



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230985</b>
<b>Nama Lengkap</b>	<b>TOMAS BECKET</b>
<b>Minggu ke / Materi</b>	<b>10 / LIST</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%)

### MATERI 1

List pada python adalah rangkaian nilai-nilai yang dapat diakses menggunakan satu nama tunggal. Perbedaan yang bisa dilihat yaitu string merupakan sebuah serangkaian karakter-karakter, sedangkan list dapat berisi karakter, integer, float, maupun tipe data lainnya.

Contohnya:

```
nilai_ujian = [80,75,70,90,81,84,92,71,65,80,70]
nama_pahlawan = ['Sukarno', 'Diponegoro', 'Jend. Sudirman', 'Cut Nya Dhien']
nilai_campuran = ['Javascript', 20, 34.4, True]
list_dalam_list = [23, [22, 20], 45]
```

### MATERI 2

List sendiri juga memiliki perbedaan dengan string, yaitu list bersifat mutable, sedangkan string bersifat immutable. Matable artinya nilai dapat diubah secara langsung sedangkan immutable nilainya tidak dapat diubah atau sudah menjadi satu.

Contoh dibawah merupakan list yang terdiri dari sebuah String, integer, dan boolean. Ketika kita meminta untuk memprint nama, maka kita dapat mencari index berapa yang akan di print untuk memunculkan ouput Nama pada list tersebut, yaitu berletakan pada index 0, dan pada sebuah list, dapat mengubah nilai index yang dimiliki, misal pada index 0 berisikan nama 'asep', maka nilai tersebut bisa digantikan dengan mencari list index 0 dan mengubahnya menjadi 'Dimas' dan melakukan print, sehingga string 'asep' digantikan dengan 'Dimas'.

```
1 list_nilai = ['asep', 100, True]
2 print(f>Nama : {list_nilai[0]}, Nilai : {list_nilai[1]})
3
4 list_nilai[0] = 'Dimas'
5 print(f>Nama : {list_nilai[0]}, Nilai : {list_nilai[1]})
6
7 list_nilai[1] = 700
8 print(f>Nama : {list_nilai[0]}, Nilai : {list_nilai[1]})
```

Bahkan kita juga bisa mengganti sebuah nilai pada index 1 untuk mengubah nilai 100 menjadi 700, dst. Hal tersebut sangat berfungsi ketika kita ingin mengganti sebuah nilai dibandingkan menggunakan sebuah string.

### MATERI 3

Berikut ada pun contoh source code ketika sebuah list menggunakan pertambahan pada programnya, sehingga ketika hasil = lista + listb. Maka ketika memprint hasil akan memunculkan Ouput [1, 2, 3, 10, 11, 12] sesuai dengan list yang dimiliki dari masing-masing kedua variable tersebut. Dan ketika Hasil\*3 maka akan memunculkan angka 1, 2, 3, 10, 11, 12 sebanyak 3 kali pada hasil ouputan

```
1  lista = [1, 2, 3]
2  listb = [10, 11, 12]
3  hasil = lista + listb
4
5  print(hasil)
6
7  print(hasil*3)
8
9  asks = [3, 4, 5, 9, 10]
10 print(asks[3]) # DARI DEPAN
11 print(asks[-2]) # DARI BELAKANG
```

Adapun cara untuk memprint list dari belakang atau pun depan dengan mengambil indexnya pada sebuah list. Misal kita mengambil index 3 maka kita akan mengambil angka 9 dimulai dari index 0, 1, 2, 3[9]. Sedangkan jika mengambil menggunakan index -2 maka akan dimulai dari belakang seperti -1[10], -2[9].

### MATERI 4

**APPEND.** Fungsi append() adalah sebuah metode yang digunakan pada objek daftar (list) dalam Python. Fungsi ini biasa digunakan untuk menambahkan elemen baru ke dalam daftar. Elemen yang ditambahkan akan ditempatkan di akhir daftar.

```
5  lst = []
6
7  lst.append(1)
8  lst.append('Satu')
9  lst.append(True)
10
11 print(lst)
```

Misal pada kode tersebut dibuat sebuah variabel lst = [] kosong, dan menggunakan fungsi append(1), append('satu'), append(True). Maka ketika melakukan print(lst) maka hasil output yang muncul yaitu [1, 'satu', True] pada sebuah lst yang awalnya kosong.

