

Entwicklung Interaktiver Anwendungen 1



Praktikum: Semesteraufgabe

Erstellen Sie Ihren persönlichen Online-Portfolio

Vorlesung 14

Entwicklung von Algorithmen

Algorithmus

Systematische Problemlösung

Begriff

- Ein Algorithmus ist eine eindeutige **Handlungsvorschrift** zur **Lösung eines Problems** oder **einer Klasse von Problemen**.
- Algorithmen bestehen aus **endlich vielen**, wohldefinierten **Einzelschritten**.
- Somit können sie zur Ausführung in einem **Computerprogramm** implementiert, aber auch in **menschlicher Sprache formuliert** werden.
- Bei der Problemlösung wird eine bestimmte **Eingabe** in eine bestimmte **Ausgabe** überführt.

Grundprinzip „Divide et impera“

- lat. für „Teile und herrsche“
- Interpretation: **Zerlege** eine Problemstellung in seine **Teilprobleme** und **löse diese Schritt für Schritt**
- **Implementiere die Teilprobleme** in der zur Verfügung stehenden Sprache, z.B. Programmiersprache wie TypeScript, Java, C++, PHP
- Idee:
 - kleine Probleme sind leichter zu lösen, als große!
 - Aufgabenverteilung auf mehrere Personen

Beispiel Algorithmus I

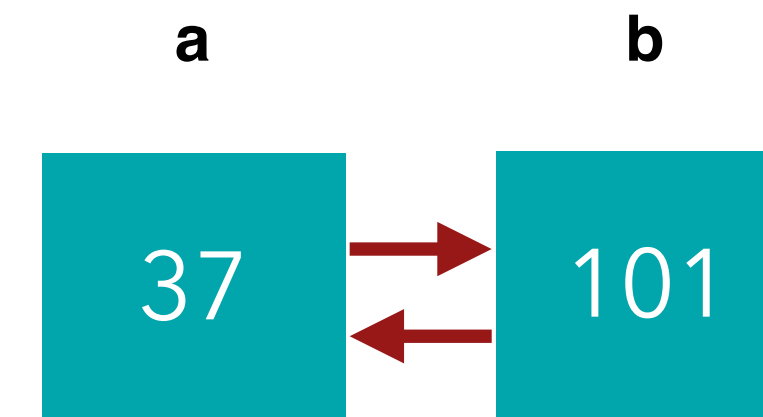
Aufgabe: Sie haben zwei Variablen, in denen jeweils ein numerischer Wert gespeichert ist. Schreiben Sie eine Funktion, die die beiden Werte vertauscht.

5 Minuten Zeit – dann Besprechung

**Sie haben zwei Variablen, in denen jeweils ein numerischer Wert gespeichert ist.
Schreiben Sie eine Funktion, die die beiden Werte vertauscht.**

1. Initialisiere zwei Variablen mit numerische Werten

```
var a : number = 37;  
var b : number = 101;
```



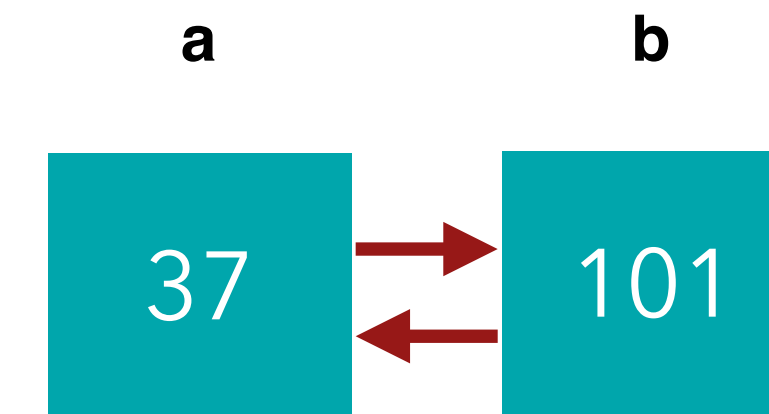
**Funktioniert nicht
in einem Schritt!**

**Sie haben zwei Variablen, in denen jeweils ein numerischer Wert gespeichert ist.
Schreiben Sie eine Funktion, die die beiden Werte vertauscht.**

1. Initialisiere zwei Variablen mit numerische Werten

```
var a : number = 37;  
var b : number = 101;
```

2. Deklaration der Funktion mit zwei Übergabeparametern a und b
`function swapValues (a: number, b: number) : void { ... }`



**Funktioniert nicht
in einem Schritt!**

**Sie haben zwei Variablen, in denen jeweils ein numerischer Wert gespeichert ist.
Schreiben Sie eine Funktion, die die beiden Werte vertauscht.**

