

Fachinformatiker/-in Fachrichtung
Anwendungsentwicklung

FA 234

Entwicklung und Umsetzung von
Algorithmen

Bearbeitungszeit:
90 Minuten

Verlangt:

Alle Aufgaben

Hilfsmittel: Nicht programmierter Taschenrechner

Bewertung: Die Bewertung der einzelnen Aufgaben ist durch Punkte näher vorgegeben.

Zu beachten: Die Prüfungsunterlagen sind vor Arbeitsbeginn auf Vollständigkeit zu überprüfen.

Dieser Aufgabensatz besteht aus:

- den Aufgaben 1 bis 3
- den Anlagen 1 bis 5

Bei Unstimmigkeiten ist sofort die Aufsicht zu informieren.

Klare und übersichtliche Darstellung der Rechengänge mit Formeln und Einheiten wird entscheidend mitbewertet.

Projekt: SmartHome**Projektbeschreibung:**

Die Firma SmartHome bietet zahlreiche Möglichkeiten, um die Steuerung in einem Haus zu automatisieren. In einem Vorzeigehaus gibt es Sensoren und Aktoren, deren Zusammenspiel mit einem Simulationsprogramm getestet werden soll. Ihre Aufgabe ist die **Erstellung einer Webseite**, mit der die aktuellen Zustände angezeigt werden können, die **Erstellung eines Programms**, um die Temperaturmesswerte zu überprüfen und die **Konfiguration von Temperatursensoren**.

Aufgabe 1 (Anlage 1)**20**

- 1.1 Um einen ersten Eindruck von den Möglichkeiten zu bekommen, soll die in Anlage 1 dargestellte Seite (ohne das Logo „Smart Home“) erstellt werden. 8
Geben Sie den entsprechenden Quelltext an.
Beachten Sie dabei:
- Die **Radiobuttons** sollen entsprechend der Abbildung vorgelegt sein.
 - Das Licht kann hier nur „an“ oder „aus“ sein.
 - Übersicht...“ ist Überschrift erster Ordnung.
 - „Aktor“ und „Sensor“ sind fett dargestellt.
 - Die **Hintergrundfarbe** ist orange.
- 1.2 Die angezeigte Temperatur soll regelmäßig aktualisiert werden. Die aktuellen Werte liegen 4
unter der Adresse **http://192.168.178.35/messung/temperatur.json** bereit
Beschreiben Sie eine Möglichkeit, um die angezeigte Temperatur regelmäßig zu aktualisieren.
- 1.3 Die aktuelle Uhrzeit soll zusätzlich auf der Seite angezeigt werden. 2
Geben Sie zwei Möglichkeiten mit den jeweiligen Vor- und Nachteilen an, um die aktuelle Uhrzeit auf der Seite anzuzeigen.
- 1.4 Die Seite soll für die Verwendung mit einem mobilen Gerät optimiert werden. Immer wenn auf 4
dem Display **mehr als 600 Pixel in der Breite** zur Verfügung stehen, sollen Aktor und Sensor nebeneinander dargestellt werden (siehe Anlage 1). In diesem Fall soll der Hintergrund eine **gelbe Farbe** bekommen.
Beschreiben Sie die notwendigen Änderungen.
- 1.5 Oben rechts soll das Logo der Firma „Smart Home“ dargestellt werden, das als Bilddatei 2
(Logo.gif) vorhanden ist (siehe Anlage 1).
Geben Sie zwei Möglichkeiten an, um das Logo rechts mit einem Abstand von 30 Pixeln zum Seitenrand zu platzieren.

Aufgabe 2 (Anlage 2, 3, 4, 5)**55**

Die **Temperatur- und die Feuchtigkeitsverteilung** in der Sauna soll mit **4 Sensoren** überwacht werden. Ein Microcontroller liefert die Werte als **JSON-Datei**. Für die Entwicklung und die Präsentation des Programms soll die JSON-Datei aus **Anlage 2** verwendet werden. Ein Mitarbeiter hat bereits ein Struktogramm für diese Methode entwickelt (siehe **Anlage 3**). Die Methode liest aus der JSON-Datei **vier Messwerte** und bildet daraus den **Mittelwert**. Sollte die der Methode als Parameter übergebene **maximale Abweichung** von einem der Messwerte überschritten werden, dann soll diese Information zusammen mit dem ermittelten Mittelwert zurückgegeben werden.
Dafür wird der Mittelwert um ein Bit verschoben und das unterste Bit enthält dann die Fehlerinformation:

0: alle Messwerte innerhalb der zulässigen Toleranz

1: ein oder mehrere Messwerte außerhalb der zulässigen Toleranz

- 2.1 Geben Sie den Quelltext für die Methode zu dem Struktogramm (Anlage 3) in der an Ihrer Schule unterrichteten Programmiersprache an. 20
- 2.2 Um die Funktion der Methode zu testen, ist ein entsprechendes Hauptprogramm zu erstellen. Geben Sie den Quelltext an. 9
- 2.3 Für eine vollständige Überprüfung der Methode ist ein Whitebox-Testfall zu erstellen. Ergänzen Sie die Tabelle in Anlage 4 mit allen erforderlichen Variablen und Werten, um die Methode vollständig zu testen. 8
- Hinweis:
Sollte eine Variable während des Programmablaufs unterschiedliche Werte haben, so soll der letzte Wert der Variablen abgespeichert werden.
- 2.4 Ein Kunde möchte, dass aus dem Rückgabewert der Methode nicht nur auf das Abweichen eines beliebigen Messwerts geschlossen werden kann. Es soll aus dem einen Rückgabewert genau erkennbar sein, welcher Temperatursensor eine Abweichung liefert. Ergänzen Sie dafür das Struktogramm aus der Anlage 5. 10
- 2.5 Die JSON-Datei soll in Zukunft durch den Aufruf einer REST-API erstellt werden. Geben Sie 4 Befehle an, über die mit einer REST-API kommuniziert werden kann. 8

Aufgabe 3**15**

Zur Messung der Gebäudetemperaturen in verschiedenen Räumen werden Messmodule verwendet. Jedes Messmodul besteht aus einem Mikrocontroller und zwei elektronischen Temperatursensoren. Die elektronischen Temperatursensoren sind als Slave über einen I2C Bus mit dem Mikrocontroller als Master verbunden.

- 3.1.1 Bestimmen Sie für die beiden Temperatursensoren jeweils eine Slave-Write Adresse. Geben Sie diese als Binär- und Hexadezimalzahl an. 6

Hinweis:

Angaben zur Adressierung aus dem Datenblatt:

D7	D6	D5	D4	D3	D2	D1	D0
1	1	0	1	x	x	x	x

D3-D1 frei wählbarer Adressbereich, D0 = 1 Write

- 3.1.2 Die Temperatursensoren haben einen Messbereich von -50 bis +50°C. Der Messwert wird nach der Adresse als 8-Bit-Datenblock übertragen. Bestimmen Sie die minimal auflösbare Temperaturdifferenz. 3
- 3.2.1 Der Mikrocontroller wird über seinen On-Board WLAN Client in das Gebäude-WLAN eingebunden. Bisher wird WPA2-PSK Verschlüsselung und Authentifizierung im Gebäude-WLAN verwendet. Im Rahmen einer Schutzanalyse müssen Sie diese Verbindung beurteilen und gegebenenfalls eine Alternative vorschlagen. Begründen Sie Ihre Entscheidung. 3
- 3.2.2 Die Messwerte werden mit MQTT auf einem Broker zugänglich gemacht. Schlagen Sie dem Kunden Maßnahmen zur Sicherstellung der Vertraulichkeit, Authentizität und Integrität vor. 3

Anlage 1

zu Aufgabe 1.1



zu Aufgabe 1.4



Anlage 2

Messung.json

```
{
  "Sensor1":
  {
    "Nummer": 1,
    "Zeit": "2022-10-26T16:23:14",
    "Temperatur": 2,
    "Feuchtigkeit": 80
  },
  "Sensor2":
  {
    "Nummer": 2,
    "Zeit": "2022-10-26T16:25:06",
    "Temperatur": 3,
    "Feuchtigkeit": 83
  },
  "Sensor3":
  {
    "Nummer": 3,
    "Zeit": "2022-10-26T16:25:15",
    "Temperatur": 4,
    "Feuchtigkeit": 79
  },
  "Sensor4":
  {
    "Nummer": 4,
    "Zeit": "2022-10-26T16:22:51",
    "Temperatur": 3,
    "Feuchtigkeit": 81
  }
}
```