**SORT.** Fungsi `sort()` adalah metode yang digunakan pada objek daftar (list) dalam Python untuk mengurutkan elemen-elemennya. Metode ini mengurutkan daftar itu sendiri dan mengubah urutan elemen-elemennya menjadi urutan yang terurut.

```
23 lst1 = [1, 6, 99, 12, 423, 8, 0]
24 lst1.sort()
25 print(lst1)
```

Di program ini akan mengurutkan angka dimulai dari yang kecil hingga terbesar ketika memanggil fungsi `lst1.sort()`

Adapun beberapa metode lainnya yang sering digunakan pada sebuah list, yaitu untuk menghapus elemen pada list dapat menggunakan `pop` ataupun `del` atau menggunakan `remove`.

**POP.** Fungsi `pop()` adalah metode yang digunakan pada objek daftar (list) dalam Python untuk menghapus dan mengembalikan elemen dari indeks yang diberikan. Metode ini juga dapat digunakan tanpa argumen untuk menghapus dan mengembalikan elemen terakhir dari daftar.

```
28 lst1 = [0, 1, 2, 3, 4, 5]
29 lst1.pop(3)
30 del lst1[3]
31 print(lst1)
```

Contoh pada source code diatas, `lst1.pop(3)` maka ketika fungsi tersebut dipanggil, `lst1` akan dicari yang bernilai angka 3 dan akan di hapus. berbeda dengan `del lst1[3]`, ketika fungsi tersebut dipanggil maka kita akan menentukan index berapa yang akan dihapus.

Contoh lain yang ada pad module M-10

**del :** Jika indeks elemen sudah diketahui, dan tidak memerlukan nilai elemen yang dihapus.

```
>>> huruf = ['a', 'b', 'c', 'd']
>>> del huruf[2]
>>> huruf
['a', 'b', 'd']
```

```
>>> huruf = ['a', 'b', 'c', 'd']
>>> del huruf[1:3]
```

**remove :** Jika yang diketahui ialah nilai elemen yang akan dihapus.

```
>>> huruf = ['a', 'b', 'c', 'd']
>>> huruf.remove("c")
>>> huruf
['a', 'b', 'd']
```

**REVERSE.** Fungsi `reverse()` adalah metode yang digunakan pada objek daftar (list) dalam Python untuk membalikkan urutan elemen-elemennya. Metode ini akan mengubah urutan elemen-elemen dalam daftar tersebut sehingga elemen terakhir menjadi elemen pertama, elemen kedua menjadi elemen kedua terakhir, dan seterusnya.

```
57 lst1 = [1, 2, 3]
58 lst2 = lst1 [:]
59
60 lst1.reverse()
```

Dengan adanya fungsi reverse pada source code diatas, maka pembuatan variabel baru yaitu `lst2 = lst1 [:]` ketika dilakukan `lst1.reverse()` maka pada saat memprint hasil output, angka yang muncul yaitu dimulai dari index paling belakang.

**Fungsi COPY.** Pembuatan list juga dapat menggunakan sebuah perulangan dimana contohnya seperti source code dibawah membedakan dengan membuat sebuah `lst1` normal, `lst2` diambil dari `lst1`, dan list menggunakan perulangan `for i in range(4)` sehingga angka yang akan append pada list yaitu angka 1 hingga 4.

```
44 lst1 = [0, 1, 2, 3]
45 lst2 = lst1[0:4]
46
47 print(lst1)
48 print(lst2)
49
50 list = []
51 for i in range(4):
52     list.append(i)
53 print(list)
```

## MATERI 5

### Perbedaan List dan String

List dan string adalah dua tipe data yang berbeda dalam Python, dan keduanya memiliki karakteristik yang unik.

String adalah urutan karakter. Ini adalah tipe data yang tidak dapat diubah (immutable) di Python, yang berarti setelah string dibuat, kita tidak dapat mengubah karakter individualnya. Sedangkan List adalah urutan elemen yang dapat berupa tipe data apa pun, termasuk tipe data campuran. List adalah tipe data yang dapat diubah (mutable) di Python, yang berarti kita dapat menambah, menghapus, atau memodifikasi elemen-elemennya setelah list dibuat.

Immutability vs. Mutability, seperti yang dijelaskan sebelumnya Karena string tidak dapat diubah, kita tidak bisa menambahkan, menghapus, atau mengubah karakter dalam string. Operasi apa pun pada string akan menghasilkan string baru. Namun pada List dapat diubah, yang berarti kita dapat

menambah, menghapus, atau mengubah elemen-elemennya. Operasi pada list dapat memodifikasi list asli.

