

### 3. WS Algorithmen und algorithmisches Denken

#### Gestaltung einer Unterrichtseinheit (90min)

dazu gehören:

- Kurze **Präsentation** der Lerninhalte (Präsentationsprogramm, Tafel, Folien, Beamer...).
- Erstellung eines **Handouts** für Ihre Mitschüler im **pdf-Format**.
- Übungsphase mit **Aufgabenblatt** am Rechner im **pdf-Format**.

Ziel:

Sie "unterrichten" die von Ihnen ausgewählten Lerninhalte.

In die Bewertung geht die Gestaltung mit dem Schwerpunkt auf dem "Nutzen", den Ihre Mitschüler aus Ihrer Stunde ziehen, ein.

Jedes Mitglied Ihrer Gruppe muss in der Lage sein, den Workshop zu halten.

#### Inhalte

In dieser Unterrichtseinheit soll das algorithmische Denken geschult werden ;-)

Stellen Sie dazu noch einmal einige Grundalgorithmen vor, z.B. Sortierverfahren, Suchalgorithmen Maximum, Minimum... Stellen Sie dabei die Algorithmen als **PAP**, **Stuktogramm** und im **Pseudocode** vor. Erklären Sie noch einmal kurz, was man unter **rekursiver Programmierung** versteht und geben Sie ein Beispiel. Stellen Sie einen rekursiven Algorithmus vor.

Stellen Sie Ihre Algorithmen als Übersicht in Ihrem **Handout** vor und fassen Sie dort stichwortartig die rekursive Programmierung zusammen.

- In dieser Unterrichtseinheit soll das algorithmische Denken geschult werden ;-)  
Stellen Sie dazu noch einmal einige Grundalgorithmen vor, z.B. Sortierverfahren, Suchalgorithmen Maximum, Minimum, alte Verschlüsselungsverfahren... Stellen Sie dabei die Algorithmen als **PAP**, **Stuktogramm** und im **Pseudocode** vor. Dabei können Sie auch gerne mal etwas "seltener" Sortieralgorithmen vorstellen...  
Erklären Sie noch einmal kurz, was man unter **rekursiver Programmierung** versteht und geben Sie ein Beispiel. Stellen Sie einen rekursiven Algorithmus vor.  
Stellen Sie Ihre Algorithmen als Übersicht in Ihrem **Handout** vor und fassen Sie dort stichwortartig die rekursive Programmierung zusammen.
- Lassen Sie die Algorithmen am Rechner programmieren. sehen Sie mindestens eine Aufgabe vor, die als Struktogramm/ Pseudocode zu lösen ist, dabei können Sie sich an alten Prüfungen orientieren.

Der Schwerpunkt eines Workshops liegt auf Work, also der Übungsphase !

Die Aufgaben müssen zur Thematik passen. Sie setzen dabei Fachinformatiker mit Programmiererfahrung voraus. Zu "leichte" Aufgaben, die jeder in Rekordzeit lösen kann führen ebenso zur Abwertung, wie zu schwere Aufgaben. Originelle Aufgaben steigern die Motivation. Tabu sind bereits im Unterricht gestellte Aufgaben! Sie dürfen gerne Aufgaben im Internet als Anregung bzw. als Vorlage benutzen, sie dürfen aber nicht einfach kopiert werden!

...siehe nächste Seite

Nutzen Sie zur Gestaltung Ihrer "Stunde" das Internet und schauen Sie sich ältere Prüfungen an.

Überlegen Sie sich eine **eigene Darstellung** der Lerninhalte und **eigene Übungsaufgaben, wobei Umformulieren erlaubt ist.**

Ein bloßes "Zusammen-Kopieren" der Präsentation aus Literatur oder Internet genügt nicht den Ansprüchen und kann zur Abwertung führen!!

Das **Handout**, die **Präsentation**, das **Aufgabenblatt** und die **Musterlösungen** aller gestellten Aufgaben werden nach dem Vortrag **in digitaler Form abgegeben!**