



Basic Programming in Python

1. Session: Organization and Introduction

Nohayr Muhammad Abdelmoneim

Summer Term 2023

April 17th, 2023

Overview

- Organizational Issues
- Requirements to Pass the Course
- Tentative Content of the Course
- Introduction to Programming
- Introduction to Python

Organizational Issues

- Teachers
 - Nohayr Muhammad (50/303): nmuhammadabd@uni-osnabrueck.de
 - Fatemeh Shetabivash (Sophie) fshetabivash@uni-osnabrueck.de
 - Marlon Dammann: mdammann@uni-osnabrueck.de
 - Melisa Altinyelek maltinyelek@uni-osnabrueck.de
- Course: Time and Place
 - Lecture: Monday 12:00 – 14:00; in 66/E34 and digital or pre-recorded
 - Tutorial 1: Tuesday 12:00 – 14:00, Room: 93/E09
 - Tutorial 2: Thursday 12:00 – 14:00, Room 32/218 (mostly online)
- Students:
 - Bachelor Students (Cognitive Science)
 - Master students (Cognitive Science)
- Modules:
 - **B.Sc modules:**
CS-BWP-MCS - Methods of Cognitive Science
KOGW-PWB - Distinguishing elective courses
 - **M.Sc modules:**
CC-MW - Distinguishing elective courses
- The course is worth 4 ECTS credit points.

Organizational Issues

- Intended audience:
 - This course is intended for those with no or very little knowledge in programming.
 - If you already have experience in programming in general or Python in particular, then the “Scientific programming in Python” course would be more suitable for you.
- What will happen in this course?
 - Lectures and tutorials are intended to complete each other, some topics maybe distributed among both.
 - Sessions will be given either **in-class**, **online** or as **pre-recorded** videos.
 - All in-class sessions will be livestreamed and recorded.

Organizational Issues

- Regulations for attending the course
 - You can attend the sessions in presence and as an online course.
 - The room hopefully suffices to give everybody the chance to participate in presence.
- If you have questions, you can either send e-mails to the teachers and tutors or you can use the forum on StudIP.

Organizational Issues: Requirements

- Potential grade distribution:
 - Requirement for final exam: Pass 50% of weekly coding tasks of. Each task is pass/fail.
 - Weekly coding tasks 10-20%
 - Final exam 80-90%
- Final exam
 - Final exam will be held in the last lecture:
Monday 10.07.2023 at 12:00pm
 - The exam will be in the form of coding tasks to be solved.

Tentative topics to be covered

- Variables assignments, Arithmetic operations, basic data types
- Data structures and types conversions.
- Conditional statements, logical operators
- Loops, recursion
- Functions, variable scope
- Strings, string methods
- Plotting
- Dealing with files: input/output
- Numpy Library
- Classes and Object-Oriented Programming*
- Web Scrapping and APIs*

* These topics are subject to time and flow of the course

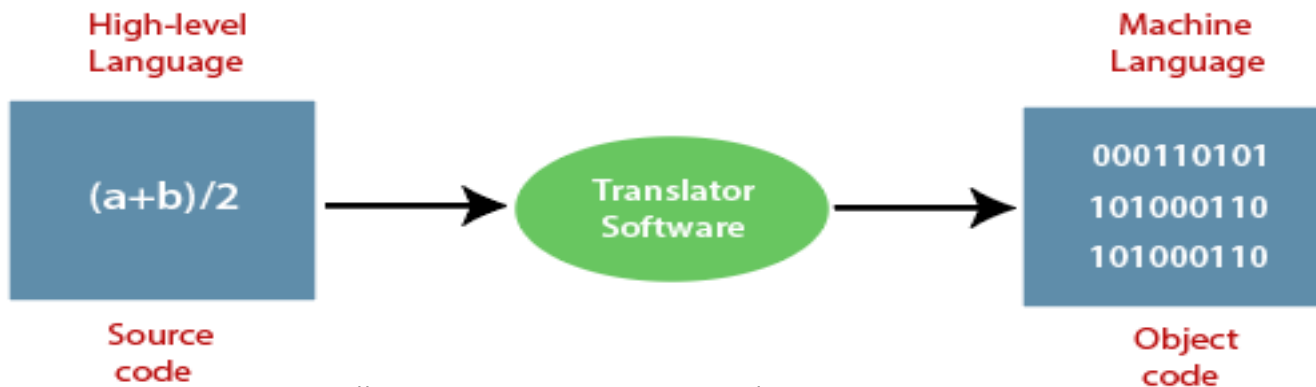
QUESTIONS?