Anlage 3

Funktion TempWert

ÜbergabeParameter: schwelle: Datentyp double Lokale Variablen: datZgr: Zeiger für Dateizugriff inhalt: Inhalt der gelesenen Datei inhaltJson: Inhalt der gelesenen Datei im JSON-Format temp[4]: Feld von 4 Temperaturmesswerten tempMittel: Durchschnittswert der 4 gemessenen Temp. abweichung: größte Abweich (gemessene und Durchschn.) rueck: Rückgabewert	
datZgr = Zeiger auf zum Lesen geöffnete Datei "messung.json"	
inhalt=lesenVonDatZgr()	
Datei von datZgr schließen	
inhaltJson=ZuJsonWandeln(inhalt)	
temp[0]=InhaltJson(Temperatur von Sensor1) temp[1]=InhaltJson(Temperatur von Sensor2) temp[2]=InhaltJson(Temperatur von Sensor3) temp[3]=InhaltJson(Temperatur von Sensor4)	
tempMittel=0	
von i=0 solange i<4	
tempMittel = tempMittel + temp[i]	
tempMittel = tempMittel / 4	
abweichung=0	
von i=0 solange i<4	
<div style="text-align: center;"> $\begin{matrix} \text{abweichung} * \text{abweichung} \\ < \\ (\text{tmpMittel} - \text{temp}[i]) * (\text{tmpMittel} - \text{temp}[i]) \end{matrix}$ </div> <div style="display: flex; justify-content: space-between;"> Ja Nein </div>	
abweichung = tempMittel-temp[i]	-
<div style="text-align: center;"> $\begin{matrix} \text{abweichung} < 0 \\ ? \end{matrix}$ </div> <div style="display: flex; justify-content: space-between;"> Ja Nein </div>	
abweichung = abweichung*(-1)	-
rueck = 2*Ganzzahl(tempMittel)	
<div style="text-align: center;"> $\begin{matrix} \text{abweichung} > \text{schwelle} \\ ? \end{matrix}$ </div> <div style="display: flex; justify-content: space-between;"> Ja Nein </div>	
rueck = rueck + 1	-
Rückgabe von rueck	

Anlage 4

Bitte geben Sie dieses Blatt mit Ihren Lösungen ab.

Name, Vorname: _____ Klasse: _____

Whiteboxtest

PrüfNR: _____

Name: _____

schwelle	0,5
temp[0]	4
temp[1]	3
temp[2]	3
temp[3]	2
tempMittel	3
Abweichung	1
rueck	7
Zurückgegebener Wert	7

Bitte geben Sie dieses Blatt mit Ihren Lösungen ab.

Name, Vorname: _____ Klasse: _____

StruktoJSON

von i=0 solange i<4	
$\begin{array}{c} \text{abweichung} * \text{abweichung} \\ < \\ (\text{tmpMittel} - \text{temp}[i]) * (\text{tmpMittel} - \text{temp}[i]) \end{array}$ <div>?</div>	
Ja	Nein
abweichung = tmpMittel - temp[i]	-
$\begin{array}{c} \text{abweichung} < 0 \\ < \\ \end{array}$ <div>?</div>	
Ja	Nein
abweichung = abweichung * (-1)	-

Fachinformatiker/-in Fachrichtung Anwendungsentwicklung
Entwicklung und Umsetzung von Algorithmen**FA 234****Lösungsvorschläge:**

Lösungsvorschläge sind in der Regel Vorschläge der einreichenden Schulen; sie sind im Wortlaut nicht bindend. Anderslautende, aber zutreffende Antworten sind ebenfalls als richtig zu werten.

**Nur für die Hand
des Prüfers!
Punkte**

Projekt: SmartHome**Aufgabe 1****20**

- 1.1 8
- ```
<!DOCTYPE html>
<html lang="de">
 <head>
 <meta charset="UTF-8">
 <title> Musterhaus</title>
 </head>

 <body style=" background-color: yellow; ">
 <H1> Übersicht Musterhaus </H1>
 <form>
 Aktor

 Licht Wohnzimmer
 <label for="wohn1"> an: </label>
 <input type="radio" id="wohn1" name="wlicht" value="Wan">
 <label for="wohn2"> aus: </label>
 <input type="radio" id="wohn2" name="wlicht" value="Waus" checked>

 Sensor

 Temperatur Wohnzimmer: 24
 </form>
 </body>
</html>
```
- 1.2 4
- Die angezeigte Temperatur kann sich immer wieder aktualisieren, wenn
- a. Die Webseite regelmäßig aktualisiert wird (z.B. mit <meta http-equiv="refresh" content="30"> Dabei wird die Seite alle 30s aktualisiert.
- b. Dazu muss ein JavaScript eingebunden werden, das den Wert aus der Datei liest und z.B. mit einem <span>-Tag auf die Webseite überträgt.
- 1.3 2
- Die aktuelle Uhrzeit kann
- I. Z. B. mit Hilfe von einer clientseitigen Skriptsprache, z. B. JavaScript vom lokalen Rechner ausgelesen und mit Hilfe eines <span>-Tags auf der Seite dargestellt werden.
  - II. Mit einer serverseitigen Skriptsprache vom Server ermittelt werden und im HTML-Dokument z. B. als IFRAME eingebunden werden.
  - III. Als IFRAME das Ergebnis des Aufrufs einer REST-API von einem entsprechenden Server im Internet dargestellt werden.



