**Änderungen in Materialien zum 01.09.2023**

* L2\_1 Information\_Algorithmik: Formulierung angepasst
* L2\_2.1 Information\_Bubble\_Sort: Formulierung angepasst
* L2\_2.1 Präsentation\_Prinzip\_Bubble\_Sort: Redaktionelle Änderung
* L3\_2.1 Arbeitsauftrag Stapelspeicher: Rechtschreibfehler korrigiert
* L3\_2.1 Lösung Stapelspeicher: Rechtschreibfehler korrigiert, Fehler in Lösung der beiden Murmelbahnaufgaben korrigiert
* L3\_4 Information\_Baum: Redaktionelle Anpassung auf Seite 3 (tabellarische Darstellung des Baums). Definition eines geordneten Baumes nochmals angepasst.
* L3\_4.1 Arbeitsauftrag Baum: Aufgabe 7 angepasst
* L3\_4.1 Lösung Baum: Aufgabe 7 angepasst
* L3\_4.2 Arbeitsauftrag Übung Baum: Aufgaben 1 und 3 angepasst
* L3\_4.2 Lösung Übung Baum: Aufgaben 1 und 3 angepasst

**Änderungen in Materialien zum 01.09.2022**

* L1\_3.6 Lösung Volleyball Inhalte entfernen: Struktogramm ersetzt, Lösungscode verbessert
* Operatorenliste auf Version 2.0 aktualisiert

**Änderungen in Materialien nach ihrer Veröffentlichung (September 2021)**

**Lernfortschritt 1:**

L3\_4 Information Baum:

* Präzisierung der Begriffe „geordnet“ und „sortiert“

L3\_4.1 Arbeitsauftrag Baum:

* Änderung von Aufgabe 4
* Austausch von Aufgabe 7
* Aktualisierung der Lösungen

L3\_4.2 Arbeitsauftrag Übung Baum:

* Aktualisierung der Lösungen

**Änderungen in Materialien nach ihrer Veröffentlichung (April 2020)**

**Lernfortschritt 1:**

L1\_1 Info Datenstrukturen.docx Beispiel für verkettete Liste geändert.

L1\_2 Info Array.docx Zu Möglichkeit 1: (stgr- und png-Dateien ebenso) Deklaration eines Arrays (ohne Initial.) ist nicht möglich è „Deklaration **und Initialisierung**: … „

L1\_3.2 Info innerHTML.docx Bei zweitem Beispiel: **;;;** entfernt

L1\_3.2 Lösung Gewinnziehung.docx Quellcode:

(html-Datei ebenso) gLosNr[i] = **parseInt**(document….) da lt. Problemanalyse gLnr vom Typ Zeichenkette

L1\_3.3 Aufgabe Trainingsanalyse.docx Problemstellung: „… IDs = ="w1", "w2", "w**3**", usw…“

L1\_3.3 Lösung Trainingsanalyse.docx Variable **summeP** zur Ermittlung der gesamten Punktezahl eingefügt (mittelP nur zur Berechnung der durchschn. Punktezahl)

(html-, stgr- und png-Dateien ebenso) Optimierung der Auswertung eingefügt: Überprüfung der Elemente wg. Min bzw. Max erst ab dem 2. Element notwendig.

L1\_3.6 Info Inhalte entfernen.docx Erster Satz: „… besteht auch **die** Möglichkeit ….

Ergänzt um Kürzung eines Arrays um eine Stelle

L1\_3.6 Lösung Volleyball Inhalte entfernen.docx Problemanalyse – (5) Verarbeitung:

„**Kürze das Array *kader* um eine Stelle**.“

(html-, stgr- und png-Dateien ebenso) Struktogramm ergänzt **Zuweisung: kader.lenght = um eine Stelle kürzen**

Quellcode ergänzt **kader.length = kader.length-1;**

L1\_4.1 Lösung Vertiefung Schreibtischtest.docx zu Aufgabe 1.1 (Aufgabe + Lösung) arr = [5, 11, 3, 1,10**9**]

zu Aufgabe 2.1 2tes Mal Durchgang 4 in Durchgang **5** geändert.

L1\_4.2 Auftrag Vertiefung Wetterstation.docx Problemstellung: Hinweis auf Vorlage- und Lösungsdatei korrigiert (L1\_4\_**2**\_...)

L1\_5 Zusatzaufgabe autoGewinnziehung.docx Problemanalyse Schritte (1) und (3) Gewinnlosnummer**n**

Problemanalyse Schritte (3) **gewinnnr** (statt aGeLose)

**Lernfortschritt 2:**

L2 Aktivitätsverfolgung\_Lernfortschritt\_2.docx L2\_2.3.3 **Arbeitsauftrag** Volleyball …docx

L2\_1 Information Algorithmik.docx Fehlende Beispiele für Eigenschaften von Algorithmen eingefügt.

