pada dictionary. Contoh

Source Code dan Output:

```
1  contoh5 = {'chuck' : 1, 'annie' : 42 , 'jan' : 100}
2  print('chuck' in contoh5)
3  print('1' in contoh5)
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** Python +

```
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 /Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py
True
False
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %
```

Dan yang terakhir yaitu metode **values**. Metode values ini digunakan untuk mengetahui nilai yang ada pada dictionary. Metode ini juga mengembalikan nilai sesuai dengan tipe datanya dan dikonversi kedalam list dan digunakan dalam operator **in**. Contoh

```
1  contoh6 = {'chuck' : 1, 'annie' : 42 , 'jan' : 100}
2
3  contoh6_vals = list(contoh6.values())
4  print(42 in contoh6_vals)
5
```

PROBLEMS OUTPUT **TERMINAL** ... Python + - [] ... ^ >

Traceback (most recent call last):

```
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 /Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py
True
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %
```

## Dictionary Sebagai Set Penghitung(Conters)

Perhitungan atau komputasi model dictionary dianggap lebih praktis dikarenakan kita tidak perlu mengetahui huruf mana yang akan muncul. Berikut merupakan beberapa contoh perhitungan model dictionary.

Source Code dan Output:

```
1  word = 'brontosaurus'
2  d = dict()
3  for c in word:
4      if c not in d:
5          d[c] = 1
6      else:
7          d[c] + 1
8  print(d)
9
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

```
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 /Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py
{'b': 1, 'r': 1, 'o': 1, 'n': 1, 't': 1, 's': 1, 'a': 1, 'u': 1}
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %
```

Dictionary memiliki metode get yang dimana berfungsi untuk mengambil kunci dan nilai default. Jika kunci muncul pada dictionary, maka akan mengembalikan nilai yang sesuai; jika tidak, maka akan mengembalikan nilai default. Berikut merupakan contoh.

Source Code dan Output :

```
1 counts = {'chuck' : 1, 'anna' : 42, 'jan' : 100}
2 print(counts.get('jan', 0))
3 print(counts.get('tin', 0))
4
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

```
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"
100
0
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %
```

## Dictionary dan File

Pada materi ini, biasanya kegunaan dictionary adalah untuk menghitung jumlah kata yang ada pada sebuah file.txt. Sebagai contoh seperti berikut ini.

Source Code dan Output:

```
1 fname = input('Enter the file name: ')
2 try:
3     fhand = open(fname)
4 except:
5     print('File cannot be opened:', fname)
6     exit()
7
8 counts = dict()
9 for line in fhand:
10     words = line.split()
11     for word in words:
12         counts[word] = 1
13     else:
14         counts[word] += 1
15 print(counts)
16
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL**

```
/usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"
macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"
Enter the file name: romeo.txt
{'But': 1, 'soft': 1, 'what': 1, 'light': 1, 'through': 1, 'yonder': 1, 'window': 1, 'breaks': 2, 'It': 1, 'is': 1, 'the': 1, 'east': 1, 'and': 1, 'Juliet': 1, 'sun': 1, 'Arise': 1, 'fair': 1, 'kill': 1, 'envious': 1, 'moon': 2, 'Who': 1, 'already': 1, 'sick': 1, 'pale': 1, 'with': 1, 'grief': 2}
```

## Looping dan Dictionary

Pada dictionary, looping statement for itu akan bekerja dengan cara menelusuri kunci yang ada didalamnya. Looping ini akan melakukan pencetakan setiap kunci sesuai dengan hubungan nilainya. Seperti contoh-contoh berikut ini.

Source Code dan Output :

```
1 counts = { 'chuck' : 1 , 'bambang' : 42 , 'jo' : 100}
2 for key in counts :
3     print(key, counts[key])
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %  
/usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"  
chuck 1  
bambang 42  
jo 100  
○ macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %

Source Code dan Output :

```
1 counts = { 'chuck' : 1 , 'bambang' : 42 , 'jo' : 100}
2
3 lst = list(counts.keys())
4 print(lst)
5 lst.sort()
6 for key in lst:
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

● macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 % /usr/local/bin/python3 "/Users/macbook/Tipe Data Dictionary LAPRAK11/contoh.py"  
['chuck', 'bambang', 'jo']  
bambang 42  
chuck 1  
jo 100  
○ macbook@MacBook-Pro-2 Tipe Data Dictionary LAPRAK11 %

## 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

Source Code dan Output :

```
#latihan 1
dict_nilai = {1:10, 2:20, 3:30, 4:40, 5:50, 6:60}
nomor = 1
print('key', 'values', 'items')
for key, values in dict_nilai.items():
    print(key, ' ', values, ' ', nomor)
    nomor += 1
```

✓ 0.0s Python

key	values	items
1	10	1
2	20	2
3	30	3
4	40	4
5	50	5
6	60	6

### SOAL 2

Source Code dan Output :

```
#latihan 2
datalist1 = ['red', 'green', 'blue']
datalist2 = ['#FF0000', '#008000', '#0000FF']
d = dict()
if datalist1 != datalist2:
    print('Value Error!')
for key, value in zip(datalist1, datalist2):
    d[key] = value
print(d)
```

[5] ✓ 0.0s Python

... Value Error!  
{'red': '#FF0000', 'green': '#008000', 'blue': '#0000FF'}

### SOAL 3

Source Code dan Output :

```
#latihan 3
def bacalog(filename):
    email_ = {}
    pesan = 0

    with open(filename, 'r') as file:
        for line in file:
            if line.startswith('From '):
                email = line.split()[1]
                email_[email] = email_.get(email, 0) + 1
                pesan += 1

    return email_, pesan

def main():
    bacalog2=input("Masukkan nama file: ")
    email_, pesan = bacalog(bacalog2)

    print(email_)

if __name__ == "__main__":
    main()

✓ 13.2s Python
```

{'stephen.marquard@uct.ac.za': 2, 'louis@media.berkeley.edu': 3, 'zqian@umich.edu': 4, 'rjlowe@iupui.edu': 2

## SOAL 4

Source Code dan Output :

```
#latihan 4
def menghitung_email(email):
    return email.split('@')[-1]

def menghitung_email1(filename):
    counts = {}

    with open(filename, 'r') as file:
        for line in file:
            if line.startswith('From '):
                email = line.split()[1]
                awal = menghitung_email(email)
                counts[awal] = counts.get(awal, 0) + 1

    return counts

filename = input("Masukkan nama file: ")
counts = menghitung_email1(filename)

print(counts)

[8] ✓ 10.8s Python
```

... {'uct.ac.za': 6, 'media.berkeley.edu': 4, 'umich.edu': 7, 'iupui.edu': 8, 'caret.cam.ac.uk': 1, 'gmail.com':

LINK GITHUB : <https://github.com/Deodewanto07/LAPORAN11.git>