zu 1.3

Möglichkeit	Vorteil	Nachteil
I.	Kein zusätzlicher Netzwerkverkehr, da nur Client betroffen geringster Aufwand	Die Zeit muss auf diesem lokalen Rechner stimmen bzw. regelmäßig überprüft werden.
II.	Auf dem lokalen Server ist davon auszugehen, dass immer die Zeit aktuell ist.	Zusätzliches serverseitiges Skript notwendig und lokaler Netzwerkverkehr
III.	Es kann ein sehr genauer Zeitserver, z.B. auch von einem anderen Land verwendet werden	Client muss direkten Zugang zum Internet haben, maximaler Netzwerkverkehr, zusätzliches „Loch“ in der Firewall

- 1.4 Wenn die 2 Blöcke „Aktor“ und „Sensor“ jeweils in ein „div“ gepackt werden, kann mit dem Tag „@media screen and (max-min: 600px)“ der Fall zusätzlich in der CSS-Datei definiert werden, dass die beiden Blöcke nur jeweils 50% des Anzeigebereichs nutzen und so nebeneinander dargestellt werden. Zusätzlich muss dann definiert werden, dass, falls die Bedingung des „Media-Tags“ nicht erfüllt ist, die Blöcke 100% des Anzeigebereichs nutzen. Eine andere Möglichkeit wäre die CSS-Eigenschaft „spalte (Column) zu verwenden. Auch hier wird ein „Media-Tag“ benötigt. 4
- 1.5 Dem Bild kann z. B. mit Hilfe von CSS die Eigenschaften „position: fixed;“ oder „right: 30px;“ zugewiesen werden. 2



## Aufgabe 2

55

2.1 Programm in Python: 20

```

def auswertung(schwelle):
 import json
 datei = open('messung.json','r')
 me = datei.read()
 print(me)
 datei.close()

 mess1 = json.loads(me)

 temp=[1,2,3,4]
 temp[0]=mess1["Sensor1"]["Temperatur"]
 temp[1]=mess1["Sensor2"]["Temperatur"]
 temp[2]=mess1["Sensor3"]["Temperatur"]
 temp[3]=mess1["Sensor4"]["Temperatur"]

 tempMittel=0
 for i in range(0,4):
 tempMittel+=temp[i]
 tempMittel=tempMittel/4;

 abweichung=0
 for i in range(0,4):
 if (abweichung*abweichung) < ((tempMittel-temp[i])* (tempMittel-temp[i])):
 abweichung=(tempMittel-temp[i])
 if abweichung <0:
 abweichung = abweichung*(-1)
 rueck=(int)tempMittel*2
 if abweichung>schwelle:
 rueck=rueck+1
 return rueck2

```

2.2 # Programm in Python 9

```

ergebnis=auswertung(4)
if ergebnis%2==1:
 print("Die Messwerte sind zu weit auseinander")
else:
 print("Die Messwerte sind in Ordnung")
ergebnis=ergebnis-1
ergebnis=ergebnis /2
print(ergebnis)