Representasi Literal:

- String: String dapat direpresentasikan dengan tanda kutip tunggal ('...') atau tanda kutip ganda ("...") di sekitar karakter atau teksnya.
- List: List direpresentasikan dengan tanda kurung siku ([...]) di sekitar elemen-elemennya, dipisahkan oleh koma.

```
1  # PERBEDAAN LIST DAN STRING
2  a = 'ini ada14h string'
3  b = ['ini', 'ada', 1, 4, 'string']
4
5
6  #
7  def hapus_index_pertama(list):
8      |   return list[1:]
9
10 list = [1, 2, 3, 4]
11 print(hapus_index_pertama(list))
```

Contoh pada source code diatas, variabel a merupakan sebuah string, sedangkan variabel b merupakan sebuah list dan kedua tersebut memiliki fungsi yang berbeda.

### Mencari Kata Depan/Belakang menggunakan SPLIT

Jika Ingin memprint Kata belakang atau depan menggunakan fungsi Split dan index yang diambil

Misal variabel a memiliki sebuah string 'abc bca', maka kita menambah variabel baru dengan fungsi nama.split() untuk membagi kalimat 'abc bca' menjadi sebuah list ['abc', 'bca'] sehingga dari sini kita bisa menentukan mana yang akan dijadikan kata depan dan mana yang akan dijadikan sebagai kata belakang.

```
1  nama = 'abc bca'
2  nama1 = nama.split()
3
4  nama_depan = nama1[0]
5  nama_belakang = nama1[1]
6
7  print(nama_depan)
8  print(nama_belakang)
```

## BAGIAN 2: LATIHAN MANDIRI (60%)

### SOAL 1

#### 1.1 SOURCE CODE

```
1  def nilai_terbaik(nilai):
2      if isinstance(nilai, list):
3          nilai.sort()
4          nilai.reverse()
5          tertinggi = nilai[0:3]
6          return tertinggi
7      else:
8          return None
9
10 print(nilai_terbaik([91, 82, 95, 73, 85, 40, 0]))
11 print(nilai_terbaik([43, 53, 12, 19, 0, 98]))
12 print(nilai_terbaik("OKE"))
```

#### 1.2 PENJELASAN

Fungsi ini akan memanggil nilai 3 tertinggi dan diurutkan dari yang terbesar hingga terkecil, dimana menggunakan def function nilai\_terbaik(nilai): Mendefinisikan sebuah fungsi bernama nilai\_terbaik yang menerima satu parameter bernama nilai.

if isinstance(nilai, list): akan memeriksa apakah parameter nilai yang diberikan merupakan sebuah list. Fungsi isinstance() digunakan untuk memeriksa apakah objek adalah instance dari suatu kelas atau tipe data tertentu.

nilai.sort(): Mengurutkan list nilai secara ascending (dari nilai terkecil ke terbesar).

nilai.reverse(): Membalikkan urutan elemen-elemen dalam list sehingga list menjadi terurut secara descending (dari nilai terbesar ke terkecil).

tertinggi = nilai[0:3]: Mengambil tiga elemen pertama dari list nilai, yang sekarang sudah terurut dari yang terbesar ke yang terkecil.

return tertinggi: Mengembalikan tiga nilai tertinggi dalam bentuk list.

else: return None: Jika parameter nilai bukanlah sebuah list, maka fungsi akan mengembalikan None.

#### 1.3 HASIL OUTPUT

```
[95, 91, 85]
[98, 53, 43]
None
```

## SOAL 2

### 2.1 SOURCE CODE

```
1  def angkaMaxdanMin():
2      angka = []
3
4      while True:
5          input_angka = input("Masukan data Angka menggunakan (,) atau Ketik 'done' untuk selesai : ")
6
7          if input_angka.lower() == 'done':
8              break
9
10         angka.extend(input_angka.split(','))
11
12         angka = [float(x) for x in angka]
13
14         if angka:
15             print("Nilai maks:", max(angka))
16             print("Nilai min:", min(angka))
17         else:
18             print("None")
19
20     angkaMaxdanMin()
```

### 2.2 PENJELASAN

Program ini akan menunjukkan nilai maks dan min yang dimasukan oleh user dalam sebuah list

Dan Fungsi pertama-tama membuat sebuah list kosong yang disebut angka.

Kemudian, menggunakan loop while True, yang dimana fungsi akan terus meminta pengguna untuk memasukkan angka. Dan pengguna dapat memasukkan angka secara terpisah menggunakan koma atau mengetikkan 'done' untuk menyelesaikan input.

Setiap kali pengguna memasukkan angka, fungsi akan memisahkan angka-angka tersebut menggunakan metode split(',') dan menambahkannya ke dalam list angka menggunakan metode extend().

Setelah pengguna menyelesaikan input dengan mengetikkan 'done', loop akan berhenti.

List angka kemudian diubah menjadi list angka floating point menggunakan list comprehension.

