# Lern- und Arbeitsauftrag *LA\_319\_6913\_FunktionenKonventionen.docx*

|  |  |
| --- | --- |
| Titel | Verwendung von Funktionen und Konventionen |
| Modul | 319 Informatiker/in EFZ |
| Autor / Version | Lars Meyer / V1.0 |
| Hilfsmittel: | PR\_319\_Funktionen.pptx  PR\_319\_Konventionen.pptx  C# Entwicklungsumgebung  LA\_319\_6913\_FunktionenKonventionen\_Abgabe.zip |
| Nachweis | Leistungsbeurteilung zu Ende des Moduls |
| Sozialform | Einzelarbeit / Partnerarbeit |
| Leistungsziele | 4.12 / 4.13 / 4.14 / 5.1 / 5.2 / 5.3 / 5.4 |

## Ausgangslage

Der Lebenszyklus von Sourcecode endet nicht nach «dem Coden», Testen und der Inbetriebnahme. Im Gegenteil fängt es danach erst an, richtig spannend zu werden. Folgendes sollten Sie im Hinterkopf behalten:

* Es wird mehr Code gelesen als geschrieben
* Code wird sich früher oder später ändern

Umso mehr muss darauf geachtet werden, dass Sourcecode einfach leserlich und wartbar ist. Funktionen und Konventionen sind ein Schritt auf dem Weg, dies zu erreichen.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

In diesem Auftrag…

* sammeln Sie Erfahrungen mit der Erstellung und der Verwendung von Funktionen.
* erfahren Sie den Sinn und Zweck von Konventionen und lernen deren Anwendung kennen.

### Aufgabe 1:

Studieren Sie die Präsentation *PR\_319\_Funktionen*.

### Aufgabe 2:

Gegeben ist das Programm in *LA\_319\_6913\_FunktionenKonventionen\_Abgabe.zip*.

Zwei Funktionen sind bereits fertig erstellt, um die Leserlichkeit zu fördern.

Zwei weitere Funktionen sind für Sie zur Ergänzung vorbereitet.

1. Machen Sie sich mit dem Programm vertraut. Führen Sie es aus, testen Sie es, verstehen Sie den Code.
2. Identifizieren Sie die wiederkehrend gleichen Code-Abschnitte bezüglich des Einlesens einer Ganzzahl. Lagern Sie den Code in die vorbereitete Funktion aus.
3. Identifizieren Sie die wiederkehrend gleichen Code-Abschnitte bezüglich des Ermittelns der maximalen Elementlänge (Anzahl Stellen) in einem Array. Lagern Sie den Code in die vorbereitete Funktion aus.

Testen Sie nach den Änderungen das Programm gemäss folgender Vorlage:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### Aufgabe 3:

Studieren Sie die Präsentation *PR\_319\_Konventionen*.

### Aufgabe 4:

Lösen Sie eine der beiden Teilaufgaben 4.1 oder 4.2 nach freier Wahl.[[1]](#footnote-1)

Folgende Anforderungen sind gegeben:

* Implementieren Sie das Programm unter sinnvoller Verwendung mindestens einer Funktion.
* Setzen Sie das Programm unter Berücksichtigung der BBB-Konventionen um[[2]](#footnote-2).

Testen Sie das Programm gemäss den Screenshots.

### Teilaufgabe 4.1 (einfacher):

Erstellen Sie ein Programm, welches eine Ganzzahl aus dem Dezimalsystem zwischen 0 und 15 in eine Hexadezimalzahl umwandelt.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### Teilaufgabe 4.2 (schwieriger):

Erstellen Sie ein Programm, welches eine beliebige positive Ganzzahl aus dem Dezimalsystem in eine Hexadezimalzahl umwandelt.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## Gütekriterien

Der Lern- und Arbeitsauftrag ist erfüllt, wenn …

* Sie beurteilen können, wo eine Auslagerung von Code in Funktionen sinnvoll ist.
* Sie Code in Funktionen auslagern können.
* Sie die erläutern können, was Code-Konventionen und deren Vorteile sind.
* Sie die BBB Code-Konventionen nach Vorgabe umgesetzt haben.

## Zusätzliche Angaben zum Auftrag

* Ein *uint* verhält sich wie ein *int*, lässt jedoch nur positive Ganzzahlen zu.
* Dezimal- in Hexadezimal umwandeln:

<https://de.wikihow.com/Zahlen-vom-Dezimalsystem-in-das-Hexadezimalsystem-umwandeln>

<https://www.matheretter.de/wiki/dezimal-hexadezimal>

## Mögliche Erweiterungsaufträge

Keine

1. Je nach Lust und Zeit auch beide. [↑](#footnote-ref-1)
2. Gilt auch für alle künftigen Aufträge. [↑](#footnote-ref-2)