

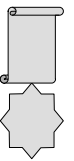
# LS 3 – CPU-Monitoring-System

Die CPU-Auslastung soll über einen längeren Zeitraum aufgezeichnet werden. Die Auslastung lässt sich mit Python herausbekommen. Für die dauerhafte Aufzeichnung kann man eine Datenbank nutzen. Können Python und Datenbank zusammenarbeiten?



## Arbeitshinweise:

- Erarbeiten Sie das Aufgabenblatt in Gruppen von 2-4 Personen
- Sie haben 10 Wochen Zeit zur Bearbeitung.
- Bei einigen Aufgaben sollen Sie Programme erstellen. Diese ist selbstständig zu den angegebenen Terminen abzugeben. Die vorgestellten Programmierrichtlinien sind einzuhalten. Auffallende Ähnlichkeiten zu anderen Gruppen sind zu vermeiden.
- Es findet eine Benotung im Rahmen der SL-Note statt. In die Benotung fließt neben den vorgezeigten Ergebnissen auch die Arbeitsweise im Unterricht, sowie die Anwesenheit ein.
- Alle Ergebnisse (Programme, Dokumentation) sollen in einem lokalen GitHub-Verzeichnis abgespeichert werden. Das lokale GitHub-Verzeichnis ist während der Projektzeit einmal dem Lehrer vorzuführen.



## Aufgabe 1 – Modularer Taschenrechner

Erstellen Sie einen modular aufgebauten Taschenrechner.

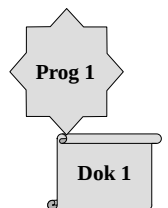
### Funktionale Anforderungen

- Grundrechenarten ( +, -, \*, / )
- Trigonometrische Funktionen ( sinus, cosinus, tangens, arcus sinus, arc cosinus, arc tangens )

### Strukturelle Anforderungen

- Taschenrechner läuft in einer Schleife, bis der Benutzer ihn explizit beendet. In einem Schleifendurchlauf kann der Benutzer die Rechenart (Funktion) wählen, seine Zahleneingaben machen und das Ergebnis sich ansehen.
- Für jede Rechenart bzw. Rechenfunktion ist eine eigene Funktion zu schreiben.
- Alle Benutzerinteraktionen sollen in eigene Funktionen ausgelagert werden.
- Der Taschenrechner ist in 4 Module aufzuteilen. Ein Modul enthält alle trigonometrischen Funktionen, ein Modul enthält alle Grundrechenarten, ein Modul enthält nur Funktionen zur Benutzerinteraktion. Diese drei ersten Module enthalten nur Funktionen. Das vierte Modul enthält eine Steuerfunktion, die innerhalb des Moduls aufgerufen wird.

Speichern Sie das Programm unter dem Namen **aufgabe\_ls3\_03...py** ab. Dokumentieren Sie die Modul- und Funktionsaufrufhierarchie in geeigneten Diagrammen. Dokumentieren Sie die Steuerfunktion mit einem Struktogramm. Die Programme und Struktogramme sollen spätestens in der dritten Projektwoche abgegeben werden.



## Aufgabe 2 – Datenbankfunktionen extrahieren

Bauen Sie `aufgabe_ls3_00.py` so um, dass die 4 allgemeinen Zugriffsfunktionen in ein eigenes Modul ausgelagert sind, die importiert wird. Die Funktion `dbZugriffsWerte` ist spezifisch und bleibt. Verlagern Sie den restlichen Skript-Code in eine Funktion. Speichern Sie das Datenbankmodul unter dem Namen **`dblib.py`** ab, das umgebaute Beispiel soll unter dem Namen **`aufgabe_ls3_01.py`**. Dokumentieren Sie die Modul- und Funktionsaufrufhierarchie.

## Aufgabe 3 – CPU-Monitoring-System

Die CPU-Auslastung soll über einen längeren Zeitraum aufgezeichnet werden. Hierzu ist ein Aufzeichnungssystem zu erstellen, welches aus mehreren Komponenten besteht:

1. Einer zentralen Datenbank zur Speicherung der Auslastungswerte
2. Ein Python Programm, welches in regelmäßigen Intervallen die CPU-Auslastung ermittelt und diese in der Datenbank abspeichert.
3. Ein Python Programm, welches die in der Datenbank enthaltenen Werte anzeigt. Wenn möglich sollen sinnvolle Auswahlkriterien unterstützt werden.
4. Ein Python Programm, welches die in der Datenbank enthaltenen Werte als CSV-Datei exportiert, so dass eine Weiterverarbeitung und Visualisierung mit Hilfe eines Tabellenkalkulationsprogramms möglich ist.



Mögliche Erweiterungen:

- Erweitern Sie Komponente 3, so dass neben der Anzeige auch eine einfache statistische Auswertung möglich ist.
- Erweitern Sie Komponente 3, so dass auch ein Löschen der Daten möglich ist.
- Erweitern Sie das System so, dass neben der CPU-Auslastung noch andere Kenngrößen Ihrer Wahl abgespeichert und angezeigt werden
- Eine zusätzliche Übersichtsdokumentation (Kontextdiagramm, Blockdiagramm mit Modulen, Funktionsaufrufhierarchie, ...) wäre hilfreich.



## Hinweise:

- Das CPU-Monitoring-System soll spätestens in der neunten Projektwoche abgegeben werden.
- Benutzen XAMP Portable, zum Aufbau einer MySQL- bzw. MariaDB-Datenbank.
- Die Komponenten 3 und 4 können zu einer zusammengefasst werden.
- Ihr Beispiel soll in einem Github Projekt abgespeichert sein.
- Die erstellten Programme sollen den Programmierrichtlinien entsprechen. Sie sollen aussagekräftig kommentarisiert sein.
- Speichern Sie die Programme unter dem Namen **aufgabe\_ls3\_02a.py**, **aufgabe\_ls3\_02b.py**, ... ab. Das Skript zur Erstellung der Datenbank soll **aufgabe\_ls3\_02.sql** heißen.

**Zusatzaufgaben (Binnendifferenzierung)****Aufgabe 4 - Lambda-Funktionen**

In Python gibt es Lambda-Funktionen. Erarbeiten Sie ein Beispiel, aus dem hervorgeht, was Lambda-Funktionen sind und wie man sie nutzen kann.

**Aufgabe 5 - Taschenrechner mit graphischer Benutzeroberfläche**

Bauen Sie den Taschenrechner aus Aufgabe 1 so um, dass er anstelle der Konsolen-Benutzerschnittstelle eine graphische Benutzeroberfläche besitzt. Realisieren Sie dies so, dass Sie die Module für die Grundrechenarten und die trigonometrischen Funktionen weiterhin nutzen können.