```

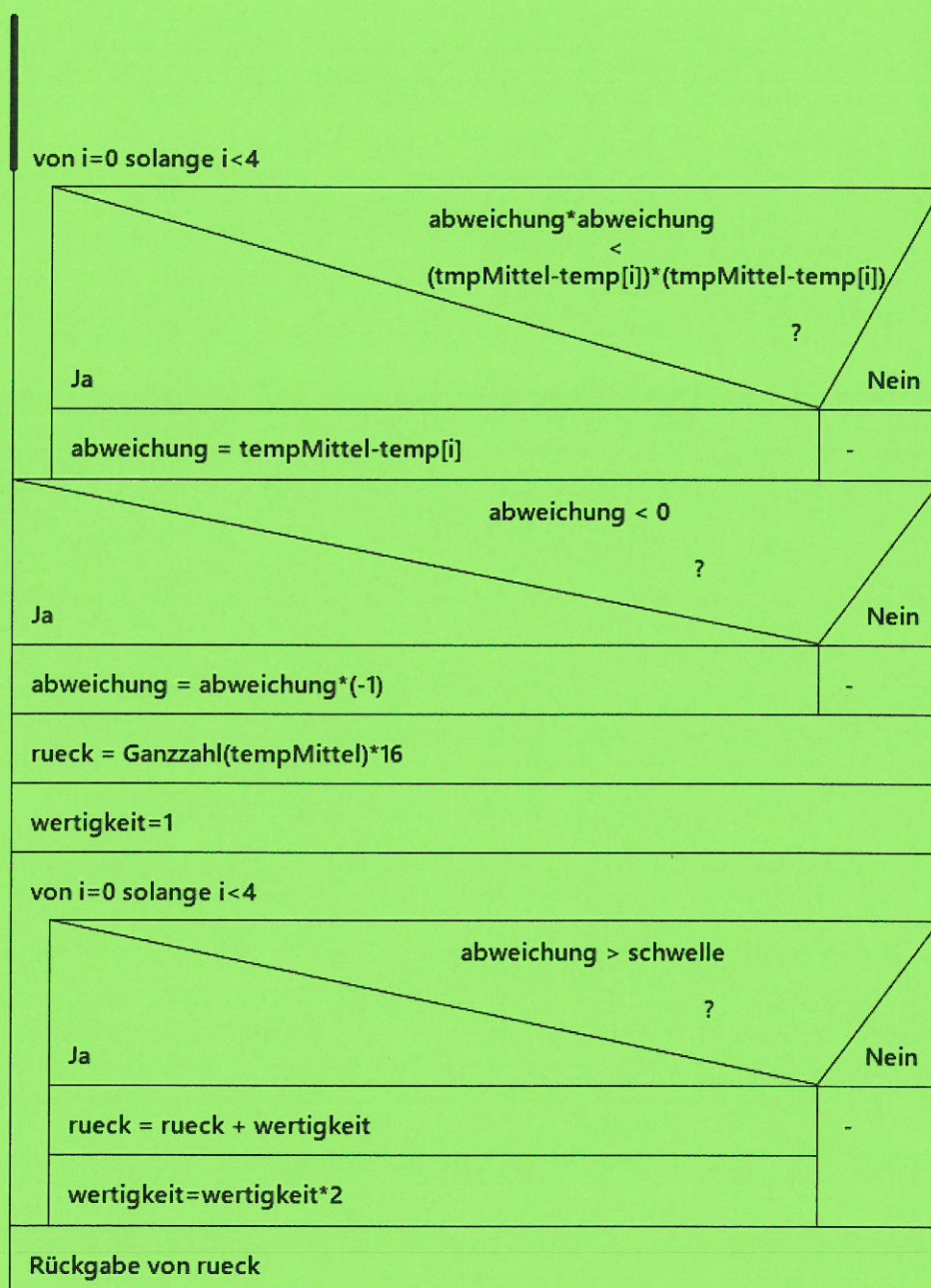
2.3 Whiteboxtest 8

schwelle	0,5
temp[0]	4
temp[1]	3
temp[2]	3
temp[3]	2
tempMittel	3
Abweichung	1
rueck	61
Zurückgegebener Wert	61



2.4

10



2.5

Befehle:

- GET – Abrufen einer bestimmten Ressource
- POST – Erstellen einer neuen Ressource
- PUT – Aktualisieren einer vorhandenen Ressource
- DELETE – Löschen einer vorhandenen Ressource

8



<b>Aufgabe 3</b>		<b>15</b>
3.1.1	1. Sensor z. B. 1101 0001 bzw. D1 2. Sensor z. B. 1101 0011 bzw. D3	6
3.1.2	Temperaturdifferenz: $101K/(256) = 0,395K$	3
3.2.1	WPA2 ist veraltet und sollte mindestens durch WPA3 besser WPA4 ersetzt werden. Pre-Shared Key ist anfällig, da nur ein Passwort für alle User verwendet wird. Z. B. kann EAP bzw. RADIUS als Alternative vorgeschlagen werden.	3
3.2.2	Z. B. Benutzer mit Passwort, TLS, etc.	3