

1. Multiple-Choice-Aufgaben

8.4 / 10 Punkte

Bitte wählen Sie *alle* zutreffenden Antwortmöglichkeiten aus. Es können beliebig viele Antwortmöglichkeiten zutreffen, auch alle oder keine.

Aufgabe 1.1.

2 / 2 Punkte

Welche Typen sind elementare Datentypen?

☐ Scanner

☒ char

☒ byte

☐ String

☒ boolean

Aufgabe 1.2.

2.4 / 3 Punkte

Welche Aussagen in Bezug auf Arrays in Java sind korrekt?

☐ Bei einem mehrdimensionalen Array in Java müssen die Unterdimensionen (ab der zweiten) die gleiche Größe haben.

☒ Die einzelnen Komponenten eines Arrays heißen Elemente und sind alle vom selben deklarierten Datentyp (Elementtyp).

☐ Die Anweisung `int[][] matrix = new int[3][4];` erzeugt drei Arrays vom Elementtyp `int[]` und eines vom Elementtyp `int`.

☐ Nach den Anweisungen

```
int[] original = {1,2,3};  
int[] copy = original;  
copy[0] = 3;
```

liefert der Ausdruck `original[0]` den Wert 3.

☒ Die Länge eines Arrays ist nach dessen Erzeugung nicht mehr änderbar.

Aufgabe 1.3.

2.4 / 3 Punkte

Welche Aussagen in Bezug auf Rekursion sind korrekt?

- ☐ Falls eine rekursive Methode ausschließlich lokale Variablen benutzt (dazu zählen auch ihre formalen Parameter), also keine Seiteneffekte hat, können sich die rekursiven Methodenausführungen nicht gegenseitig beeinflussen.
- ☒ Fundiertheit und Fortschritt sind notwendig damit rekursive Methoden ein korrektes Ergebnis berechnen.
- ☒ Bei rekursivem Methodenaufruf wird die Ausführung der aufrufenden Methode unterbrochen, bis die Ausführung der aufgerufenen Methode beendet ist.
- ☐ Bei der vollständigen Induktion wird nur durch die Korrektheit des Induktionsanfangs die Induktionsannahme bewiesen.
- ☐ Rekursion ist in der Regel schneller als eine Schleife, da es nicht zu wiederholten Berechnungen kommt.

Aufgabe 1.4.

1.6 / 2 Punkte

Welche Aussagen sind in Bezug auf Datenstrukturen in Java sind korrekt?

- ☐ Elemente in einer Queue sind immer nach Ihrem Wert sortiert, nicht aber in einer Map.
- ☒ Eine Queue arbeitet nach dem LIFO Prinzip.
- ☒ Indizes in einem Array können nur vom Typ `int` sein.
- ☐ Schlüssel in einer Map können nur vom Typ `Integer` sein.
- ☒ In einer einfach verketteten Liste ist die Nachfolgerreferenz des letzten Elements `null`.

2. Auswahlaufgaben

6 / 15 Punkte

Folgende Implementierungen der Methode `countOdd` sind syntaktisch korrekt. Die Methode soll die Anzahl von ungeraden Zahlen im Intervall `x` bis `y` (einschließlich) berechnen. Vorbedingung ist `x > 0`, `x <= y`. Geben Sie an welche Aussage auf die jeweilige Implementierung zutrifft.

Aufgabe 2.1.

0 / 3 Punkte

```
public static int countOdd(int x, int y) {  
    return (x != y) ? ( countOdd(x, y - 1) + ((y%2 == 1) ? 1 : 0) )  
        : ((y%2 == 1) ? 1 : 0);  
}
```

Welche Aussage trifft hier zu?

- ☐ erzeugt einen Laufzeitfehler bei bestimmten gültigen Eingaben (Laufzeitfehler)
- ☒ bei allen gültigen Eingaben kein Laufzeitfehler aber liefert einen falschen Wert bei bestimmten gültigen Eingaben (falscher Wert)
- ☐ liefert für alle gültigen Argumente das korrekte Ergebnis (korrekt)

Aufgabe 2.2.

0 / 3 Punkte

```
public static int countOdd(int x, int y) {  
    return (y%2==0) ? ( x == y ? 1 : (countOdd(x, y - 1) + 1 ) )  
        : ((x != y) ? countOdd(x, y - 1) : 0);  
}
```

Welche Aussage trifft hier zu?

- ☒ Laufzeitfehler
- ☐ falscher Wert
- ☐ korrekt

Aufgabe 2.3.

3 / 3 Punkte

```
public static int countOdd(int x, int y) {  
    return (x > y) ? 0 : (( (y%2 == 1) ? 1 : 0 ) + countOdd(x, y - 1));  
}
```

Welche Aussage trifft hier zu?

- ☐ Laufzeitfehler
- ☐ falscher Wert
- ☒ korrekt

Aufgabe 2.4.

0 / 3 Punkte

```
public static int countOdd(int x, int y) {  
    return ( (y%2 == 1) ? 1 : 0 ) + ((x >= y) ? 0 : countOdd(x, y - 1));  
}
```

Welche Aussage trifft hier zu?

- ☒ Laufzeitfehler
- ☐ falscher Wert
- ☐ korrekt

Aufgabe 2.5.

3 / 3 Punkte

```
public static int countOdd(int x, int y) {  
    return (x == y) ? ( (x%2 == 1) ? 1 : 0 )  
        : (countOdd(x + 1, y) + ((x%2 == 1) ? 1 : 0));  
}
```

Welche Aussage trifft hier zu?

- ☐ Laufzeitfehler
- ☐ falscher Wert
- ☒ korrekt

3. Auswahlaufgaben zur Ergänzung von Methoden

0 / 12 Punkte

In den Methoden sind die Buchstaben A, B, C, D, E und F jeweils durch Ausdrücke zu ersetzen. Bitte wählen Sie für jeden dieser Buchstaben genau eine zutreffende Antwortmöglichkeit. Die Methoden müssen sich so verhalten wie angegeben. Punkte gibt es nur, wenn die gewählten Antwortmöglichkeiten zusammenpassen.

Aufgabe 3.1.

0 / 12 Punkte

Gegeben ist eine Matrix in der Größe 480x640 welche als zweidimensionales `int` Array gespeichert ist. Die erlaubten Werte der Matrix liegen in dem Bereich von 0 bis 254 (einschließlich).

```
int[][] matrix = new int[480][640];
```

Der Ausruf `int[] histo = histogram(matrix)` berechnet für Array ein Array vom Typ `int[]` eine Häufigkeitstabelle, d.h. die Ausdrücke `histo[10]` und `histo[225]` liefert beispielsweise die Häufigkeit des Wertes 10 bzw. 225 in `matrix`.

```
public static int[] histogram(int[][] matrix){
    if (matrix == null) {
        return A;
    }
    int[] result = B
    for (int i = 0; i < C; i++) { result[i] = 0; }
    for (int col = 0; col < D; col++) {
        for (int row = 0; row < E; row++) {
            F
        }
    }
    return result;
}
```

Füllen Sie die Stellen im Code richtig ein.

A:

- ☒ 0 ☐ -1 ☐ null

B:

- ☐ new int[640]; ☒ new int[255]; ☐ new int[480];

C:

- ☐ 640 ☒ 255 ☐ 480

D:

- ☐ 640 ☐ 255 ☒ 480

E:

- ☒ 640 ☐ 255 ☐ 480

F:

☐ `result[matrix[col][row]] += matrix[col][row];`

☐ `result[col][row] += matrix[col, row];`

☒ `result[matrix[col][row]++];`

☐ `result[matrix[col][row]]++;`