1. Initialisiere zwei Variablen mit numerische Werten

```
var a : number = 37;  
var b : number = 101;
```

2. Deklaration der Funktion mit zwei Übergabeparametern a und b

```
function swapValues (a: number, b: number) : void { ... }
```

3. Weise den Wert der ersten Variable einer temporären Variable zu

```
var temp : number = a;
```



**Sie haben zwei Variablen, in denen jeweils ein numerischer Wert gespeichert ist.
Schreiben Sie eine Funktion, die die beiden Werte vertauscht.**

1. Initialisiere zwei Variablen mit numerische Werten

```
var a : number = 37;  
var b : number = 101;
```

2. Deklaration der Funktion mit zwei Übergabeparametern a und b

```
function swapValues (a: number, b: number) : void { ... }
```

3. Weise den Wert der ersten Variable einer temporären Variable zu

```
var temp : number = a;
```

4. Weise den Wert der zweiten Variable der ersten Variable zu

```
a = b;
```



**Sie haben zwei Variablen, in denen jeweils ein numerischer Wert gespeichert ist.
Schreiben Sie eine Funktion, die die beiden Werte vertauscht.**

1. Initialisiere zwei Variablen mit numerische Werten

```
var a : number = 37;  
var b : number = 101;
```

2. Deklaration der Funktion mit zwei Übergabeparametern a und b

```
function swapValues (a: number, b: number) : void { ... }
```

3. Weise den Wert der ersten Variable einer temporären Variable zu

```
var temp : number = a;
```

4. Weise den Wert der zweiten Variable der ersten Variable zu

```
a = b;
```

5. Weise den Wert der temporären Variable der zweiten Variable zu

```
b = temp;
```



Beispiel Algorithmus II

Aufgabe: Ermitteln Sie aus einer beliebigen Anzahl numerischer Werte den höchsten Wert und geben Sie diesen auf die Konsole aus.

5 Minuten Zeit – dann Besprechung

Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten

```
var myNumbers : number [] = [89,67,58,2,93,...,234,23];
```

89

67

58

234

...

23

Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten

```
var myNumbers : number [] = [89,67,58,2,93,...,234,23];
```

