

# Exercises 01: Basics

## Content

Exercise 1: Development Environment .....	1
Exercise 2: Overview of Extensions .....	1
Exercise 3: Algorithm Description of A* .....	2
Exercise 4: Deep Packet Inspection .....	3

## Exercise 1: Development Environment

Please install the following software:

- [Anaconda Individual Edition](#)
- [Visual Studio Code](#)
  - [Python extension for Visual Studio Code](#)
  - [Jupyter Extension for Visual Studio Code](#)
  - [LaTeX Workshop](#)
- [MiKTeX](#)
- [Perl for MS Windows \(required by LaTeX Workshop\)](#)
- [Wireshark](#)

If you are using Linux, you will find equivalents in the Linux packet managers, e.g. apt.

## Exercise 2: Overview of Extensions

Create an overview of the following Python extensions, cf. Figure 1:

- Scipy
- Scikit-learn
- Pandas
- NetworkX
- Keras
- Gplearn
- NumPy
- Matplotlib
- PM4Py
- ...

The overview should include a short description, a link to the documentation and algorithms that are included. You can choose a diagram or a table. The overview should not be longer than two pages. The result of this exercise is a document (pdf).



Figure 1: Extensions

## Exercise 3: Algorithm Description of A\*

Create an algorithm description of the A\* search algorithm, cf. Figure 2.

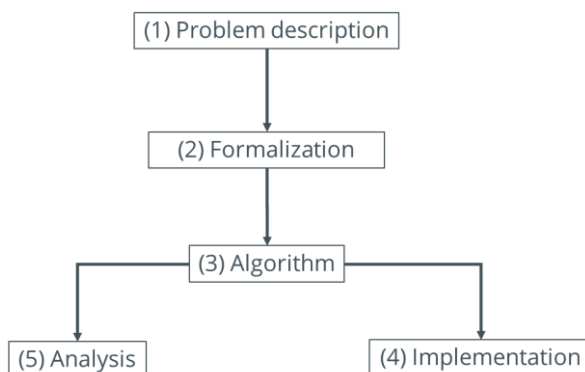


Figure 2: Algorithm Design

First, open the “Graph.py” file with Visual Studio code. Second, use the template “Report.tex” file to describe (1) problem, (2) formalization, (3) algorithm, (4) results and (5) Conclusion. You can use generated figures by the “Graph.py” tool to explain the algorithm steps. The report should not be longer than four pages. The result of this exercise is a document (pdf).

### The XYZ Algorithm

John Doe  
2021

#### 1 Problem

TBD

#### 2 Formalization

TBD

#### 3 Algorithm

TBD

---

**Algorithm 1:** The A\* ...

**Input:** Directed graph  $D$

**Output:**  $n$ -tuple of point tuple

// Step 1: (Calculate shortest path):

1  $P \leftarrow A^*(D)$

2 return  $P$

---

#### 4 Results

TBD

#### 5 Conclusion

TBD

Figure 3: Template

## Exercise 4: Deep Packet Inspection

Create an algorithm description for an algorithm of your choice, cf. Exercise 2.

You should analyze “pcap” files. The files are located in the Data folder. First, open the files with “Wireshark”. Second, export “pcap” to “csv” files, cf. Figure 4. Third, open each “csv” file with pandas. Now you can choose an algorithm and analyze the data. Search for anomalies in the data and visualize your results. Use the same template as described in Exercise 3.

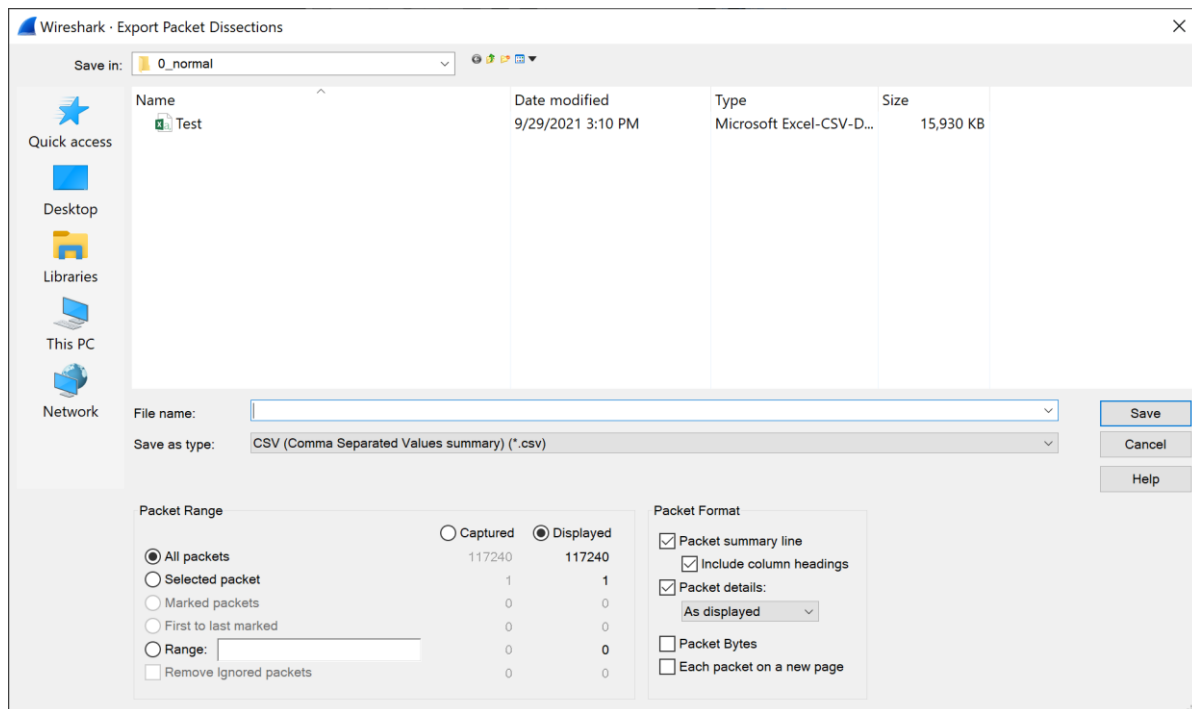


Figure 4: Export Dialog of Wireshark

The report should not be longer than two pages. The result of this exercise is a document (pdf) and a “Python” file with the source code.