

Test – Informatik GK 12 – Grundlagen und Felder

1.) Analysieren Sie folgenden Quelltext.

(Datentypen, Grundstrukturen, Eingabe, Ausgabe – ggf. als Kommentar der Codezeilen)

```
var array = [1,1,7,1,1];

var z=0;
for(var i=0; i<array.length; i++){

    if(array[i] == 1){
        z=z+1;
    }
}
return z;
}
```

Feld aus ganzen Zahlen definiert
(eingegeben)

Deklaration z („Zähler“)

Zählschleife mit Laufindex i bis
Feldlänge

innerhalb der Schleife:

Verweigung: wenn an der Stelle i
des Feldes eine 1 steht, dann
erhöhe Zähler z um 1

Ausgabe Anzahl der Einsen im
Feld

2.) Vervollständigen Sie die gegebene Funktion, die ein Array als Parameter entgegennimmt und die **Summe** sowie **den Mittelwert** aller Elemente zurückgibt.

```
function summe(array){

    // ....    var s = 0;

    for(var i=0; i<array.length; i++){

        s = s + array[i];
    }
    var mw = s / array.length;
    console.log (s);
    console.log (mw);

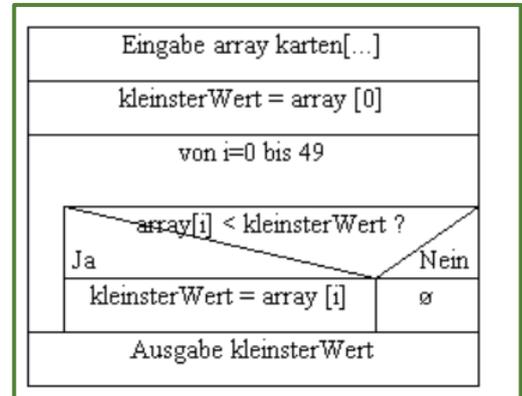
}
```

```
var array = [4,5,7,8,9,10];
```

```
summe(array);
```

- 3.) In einem Programm wurde ein Feld mit dem Namen „karten“ definiert: `var karten = [...]`
Es enthält bereits 50 verschiedene ganzzahlige Werte.

Erstellen Sie ein Struktogramm für einen Algorithmus, welcher den kleinsten Wert dieses Felds ermitteln soll.



- 4.) Erläutern Sie anhand der Abbildung den grundlegenden Aufruf von Funktionen.

```

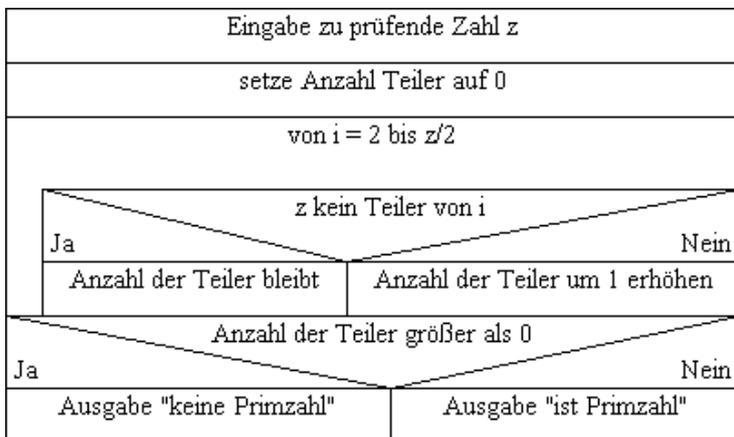
x 1 function Max3(a, b, c) {
  2   if (a>b) {
  3     if (a>c) {console.log("Maximum: " + a)}
  4     else {console.log("Maximum: " + c)}
  5   }
  6   else
  7     if (b>c) {console.log("Maximum: " + b)}
  8     else {console.log("Maximum: " + c)}
  9 }
10
x 1 Max3(44, 33, 55);
  2

```

siehe hier:

https://programmingwiki.de/PMG2224_1/Modularisierung

- 5.) Übertragen Sie den im Struktogramm vorliegenden Algorithmus in ein Funktion in der Programmiersprache JavaScript.



```

function PZ(z){
var AnzT = 0;
for(var i=2; i<=z/2; i++){
  if(z % i != 0( {
    AnzT = AnzT}
  else {AnzT = AnzT + 1}
}
if(AnzT>0){console.log („keine Primzahl“)}
else {console.log („PZ“)}
}

```

```
PZ(17);
```