

**KLAUSUR****PROGRAMMIERUNG 1****03. FEBRUAR 2020****Bedingungen der Klausur:**

1. Wählen Sie einen neuen Workspace! Es darf nur das in dieser Klausur bearbeitete Projekt in Ihrem Workspace enthalten sein! Ein weiteres Projekt in Ihrem Workspace führt zu einem automatischen Ausschluss aus der Bewertung!
2. Die Nutzung von Handys, Abschreiben oder Kommunikation mit anderen während der Klausur ist nicht zulässig! Handys müssen ausgeschaltet und in der Tasche sein.
3. Ein handgeschriebenes A4-Blatt ist zugelassen.
4. Eclipse und ein Browser sind die einzigen Anwendungen, die auf Ihrem Rechner geöffnet sein dürfen. Im Browser darf nur die Moodle-Seite für die Klausur-Aufgabe geöffnet sein. Jede andere geöffnete Seite führt zu einem automatischen Ausschluss aus der Bewertung!
5. Am Ende der Prüfung: Öffnen Sie den Dateexplorer, wechseln Sie in Ihren Workspace, zippen Sie Ihr gesamtes Projekt und laden Sie die zip-Datei in Moodle hoch (bei Aufgabe Klausur1)!
6. Es sind insgesamt 56 Punkte zu erzielen (Teil 1: 21 Pkt., Testklasse: 6 Pkt., Teil 2: 22 Pkt., Testklasse: 4 Pkt., fehlerfreies Programm: 3 Pkt.).

**Notenspiegel:**

<b>Note</b>	<b>1,0</b>	<b>1,3</b>	<b>1,7</b>	<b>2,0</b>	<b>2,3</b>	<b>2,7</b>	<b>3,0</b>	<b>3,3</b>	<b>3,7</b>	<b>4,0</b>	<b>5,0</b>
<b>Punkte</b>	>52,5	52.5	50.0	47,0	44,0	41.5	38.5	36.0	33.0	30.0	<28
		– 50.5	– 47.5	– 44.5	– 42.0	– 39,0	– 36.5	– 33.5	– 30.5	– 28.0	

**Vorbereitung:**

1. Wechseln Sie in einen **leeren** Workspace!
2. Legen Sie ein neues Java-Projekt an!
3. Schreiben Sie Ihre Klassen im package **klausur**!

## Teil 1 (Klasse Lottery)

21 Punkte

<p>Schreiben Sie eine Klasse <b>Lottery</b>.</p> <p><i>Lottery steht für eine Lotterie, bei der aus 9 Zahlen (1..9) 5 Zahlen zufällig gelost werden (5 aus 9).</i></p>	1 Pkt.
<p>Objektvariable ist</p> <ul style="list-style-type: none"> <li>• <b>drawingResults</b> vom Typ <b>int []</b></li> </ul> <p>Die Objektvariable ist nur innerhalb der Klasse sichtbar! <i>Das Array dient zum Speichern der gezogenen 5 Zahlen.</i></p>	2 Pkt.
<p>Schreiben Sie für die Klasse <b>Lottery</b> einen parameterlosen Konstruktor. In diesem Konstruktor wird das Array <b>drawingResults</b> mit der Länge 5 erzeugt.</p>	2 Pkt.
<p>Schreiben Sie eine Objektmethode <b>contains (int number)</b>. Diese Methode gibt ein <b>true</b> zurück, wenn <b>number</b> in <b>drawingResults</b> enthalten ist und <b>false</b> sonst.</p>	2 Pkt.
<p>Schreiben Sie eine Objektmethode <b>drawing ()</b>. In dieser Methode werden die 5 Zufallszahlen gezogen (5 aus 9). Sie benötigen dafür ein Objekt der Klasse <b>Random</b> (<b>Random</b> muss aus <b>java.util</b> importiert werden). „Ziehen“ Sie nun zufällig 5 Zufallszahlen aus dem Bereich 1 . . 9 (1 und 9 inklusive) und speichern Sie diese im Array <b>drawingResults</b>.</p> <p><b>Achtung: Die gleiche Zahl darf nicht doppelt gezogen (gespeichert) werden! D.h. die 5 im Array gespeicherten Zufallszahlen müssen sich voneinander unterscheiden!</b></p>	4 Pkt.
<p>Schreiben Sie eine Objektmethode <b>sort ()</b>. Diese Methode sortiert das Array <b>drawingResults</b> aufsteigend (von klein nach groß).</p>	3 Pkt.
<p>(Über-)Schreiben Sie die Objektmethode <b>toString ()</b>, die das <b>drawingResult</b>-Array als ein <b>String</b> in folgender Form zurückgibt (Beispielwerte, falls eine 1, 3, 5, 6, 7 gezogen wurde):</p> <p>( 1 - 3 - 5 6 7 - - )</p> <ol style="list-style-type: none"> <li>1. das <b>dawingResult</b>-Array wird zunächst sortiert</li> <li>2. ist die Zahl im Array enthalten, wird sie „ausgegeben“</li> <li>3. ist die Zahl nicht enthalten, wird ein „-“ ausgegeben</li> </ol> <p>d.h. es werden immer die 5 gezogenen Zahlen ausgegeben und 4 Striche.</p>	3 Pkt.
<p>Schreiben Sie eine Objektmethode <b>print ()</b>, die mit Hilfe von <b>toString ()</b> eine Ausgabe auf der Konsole erzeugt.</p>	1 Pkt.