Fungsi kemudian memeriksa apakah list angka tidak kosong. Jika tidak kosong, fungsi akan mencetak nilai maksimum dan minimum dari list tersebut menggunakan fungsi max() dan min(). Jika list angka kosong, fungsi akan mencetak None.

### 2.3 HASIL OUTPUT

```
Masukan data Angka menggunakan (,) atau Ketik 'done' untuk selesai : 9, 10, 7, 8, 5
Masukan data Angka menggunakan (,) atau Ketik 'done' untuk selesai : done
Nilai maks: 10.0
Nilai min: 5.0
```



## SOAL 3

### 3.1 SOURCE CODE

```
1  with open('file.txt', 'r') as file:
2      sentences = file.readlines()[1:]
3
4  words_list = []
5  for sentence in sentences:
6      words = sentence.split()
7      words_list.extend(words)
8
9  print('Kata-kata dalam file:')
10 print(words_list)
11
12
13 with open('file.txt', 'a') as file:
14     file.write('\n')
15     file.write('====Kata Unik pada Isi Berita====\n')
16
17     for i in range(0, len(words_list), 12):
18         words_chunk = words_list[i:i+12]
19         file.write(str(words_chunk) + '\n')
```

```
file.txt
1  =====ISI BERITA=====
2  Korban luka akibat gempa bumi berkekuatan Magnitudo 6,3 yang mengguncang Jepang bertambah menjadi
3  sembilan orang, dan tidak ada laporan korban tewas. Gempa bumi ini memicu longsor pada ruas jalan
4  raya nasional, menyebabkan kerusakan pada saluran pipa air dan menumbangkan tiang lampu jalanan.
```

### 3.2 PENJELASAN

Source code diatas akan dimulai dengan membuka file 'file.txt' dalam mode baca ('r') menggunakan fungsi open().

Kemudian, setiap baris dari file tersebut dibaca dan dimasukkan ke dalam daftar sentences, mulai dari baris kedua ke atas. Hal ini dilakukan dengan menggunakan metode readlines() dan mengindeks daftar hasilnya mulai dari indeks 1 ([1:]) untuk mengabaikan baris pertama.

Setiap kalimat dalam sentences dipisahkan menjadi kata-kata menggunakan metode split() dan disimpan dalam daftar words\_list.

Proses ini dilakukan menggunakan loop for yang mengiterasi melalui setiap kalimat dalam sentences, membaginya menjadi kata-kata, dan menambahkannya ke dalam words\_list menggunakan metode extend().

Setelah semua kata dipisahkan dan disimpan dalam words\_list, kode mencetak kata-kata tersebut dengan menggunakan pernyataan print().

Dan ketika sudah disimpan, akan dibuat fungsi Kode kembali membuka file 'file.txt', namun kali ini dalam mode menambahkan ('a'), sehingga data baru akan ditambahkan ke akhir file.

Baris kosong ditambahkan ke file menggunakan file.write('\n'), yang akan membuat baris baru.

Baris berjudul '=====Kata Unik pada Isi Berita===== ' juga ditambahkan ke file menggunakan file.write().

Dan Kata-kata dalam words\_list kemudian dibagi menjadi potongan-potongan berukuran 12 kata menggunakan loop for yang mengiterasi melalui daftar kata-kata tersebut dengan langkah 12 (range(0, len(words\_list), 12)).

Setiap potongan kata-kata dituliskan ke file menggunakan file.write(), diikuti dengan baris baru untuk memisahkan setiap potongan.

### 3.3 HASIL OUTPUT

```
file.txt
1  =====ISI BERITA=====
2  Korban luka akibat gempa bumi berkekuatan Magnitudo 6,3 yang mengguncang Jepang bertambah menjadi
3  sembilan orang, dan tidak ada laporan korban tewas. Gempa bumi ini memicu longsor pada ruas jalan
4  raya nasional, menyebabkan kerusakan pada saluran pipa air dan menumbangkan tiang lampu jalanan.
5
6  =====Kata Unik pada Isi Berita=====
7  ['Korban', 'luka', 'akibat', 'gempa', 'bumi', 'berkekuatan', 'Magnitudo', '6,3', 'yang', 'mengguncang', 'Jepang', 'bertambah']
8  ['menjadi', 'sembilan', 'orang', 'dan', 'tidak', 'ada', 'laporan', 'korban', 'tewas.', 'Gempa', 'bumi', 'ini']
9  ['memicu', 'longsor', 'pada', 'ruas', 'jalan', 'raya', 'nasional,', 'menyebabkan', 'kerusakan', 'pada', 'saluran', 'pipa']
10 ['air', 'dan', 'menumbangkan', 'tiang', 'lampu', 'jalanan.']
```