Algorithm and Data Structures

Endang Wahyu Pamungkas, P.hD.

Topics

- Overview of Algorithms and Data Structures
- Fundamentals of Python
- Object Oriented Programming
- Array
- Linear List
- Search
- Sorting
- Stack and Queue
- Trees
- Graph

Study Contract

- The course will be delivered offline (75%) and online (25%).
- The online meeting could be either synchronous (Google Meet) or asynchronous (Open Learning).
- The maximum late for students is 15 minutes after the lecturer enters class.
- Students need to inform the lecture if they are unable to attend the class maximum 1 hour before the class.
- Please be on time when submitting your assignment.

Study Contract

- Lecture need to inform maximum 24 hours prior to the scheduled lecture if there are some changes plan.
- If the lecture do not come 30 minutes after the scheduled class, the class will be canceled.
- Please choose one student as the class coordinator.

Class Assistant

- I will be assisted by two lecture assistants to held the class.
- The class assistant responsible to check the assignment and help student to communicate with the lecture.
- The assistant are your senior :
  - Arzad Lintang Maharani
  - Akhsan Ibrahim

LeetCode

- Please make an account in LeetCode (https://leetcode.com/). Please remember your account (username and password).
- Most of the task given in this class will be based on the problem available in LeetCode.

Assessment Components

- Final semester examination (UAS) 35%
- Mid-semester examination (UTS) 25%
- Assignments and portfolio 20%
- LeetCode portofolio 20%

LeetCode Grading

- Solved 10 problems ⟶ 80
- Solved 15 problems ⟶ 90
- Solved more than 20 problems ⟶ 100

Introduction

- Every computer program consists of two parts: **instructions** and **data**. **Instructions** are command codes that are known and can be executed by the computer. A program consists of a series of instructions, to complete a specific task.
- What is processed by the program is **data**, which is then displayed on the screen in the form of text, tables, graphics, images, sound, video or in other forms.

Introduction

- This course discusses how to store data in a program. Again, **storing data in a program**, while data is processed by the program.
- When the amount of data is getting bigger, certain storage techniques may be needed so that the data is easy to find and process. Processing different types of data requires different techniques.

Installing Python

- You can do it by yourself...

Basic Instruction (*print*)

- The *print* command is used to show a data value to the computer screen. If there is no *print* command, people will not know what is going on in the computer.

```
print(“Hi! My name Fulan!”)
```

Basic Instruction (*variable assigment*)

- The variable assignment command is used to store data or information in computer memory.

```
pi = 3.14
myaddress = "Yogyakarta"
age = 31
```

Basic Instruction (*input*)

- The *input* command is used to allow the computer to request data from the users who running the program.

```
fav = input("Enter your favorite food: ")
print("Thank you. Data saved.")
print(“Your favourite food is”, fav)
```

Basic Data Types and Syntax

- Numerical data types: integer, float, and boolean

```
a = 33
b = 33.33
c = True
```

- Character and Strings

```
characters = ‘c’
strings = “my name Fulan”
huruf = strings[0]
```

- Casting

```
num = 32
num_str = str(num)
sum = num_str + num_str
```

Arithmetic and Logical Operations

- Addition, subtraction, multiplication
  `2+2 ; 4-2 ; 2*2`
- Division (type matters)
  `3/2 = ?`
- Modulus
  `3%2 = 1`
- Exponential
  `3 ** 3 = 27`
- Logical Operator
  `and, or, not`
  `==, !=, <, <=, >, >=`

Complex Data Type (List)

- List is a collection of data where each item has a sequence number. For example :

```
beta = [2, 7, 3.5, "test", True]
```

- The item of the list can be accessed by its index.

```
print(beta[2])
b = beta[2]
beta[2] = 8
```

Complex Data Type (Tuple)