<p>Schreiben Sie eine Objektmethode <b>equal5 (Lottery lo)</b>. Diese Methode gibt ein <b>true</b> zurück, wenn bei dem aufrufenden Objekt die gleichen Zahlen gezogen wurden, wie bei <b>lo</b>. Sonst <b>false</b>.</p> <p>Tipp: Implementieren Sie die Methode am einfachsten so, dass Sie die <b>contains ()</b> - Methode verwenden. Sie können ansonsten auch die beiden <b>drawingResult ()</b> - Arrays erst sortieren und dann die sortierten Arrays miteinander vergleichen.</p>	3 Pkt
--	-------

**Testklasse****6 Punkte**

<p>Schreiben Sie eine Klasse <b>Testklasse</b> mit <b>main ()</b> -Methode. Erzeugen Sie in der <b>main ()</b> -Methode in einer Schleife 10 Objekte der Klasse <b>Lottery</b> und rufen (auch in der Schleife) jeweils die <b>drawing ()</b> und die <b>print ()</b> -Methode auf.</p> <p>Es entsteht folgende Ausgabe (Beispielwerte sind zufällig und unterscheiden sich!):</p> <pre>( 1 - 3 - 5 6 7 - - ) ( 1 2 3 - 5 - 7 - - ) ( 1 - 3 - - 6 7 8 - ) ( - - 3 4 5 6 - - 9 ) ( 1 2 3 4 - - - - 9 ) ( 1 2 - 4 - 6 - 8 - ) ( - 2 3 - - - 7 8 9 ) ( 1 2 3 - - 6 - - 9 ) ( 1 - - 4 5 - 7 8 - ) ( - 2 3 - 5 - - 8 9 )</pre>	3 Pkt.
<p>Erzeugen Sie 1 Objekt von <b>Lottery</b> und rufen für dieses Objekt die <b>drawing ()</b> - Methode auf. Erzeugen Sie in einer Schleife so lange ein weiteres Objekt von <b>Lottery</b> und rufen dafür die <b>drawing ()</b> -Methode auf, bis die beiden Objekte die gleichen gezogenen Zahlen enthalten, d.h. laut <b>equal5 ()</b> -Methode gleich sind. Geben Sie dann beide Objekte mithilfe der <b>print ()</b> -Methode aus.</p> <p>Es entsteht folgende Ausgabe (zufällige Beispielwerte):</p> <pre>( - - - - 5 6 7 8 9 ) ( - - - - 5 6 7 8 9 )</pre>	3 Pkt.