2. Die erste Zahl ist zunächst die Größe

```
var highestNumber : number = myNumbers[0]
```

89

89

67

58

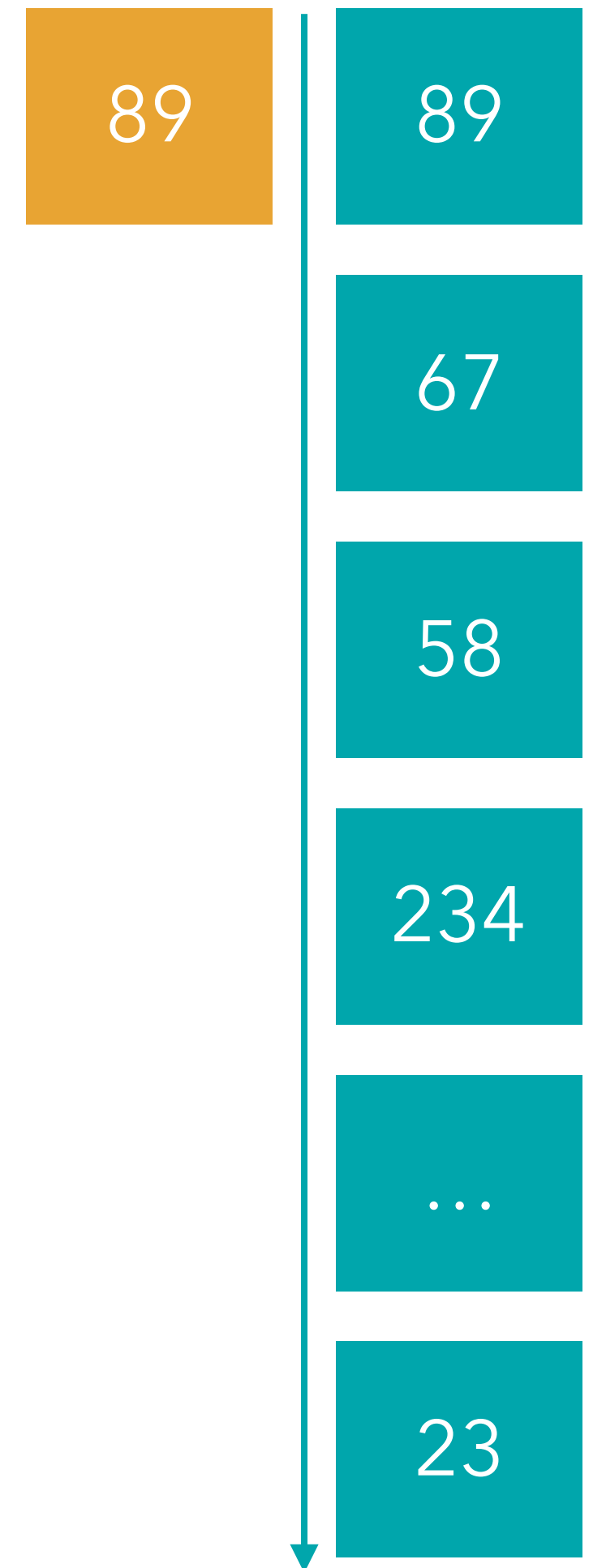
234

...

23

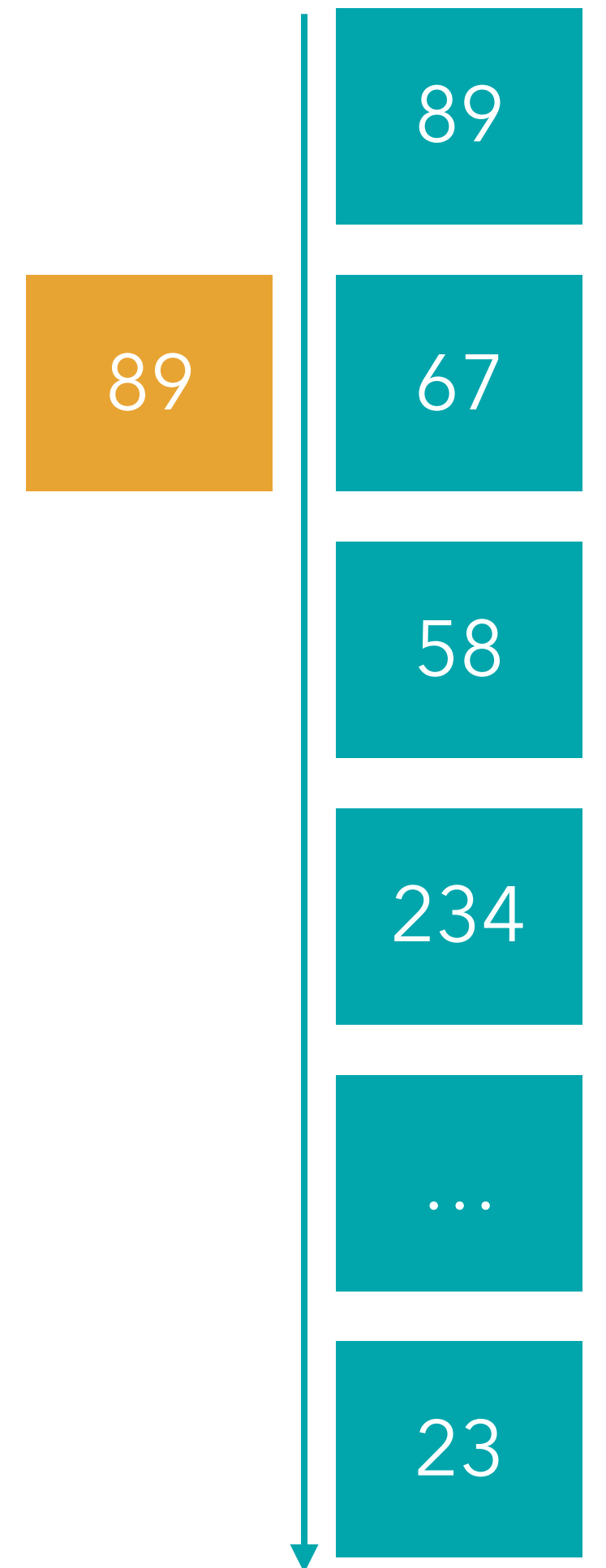
Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten
`var myNumbers : number [] = [89,67,58,2,93,...,234,23];`
2. Die erste Zahl ist zunächst die Größe
`var highestNumber : number = myNumbers[0]`
3. Durchlaufe nun das Array
`for-Schleife über alle Zahlen`