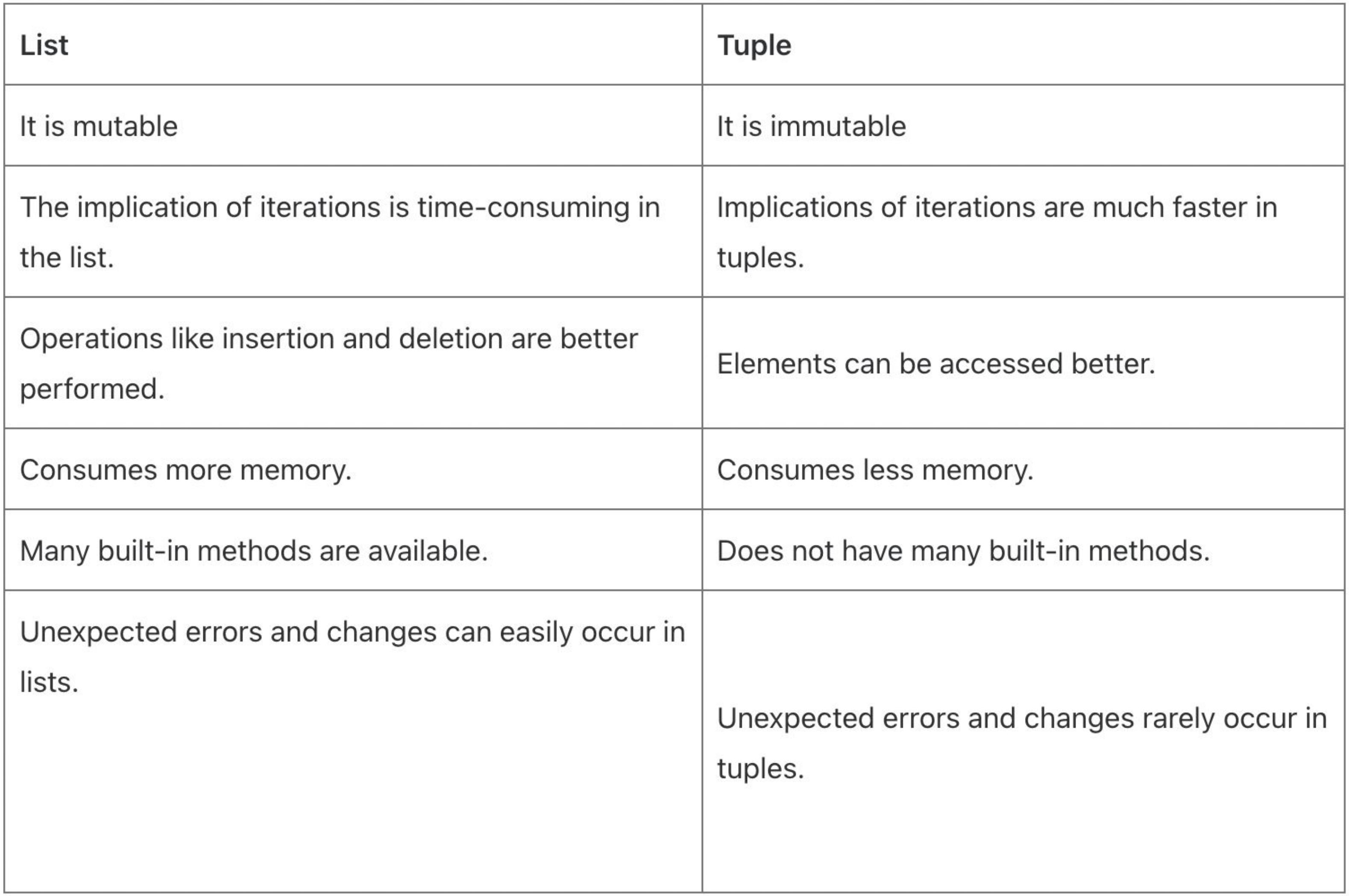
- A tuple is a collection of data like a List. For example:

```
a = (2, 7, 3.5, "test", True)
```

- Similar to List, the item of the tuple can be accessed by its index.

```
print(a[1])
d = a[1]
```

The difference between List and Tuple



List	Tuple
It is mutable	It is immutable
The implication of iterations is time-consuming in the list.	Implications of iterations are much faster in tuples.
Operations like insertion and deletion are better performed.	Elements can be accessed better.
Consumes more memory.	Consumes less memory.
Many built-in methods are available.	Does not have many built-in methods.
Unexpected errors and changes can easily occur in lists.	Unexpected errors and changes rarely occur in tuples.

Complex Data Type (Dictionary)

- Dictionary is a collection of data pairs. Each dictionary item is a pair of data, one is called a key, the other is called a value. For example:

```
age = {"Siti": 18, "Jono": 20, "Sulis": 19}
temp = age["Siti"]
print(age["Jono"])
```