**Teil 2 (Klasse LotteryYear)****22 Punkte**

<p>Schreiben Sie eine Klasse <b>LotteryYear</b>.</p> <p>Objektvariable (z.B. <b>lotteryYear</b>) in dieser Klasse ist ein Array, das Elemente von <b>Lottery</b> aufnimmt. Die Variable ist nur innerhalb der Klasse sichtbar.</p> <p><i>Das Array speichert alle 52 Lotto-Ziehungen eines Jahres.</i></p>	1 Pkt.
--	--------

Schreiben Sie einen parameterlosen Konstruktor <b>LotteryYear()</b> .	3 Pkt.									
Innerhalb des Konstruktors werden: <ul style="list-style-type: none"><li>das Array erzeugt, auf das die Objektvariable referenziert. Das Array hat die Länge 52,</li><li>sowie 52 <b>Lottery</b>-Objekte, für die jeweils <b>drawing()</b> und <b>sort()</b> aufgerufen wird. Mit diesen Objekten wird das Array befüllt.</li></ul>										
Schreiben Sie eine Objektmethode <b>nrOfNumber(int number)</b> . Diese Methode gibt ein <b>int</b> zurück. Diese Methode ermittelt, wie oft die Zahl <b>number</b> im Jahr gezogen wurde, d.h. wie oft <b>number</b> im <b>lotteryYear</b> -Array vorkommt. Kommt <b>number</b> gar nicht vor (nicht aus dem Bereich 1..9), dann wird <b>-1</b> zurückgegeben.	3 Pkt.									
Schreiben Sie eine Objektmethode <b>frequency()</b> . Diese Methode gibt ein <b>int</b> -Array der Länge 9 zurück. In diesem Array wird für jede der Zahlen 1 bis 9 gespeichert, wie oft sie im <b>lotteryArray</b> vorkommt, d.h. wie oft sie im Jahr gezogen wurde. Verwenden Sie für jede der Zahlen 1 bis 9 die <b>nrOfNumber</b> -Methode.  <i>Frequency steht für die Häufigkeit des Vorkommens jeder einzelnen Zahl 1 bis 9 im Jahr.</i>  Beispiel-Belegung des zu erzeugenden Arrays (Werte sind zufällig erzeugt): <table><tr><td>28</td><td>34</td><td>27</td><td>26</td><td>35</td><td>24</td><td>31</td><td>27</td><td>28</td></tr></table> <div><div>die 1 wurde 28x gezogen</div><div>die 2 wurde 34x gezogen</div><div>die 3 wurde 27x gezogen</div><div>die 4 wurde 26x gezogen</div><div>die 5 wurde 35x gezogen</div><div>die 6 wurde 24x gezogen</div><div>die 7 wurde 31x gezogen</div><div>die 8 wurde 27x gezogen</div><div>die 9 wurde 28x gezogen</div></div>	28	34	27	26	35	24	31	27	28	3 Pkt.
28	34	27	26	35	24	31	27	28		
Schreiben Sie eine Objektmethode <b>printFrequencyString()</b> . Erzeugen Sie innerhalb dieser Methode mithilfe der <b>frequency()</b> -Methode das entsprechende Array und geben es in der folgenden Form aus (Beispielwerte zufällig erzeugt):  <b>[ 28, 34, 27, 26, 35, 24, 31, 27, 28 ] --&gt; 260</b>  Es werden also alle Einträge des Arrays durch Komma getrennt ausgegeben und am Ende (nach „-->“ ) noch die Summe aller Einträge (260).	2 Pkt.									
Schreiben Sie eine Objektmethode <b>printFrequencyDiagram()</b> . Erzeugen Sie innerhalb dieser Methode mithilfe der <b>frequency()</b> -Methode das entsprechende Array und geben es in der folgenden Form aus (Beispielwerte zufällig erzeugt):  1 : ***** 2 : ***** 3 : ***** 4 : ***** 5 : ***** 6 : *****	3 Pkt.									