L2\_1 Aufg./Lösg. Einführung Algorithmik.docx Aufgabe 3.2 „Für Dieselfahrz. werden pro **angefangene** ..“ „Für ein Dieselfahrzeug mit 2090 **1990** ccm ..“

L2\_2.1 Präsentation\_Prinzipt\_Bubble\_Sort.pptx Folie 7: **5 > 4**

Folie 13:5 > **3**

L2\_2.1.1 Lösung Bubble Sort Lotto.docx (5) Verarbeitung: „…**so oft** …“

L2\_2.1.2 Lösung Bubble Sort Zahlenreihe.docx (5) Verarbeitung: „…**so oft** …“

L2\_2.1.3 Lösung Bubble\_sort.docx Zu Aufgabe 2.1: „…immer **len[array]** – 1) Phasen…“

(py-, stgr- und png-Dateien ebenso) Zu Aufgabe 2.4: Sortieralgorithmus (while-Schleife) angepasst an Lösungen in L2\_2.1.1 und L2\_2.1.2

L2\_2.2 Information\_Selection\_Sort.docx Struktogramm – 1. Wiederholungsschleife:

(stgr- und png-Dateien ebenso) … solange aktIndex < Anzahl Elemente *zahlen* **– 1**., ...

L2\_2.2.1 Lösung Selection Sort Lotto.docx Struktogramm/Quellcode

(html-, stgr- und png-Dateien ebenso) 1. Wiederholungsschleife: äußere Schleife muss nur bis zum vorletzten Element laufen … solange aktIdx < Anzahl Elemente lotto **– 1**

L2\_2.2.2 Lösung Selection Sort Zahlen.docx Struktogramm/Quellcode

(html-, stgr- und png-Dateien ebenso) 1. Wiederholungsschleife: … solange aktIdx < Anzahl Elemente zahlen **– 1**

L2\_2.2.3 Lösung Selection Sort absteigend.docx Struktogramm/Quellcode

(html-, stgr- und png-Dateien ebenso) 1. Wiederholungsschleife: … solange aktIdx < laenge **– 1**

L2\_2.3.2 Auftrag Sortieralgorithmen prüfen.docx Aufgabe 1

(html-, stgr- und png-Dateien ebenso) 1. Wiederholungsschleife: solange zaehler1 < Anzahl Elemente… a **– 1**

Fehler im Struktogramm korrigiert: a[zaehler**2**] < a[index\_klein]

**Lernfortschritt 2:**

L2\_3.1.3 Lösung Volleyball Spieler suchen.docx Problemanalyse – (3) Variablenliste

(html-, stgr- und png-Dateien ebenso) indexNr entfernt, da nie verwendet

Quellcode: **anzahl = 0; anzahl = anzahl + 1**

(html-, stgr- und png-Dateien ebenso) Struktogramm ergänzt: Ausgabe: ausgabe

(html-, stgr- und png-Dateien ebenso) Struktogramm, Quellcode, (5) Verarbeitung: Variable *ausgabe* initialisiert mit „..nicht im Mannschaftskader“.

else if – Zweig entfernt, da überflüssig.

L2\_3.2.1 Lösung Binäre Suche Losnummer.docx Schreibtischtest – 1. Durchlauf: ende: **9**

L2\_3.3.1 Lösung Übung Suche.docx Aufgabe 2: „.. setze **e** auf m - 1

L2\_2.3.3 Lösung Volleyball Spieler sortieren.docx Struktogramm:

(html-, stgr- und png-Dateien ebenso) 1. Wiederholungsschleife: … solange i < Anzahl Elemente… kader **– 1**

L2\_3.3.2 Lösung Suchalgorithmen prüfen.docx Aufgabe 2.2 merker = false à = **falsch**

merker = true à = **wahr**

L2\_3.6 Lösung Volleyball Inhalte entfernen.docx Problemanalyse – (4) Formdesign ergänzt um „Spieler nicht gefunden“

Problemanalyse – (5) Verarbeitung:

„**Kürze das Array *kader* um eine Stelle**.“

(html-, stgr- und png-Dateien ebenso) Struktogramm ergänzt

**Verzweigung: gefunden GLEICH wahr …**

**Zuweisung: kader.lenght = um eine Stelle kürzen**

Quellcode ergänzt **gefunden == wahr …. kader.length = kader.length-1;**