

Algorithmik kontinuierlicher Systeme Einrichten der Python-Programmierungsumgebung

Willkommen zu den AlgoKS-Programmierungsübungen! Auf diesem Blatt finden Sie Anweisungen, um Ihre Programmierungsumgebung für die AlgoKS-Übungen einzurichten. Sie können entweder die vorbereitete Programmierungsumgebung im CIP-Pool der Informatik verwenden (**empfohlen**; dies ist auch Remote per SSH möglich), oder eine Programmierungsumgebung lokal einrichten.

1 Python-Umgebung im CIP-Pool verwenden

1.1 Vor Ort im CIP

Loggen Sie sich ein und öffnen Sie eine Konsole. Navigieren Sie in ein Verzeichnis, in dem Sie ihre Übungen machen wollen. Führen Sie nun die folgenden Befehle aus, um das AlgoKS-Umgebungsmodul zu laden und anschließend Jupyter Lab zu starten:

```
$ module load python3/AlgoKS
$ jupyter lab
```

Drücken Sie einmal **Enter**, um das Passwort für die Jupyter-Instanz anzuzeigen. Nun öffnet sich Jupyter Lab in einem Browser. Geben Sie das soeben angezeigte Passwort ein, um mit der Arbeit zu beginnen.

1.2 Remote per SSH

Mithilfe von SSH können Sie die Jupyter Lab-Instanz des CIP auch von Ihrem eigenen PC aus nutzen. Verbinden Sie sich per SSH mit einem CIP-Rechner¹. Aktivieren Sie dabei *port forwarding* für den Port 8080; z.B.:

```
$ ssh -L 8080:127.0.0.1:8080 <username>@cipterm0.cip.cs.fau.de
```

Starten Sie nun den Jupyter Server innerhalb der SSH-Session:

```
$ module load python3/AlgoKS
$ jupyter lab --no-browser
```

Drücken Sie einmal **Enter**, um das Instanzpassword zu sehen. Öffnen Sie dann die Seite `localhost:8080` in Ihrem Browser. Nun sollte die Login-Seite von Jupyter Lab erscheinen; geben Sie dort das Instanz-Passwort ein und legen Sie los.

2 Python-Umgebung Lokal Einrichten

Zum Einrichten der AlgoKS-Programmierungsumgebung gehören zwei Dinge:

Python-Installation Die Zielversion der Programmieraufgaben ist *Python 3.11*; diese ist auch in den CIP-Pools des Department Informatik installiert. Ihre Lösungen zu den Hausaufgaben werden mit Python 3.11 getestet und bewertet – Sie können allerdings auch mit einer neueren Python-Version programmieren. Dabei empfiehlt es sich allerdings, Ihre Lösungen stets mit Python 3.11 im CIP-Pool zu testen.

¹ siehe www.cip.cs.fau.de/documentation/services.de.html

Virtuelle Umgebung und Jupyter Lab Als Programmierumgebung werden wir *Jupyter Lab* verwenden. Jupyter Lab ist eine als Browser-Anwendung konzipierte interaktive Programmierumgebung. Wir werden Jupyter Lab, sowie weitere für AlgoKS benötigte Pakete, innerhalb einer virtuellen Umgebung einrichten.

Folgen Sie den ihrer Plattform entsprechenden Anweisungen, um Python zu installieren und anschließend die virtuelle Umgebung einzurichten.

Linux und MacOS

Installieren Sie Python mithilfe ihres System-Paketmanagers, oder über den MacOS-Installer (siehe <https://docs.python.org/3/using/mac.html>).

Erstellen Sie einen neuen Ordner, welcher Ihre AlgoKS-Programmierumgebung beinhalten wird. Laden Sie anschließend die Datei `algoks-requirements.txt` aus dem StudOn-Kurs (Ordner **Übung** > **Python-Setup**) herunter und legen Sie diese in dem neuen Ordner ab. Öffnen Sie nun eine Konsole in diesem Verzeichnis und führen Sie folgenden Kommandos aus, um die virtuelle Python-Umgebung zu erstellen:

```
$ python3 -m venv algoks-venv
$ source algoks-venv/bin/activate
$ pip install -r algoks-requirements.txt
```

Dies erzeugt ein sogenanntes *virtual environment* im Ordner `algoks-venv`, aktiviert dieses, und installiert die in der Datei `algoks-requirements.txt` aufgelisteten notwendigen Pakete.

Windows

Python 3.11 Installieren

Öffnen Sie <https://www.python.org/downloads/windows>, laden Sie den Windows-Installer für Python 3.11.9 herunter, und führen Sie diesen aus. Detaillierte Anweisungen zur Installation und Einrichtung von Python unter Windows finden Sie unter <https://docs.python.org/3/using/windows.html>.

Virtuelle Umgebung Aufsetzen

Erstellen Sie einen neuen Ordner, welcher Ihre AlgoKS-Programmierumgebung beinhalten wird. Laden Sie anschließend die Datei `algoks-requirements.txt` aus dem StudOn-Kurs (Ordner **Übung** > **Python-Setup**) herunter und legen Sie diese in dem neuen Ordner ab. Öffnen Sie nun entweder eine Windows-Kommandozeile (`cmd.exe`) oder eine PowerShell-Konsole. Prüfen Sie Ihre Python-Installation mit dem folgenden Befehl:

```
$ py --list
```

Führen Sie anschließend die entsprechenden folgenden Kommandos aus, um die virtuelle Python-Umgebung zu erstellen. Hier wird jeweils eine virtuelle Umgebung im Ordner `algoks-venv` erzeugt und aktiviert; anschließend werden die in der Datei `algoks-requirements.txt` aufgelisteten notwendigen Pakete installiert.

Windows Batch (`cmd.exe`)

Führen Sie die folgenden Befehle aus:

```
$ py -3.11 -m venv algoks-venv
$ .\algoks-venv\Scripts\activate.bat
$ pip install -r algoks-requirements.txt
```

PowerShell

Führen Sie die folgenden Befehle aus:

```
> py -3.11 -m venv algoks-venv  
> .\algoks-venv\Scripts\Activate.PS1  
> pip install -r algoks-requirements.txt
```

Eventuell müssen Sie zunächst Ihre *Execution Policy* ändern, um das Ausführen von `Activate.PS1` zu erlauben:

```
> Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
```

Jupyter Lab Starten

Am Anfang einer neuen Sitzung, öffnen Sie zunächst ihre Konsole und aktivieren Sie die virtuelle Umgebung. Starten Sie dann die Programmierungsumgebung *Jupyter Lab* wie im Folgenden gezeigt. Die Jupyter Lab-Oberfläche öffnet sich anschließend in Ihrem Browser.

POSIX Shell (Linux & MacOS)

```
$ source algoks-venv/bin/activate  
$ jupyter lab
```

Windows Batch (cmd.exe)

```
$ .\algoks-venv\Scripts\activate.bat  
$ jupyter lab
```

PowerShell

```
> .\algoks-venv\Scripts\Activate.PS1  
> jupyter lab
```