|  |  |  |
| --- | --- | --- |
| **Zentralabitur 2024** | **Informatik** | **Material für Prüflinge** |
| **Block 1: Aufgabe A** | **gA** | **Prüfungszeit\*: 250 min** |

**Name:** **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Klasse:** **\_\_\_\_\_\_\_**

Inhaltsverzeichnis

[Material 3](#_Toc162335427)

[zu Aufgabenteil a) 3](#_Toc162335428)

[zu Aufgabenteil b) 4](#_Toc162335429)

[zu den Aufgabenteilen e) ‒ g) 5](#_Toc162335430)

[zu den Aufgabenteilen e) - g) 6](#_Toc162335431)

[zu Aufgabenteil e) 7](#_Toc162335432)

[zu den Aufgabenteilen j) ‒ k) 8](#_Toc162335433)

# Material

## zu Aufgabenteil a)

**Abbildung 1:** Klassenkarte der Klasse **Umwelt**

|  |
| --- |
| **Umwelt** |
| - xPos: Ganzzahl  - yPos: Ganzzahl  - datum: Zeichenkette  - gebiet: Zeichen  - luftTemperatur: Fließkommazahl  - wolfHF: Ganzzahl // Herzfrequenz  … |
| c Umwelt(xPos: Ganzzahl, yPos: Ganzzahl, datum: Zeichenkette, luftTemperatur: Fließkommazahl, wolfHF: Ganzzahl)  + getDatum(): Zeichenkette  + gebietBestimmen()  + getGebiet(): Zeichen  + getLuftTemperatur(): Fließkommazahl  + getWolfHF(): Ganzzahl  … |

## zu Aufgabenteil b)

#### Abbildung 2:

Tabelle zur Bestimmung der Gebiete Feld (F), Wald (W) und Stadt (S) basierend auf der durch (xPos, yPos) gegebenen aktuellen Position

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **xPos** von aktueller Position | |
|  |  | **xPos** ≤ 10.000 | 10.000 < **xPos** ≤ 15.000 |
| **yPos** von aktueller Position | **yPos** ≤ 20.000 | W | F |
| **yPos** > 20.000 | F | S |

## zu den Aufgabenteilen e) ‒ g)

#### Abbildung 3: Struktogramm der Operation gSort

**Hinweis:** Mit dem Vergleichsoperator ≥ wird im Struktogramm auch das lexikografische Vergleichen bezeichnet

**gSort()**

anzahlWerte data.getLength()

wahr

anzahlWerte > 0

pos 1

//Protokollierung

wiederhole solange pos < anzahlWerte

wahr

pos = 0

falsch

falsch

falsch

tausche data.getItem(pos-1)  
mit data.getItem(pos)

verringere pos um 1

//Protokollierung

erhöhe pos um 1

erhöhe pos um 1

wahr

gebiet von data.getItem(pos-1) gebiet von   
data.getItem(pos)

## zu den Aufgabenteilen e) - g)

#### Abbildung 4:

Beispiel einer global definierten dynamischen Reihung **data** vom Typ **Umwelt**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Index** | **0** | **1** | **2** | **3** | **4** |
|  | **u1: Umwelt**  gebiet = 'F'  luftTemperatur = 13.0  ... | **u2: Umwelt**  gebiet = 'S'  luftTemperatur = 17.0  ... | **u3: Umwelt**  gebiet = 'F'  luftTemperatur = 14.5  ... | **u4: Umwelt**  gebiet = 'W'  luftTemperatur = 14.0  ... | **u5: Umwelt**  gebiet = 'N'  luftTemperatur = 14.5  ... |

## zu Aufgabenteil e)

#### Vorlage für die Tracetabelle

Hinweis zur Protokollierung des Ablaufs der Operation **gSort**: In einer neuen Zeile müssen nur die Werte eingetragen werden, die sich ändern.

#### Abbildung 5:

Vorlage zur Dokumentation des Ablaufs der Operation **gSort**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **pos** | **Index 0** | **Index 1** | **Index 2** | **Index 3** | **Index4** |
| 1 | u1, F | u2, S | u3, F | u4, W | u5, N |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## zu den Aufgabenteilen j) ‒ k)

#### Abbildung 6:

Codetabelle für die Gebiete mit ihren durchschnittlichen Häufigkeitswerten

|  |  |  |  |
| --- | --- | --- | --- |
| **Gebiets-name** | **Bezeichnung** | **2-Bit Codierung** | **durchschnittliche Häufigkeit** |
| Feld | F | 00 | 65 % |
| Wald | W | 01 | 30 % |
| Stadt | S | 10 | 4 % |
| unbekanntes Gebiet | N | 11 | 1 % |