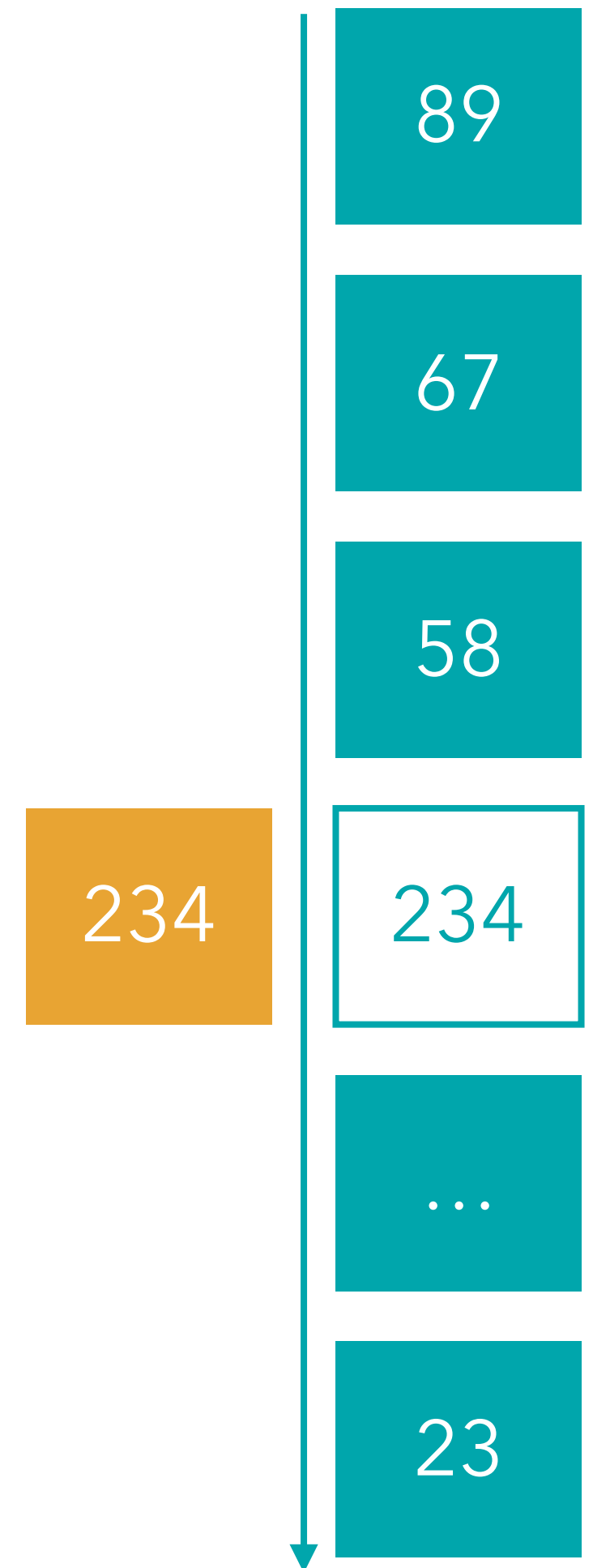
Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten
`var myNumbers : number [] = [89,67,58,2,93,...,234,23];`
2. Die erste Zahl ist zunächst die Größe
`var highestNumber : number = myNumbers[0]`
3. Durchlaufe nun das Array
for-Schleife über alle Zahlen
 - 3.1. Prüfe, ob die aktuelle Zahl größer als die bisher größte Zahl ist.
`if (myNumbers[i] > highestNumber)`



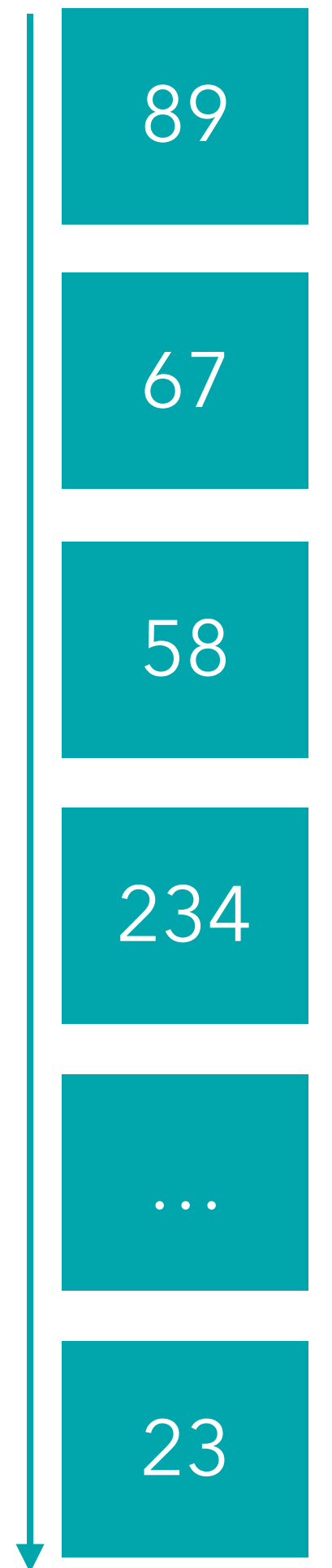
Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten
`var myNumbers : number [] = [89,67,58,2,93,...,234,23];`
2. Die erste Zahl ist zunächst die Größe
`var highestNumber : number = myNumbers[0]`
3. Durchlaufe nun das Array
for-Schleife über alle Zahlen
 - 3.1. Prüfe, ob die aktuelle Zahl größer als die bisher größte Zahl ist.
`if (myNumbers[i] > highestNumber)`
 - 3.1.1. Wenn Bedingung erfüllt, speichere die aktuelle Zahl als die größte ab
`highestNumber = myNumbers[i]`



Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten
`var myNumbers : number [] = [89,67,58,2,93,...,234,23];`
2. Die erste Zahl ist zunächst die Größe
`var highestNumber : number = myNumbers[0]`
3. Durchlaufe nun das Array
for-Schleife über alle Zahlen
 - 3.1. Prüfe, ob die aktuelle Zahl größer als die bisher größte Zahl ist.
`if (myNumbers[i] > highestNumber)`
 - 3.1.1. Wenn Bedingung erfüllt, speichere die aktuelle Zahl als die größte ab
`highestNumber = myNumbers[i]`
4. Nach Durchlauf des Arrays, ist der zuletzt gespeicherte höchste Wert auch gleichzeitig der höchste Wert aller Zahlen
`return highestNumber`



234

Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten

→ `var myNumbers : number [] = [89,67,58,2,93,...,234,23];`

2. Die erste Zahl ist zunächst die Größe

`var highestNumber : number = myNumbers[0]`

3. Durchlaufe nun das Array

`for-Schleife über alle Zahlen`

- 3.1. Prüfe, ob die aktuelle Zahl größer als die bisher größte Zahl ist.

`if (myNumbers[i] > highestNumber)`

- 3.1.1. Wenn Bedingung erfüllt, speichere die aktuelle Zahl als die größte ab

`highestNumber = myNumbers[i]`

4. Nach Durchlauf des Arrays, ist der zuletzt gespeicherte höchste Wert auch gleichzeitig der höchste Wert aller Zahlen

`return highestNumber`

function getHighestNumber (numbers : number[]) : number

89

67

58

234

...

23

234

Geben Sie aus einem beliebig langen Array an numerischen Werten den höchsten Wert auf die Konsole aus.

1. Initialisiere ein numerisches Array mit einer beliebigen Anzahl an Werten

→ `var myNumbers : number [] = [89,67,58,2,93,...,234,23];`

2. Die erste Zahl ist zunächst die Größe

`var highestNumber : number = myNumbers[0]`

3. Durchlaufe nun das Array

`for-Schleife über alle Zahlen`

- 3.1. Prüfe, ob die aktuelle Zahl größer als die bisher größte Zahl ist.

`if (myNumbers[i] > highestNumber)`

- 3.1.1. Wenn Bedingung erfüllt, speichere die aktuelle Zahl als die größte ab

`highestNumber = myNumbers[i]`

4. Nach Durchlauf des Arrays, ist der zuletzt gespeicherte höchste Wert auch gleichzeitig der höchste Wert aller Zahlen

`return highestNumber`

function `getHighestNumber (numbers : number[]) : number`

5. Löse die Funktion aus: `console.log (getHighestNumber(myNumbers));`

89

67

58

234

...

23

234

Beispiel Algorithmus III

Aufgabe: Ermitteln Sie aus einer beliebigen Anzahl numerischer Werte deren Summe und geben Sie diese auf die Konsole aus.

5 Minuten Zeit – dann Besprechung

Prüfung

06. Februar, 2019 | Aula



Themenkomplexe und Aufteilung

- HTML5/CSS – ca. 50%
- TypeScript – ca. 50%

Jeweils alle in Unterricht und Praktikum behandelten Themen

90 Punktesystem

> 50% der Gesamtpunkte: bestanden – 4,0 :)

Art der Fragestellungen

- Verständnis- und Wissensfragen: Multiple Choice und freie Antworten
- Code verstehen: lesen und interpretieren
- Code verändern
- Code erzeugen: Algorithmus und Implementierung
- Logik

Rahmenbedingungen Prüfung

Zulässig

2 Seiten DIN A4 handschriftliche, persönliche
Zusammenfassungen
Beidseitig beschrieben

Nicht zulässig

Weitere Hilfen, insbesondere digitale Medien und
Smartphones

Last but not least



Praxissemester @T7Lab



Vielen Dank.