What is programming

- Programming is to instruct the computer to perform certain tasks. No matter how simple or complicated the task is.
- Coding is part of the programming process, where the instructions are written in order for the computer to execute them.
- These instructions or code are written using special languages, called “Programming languages”

What is a programming language

- Machines like computers consist of electric circuits with tiny electric switches which can be either on or off.
- This can be translated to a “Machine language” which consists of 0s and 1s, corresponding to off and on.
- It would be impractical for humans to write instructions using this language.
- A programming language is a predefined high level language that humans can use to write instructions, which is then translated to machine language that is executable by a PC.



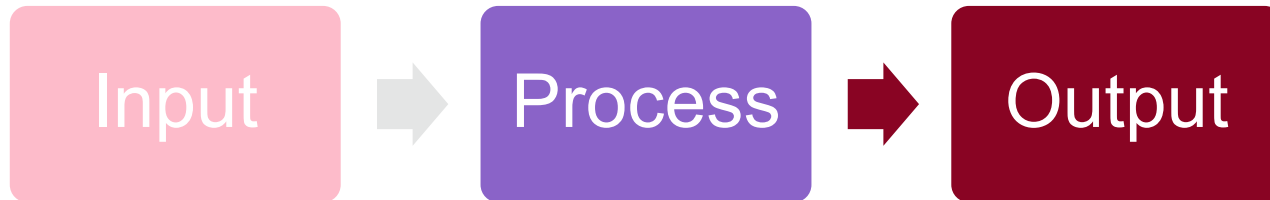
<https://www.tutorialandexample.com/high-level-language>

How to use a programming language

- In order to be able to write code using a programming language that can be executed by a machine we need:
 - Editor: a space for writing the instructions using a specific programming language
 - Compiler/interpreter: a program that translates and converts the source code written in a programming language into machine language that is executable by PC
- Each programming language has its own syntax and its own rules.
- Examples of programming languages:
Python, C/C++, Java, JavaScript, LISP, etc.

What is a program

- A program is a set of instructions written to solve a certain task.
- Usually, a program has set of inputs. Then some operations are applied on the inputs, to finally reach the desired output.



How to solve tasks by programming

- To solve a task you need to divide it to simple sequential instructions that would eventually lead to solving the problem. This is called an “Algorithm”
- You need to determine the inputs needed to solve the task.
- You need to determine the desired output of the program.
- You need to find a set of executable instructions that would transform the given inputs to the desired outputs.
- The order of the instructions matter. I.e., if there are dependencies between instructions that needs to be taken into consideration.

What is Python?

- Python is a widely-used, interpreted, object-oriented, and high-level programming language with dynamic semantics.
- It was first created and released by “Guido Van Rossum” in 1991.
- The name was inspired from an old BBC television comedy sketch series called “Monty Python’s Flying Circus”.
- Python is an open-source language which many anonymous programmers contribute to developing it.



Why Python?

- Python is considered one of the simplest, yet powerful programming languages to use.
- It is currently the most widely used multi-purpose, high-level programming language.
- Python programs generally are smaller than other programming languages. A program written in Python is 2 to 10 times shorter than that written in other languages.
- Python has a huge collection of libraries that is used in many fields: machine learning, image processing, web scraping, etc.

“Hello, World”

• C

```
#include <stdio.h>

int main(int argc, char ** argv)
{
    printf("Hello, World!\n");
}
```

• Java

```
public class Hello
{
    public static void main(String argv[])
    {
        System.out.println("Hello, World!");
    }
}
```

• now in Python

```
print "Hello, World!"
```

<https://analyticsprofile.com/business-analytics/why-should-you-learn-python/>

How to code in Python

- There are different ways to code in Python:
 - Colab: or “colaboratory” supported by google research allows you to write and execute Python code without configuration, with access to GPUs and easy to share.
 - Jupyter notebook: similar to colab but is executed on your own machine and needs configuration of Python and other libraries.
 - Editor and compiler/interpreter: to install an editor where you can write the code. A compiler/interpreter is also needed to translate and execute the code.
- You will learn more about each of these methods in details in the tutorials.
- You need to have an installed version of Python for the rest of the semester.

Your first Python program

```
>>> print("Hello World!")  
Hello World!  
>>>
```

Next..

Tentative content for next session:

- How to write a simple program in Python.
- What are variables and how to use them.
- Rules for variable names.
- Arithmetic operators and their precedence.
- Assignment operators.
- Comparison operators.
- Logical operators.
- Indentation in Python.
- Possibly: If statement.

QUESTIONS?