<pre> 7 : ***** 8 : ***** 9 : ***** </pre> <p>Es werden also für jede Zahl so viele Sterne ausgegeben, wie ihr entsprechender Eintrag im Array.</p>	
<p>Schreiben Sie eine Objektmethode <b>contains5equals()</b>. Diese gibt ein <b>true</b> zurück, wenn im <b>lotteryArray</b> zwei <b>Lottery</b>-Objekte die gleichen Zahlen gezogen haben, d.h. laut <b>equal5()</b>-Methode gleich sind. <b>false</b> sonst.</p>	4 Pkt.
<p>Schreiben Sie eine Objektmethode <b>getArrayOfDoublets()</b>. Diese Methode gibt ein <b>Lottery</b>-Array der Länge 2 zurück. Dieses Array enthält 2 <b>Lottery</b>-Objekte aus dem <b>lotteryArray</b>, die die gleichen gezogenen Zahlen beinhalten, also laut <b>equal5()</b>-Methode gleich sind.</p> <p>Sollten solche <b>Lottery</b>-Objekte nicht im <b>lotteryArray</b> existieren, kann das zurückgegebene Array leer bleiben, also keine <b>Lottery</b>-Objekte enthalten.</p>	3 Pkt.

## Testklasse

4 Punkte

<p>Erzeugen Sie in der <b>main()</b>-Methode ein Objekt der Klasse <b>LotteryYear</b>. Rufen Sie dafür die <b>printFrequencyString()</b>- und <b>printFrequencyDiagram()</b>-Methoden auf.</p> <p>Es entsteht folgende Ausgabe (zufällige Beispielwerte):</p> <pre> [ 27, 27, 30, 33, 34, 29, 25, 24, 31 ] --&gt; 260 </pre> <pre> 1 : ***** 2 : ***** 3 : ***** 4 : ***** 5 : ***** 6 : ***** 7 : ***** 8 : ***** 9 : ***** </pre>	2 Pkt.
<p>Prüfen Sie, ob das erzeugte Objekt <b>Lottery</b>-Dopplungen enthält (<b>contains5equals()</b>). Wenn ja, dann sollen diese ermittelt (<b>getArrayOfDoublets()</b>) und ausgegeben (<b>print()</b>) werden.</p> <p>Es entsteht folgende Ausgabe (zufällige Beispielwerte):</p> <pre> ( 1 - - 4 5 - 7 - 9 ) ( 1 - - 4 5 - 7 - 9 ) </pre>	2 Pkt.

Zur Kontrolle : Ungefähre Ausgabe auf der Konsole (alles zufällige Beispielwerte):

----- Teil 1 -----

```
( - 2 3 - 5 - - 8 9 )
( 1 2 - 4 5 - - - 9 )
( 1 - - - - 6 7 8 9 )
( 1 2 - 4 - - 7 8 - )
( - 2 - 4 - - 7 8 9 )
( 1 - 3 - 5 6 - - 9 )
( 1 2 3 4 - - 7 - - )
( 1 2 3 - - 6 7 - - )
( - - - - 5 6 7 8 9 )
( - 2 - 4 - - 7 8 9 )

( - - - - 5 6 7 8 9 )
( - - - - 5 6 7 8 9 )
```

----- Teil 2 -----

[ 27, 27, 30, 33, 34, 29, 25, 24, 31 ] --> 260

```
1 : *****
2 : *****
3 : *****
4 : *****
5 : *****
6 : *****
7 : *****
8 : *****
9 : *****
```

```
( 1 - - 4 5 - 7 - 9 )
( 1 - - 4 5 - 7 - 9 )
```

Klassendiagramme:

Lottery	
- drawingResults	: int[]
+ Lottery()	
+ contains(int)	: boolean
+ drawing()	
+ sort()	
+ toString()	: String
+ print()	
+ equals(Lottery)	: boolean

Testklasse	
+ <u>main(String[])</u>	

- private  
+ public  
\_ static

LotteryYear	
- lotteryYear	: Lottery[]
+ LotteryYear()	
+ nrOfNumber(int)	: int
+ frequency()	: int[]
+ printFrequencyString()	
+ printFrequencyDiagram()	
+ contains5equals()	: boolean
+ getArrayOfDoublets()	: Lottery[]