

KLAUSUR**PROGRAMMIERUNG 1****29. JANUAR 2019****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 61 Punkte zu erzielen (Teil 1: 14 Pkt., Testklasse: 7 Pkt., Teil 2: 28 Pkt., Testklasse: 9 Pkt., fehlerfreies Programm: 3 Pkt.).

Notenspiegel:

Note	1,0	1,3	1,7	2,0	2,3	2,7	3,0	3,3	3,7	4,0	5,0
Punkte	61,0 58,0	57,5 - 55,0	54,5 - 52,0	51,5 - 49,0	48,5 - 45,5	45,0 - 42,5	42,0 - 39,5	39,0 - 36,5	36,0 - 33,5	33,0 - 30,5	<30,5

Vorbereitung:

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

Teil 1 (Klasse PosZahl)**14 Punkte**

Schreiben Sie eine Klasse PosZahl . <i>PosZahl steht für Positive Zahl, also ganze Zahlen größer oder gleich 0.</i>	1 Pkt.
Objektvariable ist <ul style="list-style-type: none"> • wert vom Typ int Die Objektvariable ist nur innerhalb der Klasse sichtbar!	2 Pkt.
Schreiben Sie für die Klasse PosZahl einen parametrisierten Konstruktor PosZahl(int wert) . Der Wert des Parameters wird verwendet, um der Objektvariablen den Wert zuzuweisen. Achtung: Ist der Parameterwert kleiner als 0, wird der Objektvariablen der negative Wert von wert zugewiesen, d.h. der zugewiesene Wert muss positiv sein!	2 Pkt.
Schreiben Sie eine Objektmethode summe(PosZahl z) . Diese Methode gibt ein Objekt von PosZahl zurück! Der Wert des zurückgegebenen Objektes ergibt sich aus der Summe der Werte des aufrufenden Objektes und des Wertes von z .	2 Pkt.
Schreiben Sie eine Objektmethode differenz(PosZahl z) . Diese Methode gibt ein Objekt von PosZahl zurück! Der Wert des zurückgegebenen Objektes ergibt sich aus der Differenz der Werte des aufrufenden Objektes und des Wertes von z .	1 Pkt.
Schreiben Sie eine Objektmethode istKleiner(PosZahl z) , die ein true zurückgibt, wenn der Wert des aufrufenden Objektes kleiner ist als der Wert von z . false sonst.	2 Pkt.
Schreiben Sie eine Objektmethode istGleich(PosZahl z) , die ein true zurückgibt, wenn der Wert des aufrufenden Objektes gleich dem Wert von z ist. false sonst.	1 Pkt.
(Über-)schreiben Sie die Methode toString() , so dass der Wert des aufrufenden Objektes in eckigen Klammern als String zurückgegeben wird, z.B. [4]	2 Pkt.
Schreiben Sie eine Methode print() , die mit Hilfe von toString() eine Ausgabe auf der Konsole erzeugt.	1 Pkt.

Testklasse**7 Punkte**

<p>Schreiben Sie eine Klasse Testklasse mit main()-Methode. Erzeugen Sie in der main()-Methode drei Objekte der Klasse PosZahl:</p> <ul style="list-style-type: none">• verwenden Sie als Parameter: 8, -17, -8 <p>Wenden Sie jeweils die Methoden System.out.print() und print() aus PosZahl so an, dass folgende Ausgabe entsteht:</p> <pre>z1 = [8] z2 = [17] z3 = [8]</pre>	3 Pkt.
<p>Wenden Sie jeweils die Methoden System.out.println() sowie istKleiner(), istGleich(), summe(), differenz() (und evtl. toString()) aus PosZahl so an, dass folgende Ausgabe entsteht:</p> <pre>z1 kleiner z2 ? : true z1 kleiner z3 ? : false z1 gleich z2 ? : false z1 gleich z3 ? : true z1 + z2 = [25] z1 - z2 = [9]</pre>	4 Pkt.

Teil 2 (Klasse **PosZahlArray**)

28 Punkte

Schreiben Sie eine Klasse PosZahlArray . Objektvariable (z.B. pza) in dieser Klasse ist ein Array, das Elemente von PosZahl aufnimmt. Die Variable ist nur innerhalb der Klasse sichtbar.	1 Pkt.
Schreiben Sie einen parametrisierten Konstruktor PosZahlArray(int laenge) . Innerhalb des Konstruktors werden: <ul style="list-style-type: none"> das Array erzeugt, auf das die Objektvariable referenziert. Das Array hat die Länge laenge (Parameterwert), ein Objekt der Klasse Random erzeugt (dafür muss java.util.Random importiert werden), das Array mit Objekten der Klasse PosZahl befüllt. Die Werte dieser Objekte werden zufällig (unter Verwendung des Random-Objektes) erzeugt. Erzeugen Sie zufällig Werte aus dem Bereich [-20 bis +20] (41 verschiedene Zufallszahlen möglich – also -20 und +20 inklusive) 	5 Pkt.
(Über-)schreiben Sie die Methode toString() , so dass eine Zeichenkette der Form (Beispielwerte) : [[9], [15], [1], [9], [2], [2], [0], [4], [13], [18]] zurückgegeben wird. Das heißt, es wird das komplette Array pza ausgelesen (hier Länge 10), die einzelnen Elemente sind durch Komma getrennt.	3 Pkt.
Schreiben Sie eine Objektmethode print() , die den mithilfe von toString() erzeugten String auf die Konsole ausgibt.	1 Pkt.
Schreiben Sie eine Objektmethode kleinste() . Diese Methode gibt das kleinste PosZahl -Objekt aus dem Array pza zurück.	4 Pkt.
Schreiben Sie eine Objektmethode summe() . Diese Methode gibt ein PosZahl -Objekt zurück, dessen Wert der Summe aller Werte aus dem Array pza entspricht.	3 Pkt.
Schreiben Sie eine Objektmethode elementeGroesserAls(PosZahl schranke) . Diese Methode gibt ein Array vom Typ PosZahl[] zurück. Dieses Array enthält alle Objekte aus pza , deren Wert größer ist als der Wert von schranke .	5 Pkt.
Schreiben Sie eine statische Methode printPosZahlArray(PosZahl[] pa) . Diese Methode gibt das als Parameter übergebene Array auf die Konsole aus, z.B. (Beispielwerte): [[15], [13], [18], [16], [17], [20], [17]]	3 Pkt.

Schreiben Sie eine Objektmethode sortieren() . Diese Methode sortiert das Array pza aufsteigend (kleinster Wert zuerst).	3 Pkt.
--	--------

Testklasse

9 Punkte

<p>Erzeugen Sie in der main() –Methode ein Objekt der Klasse PosZahlArray. Das Array soll die Länge 20 haben. Rufen Sie die print() –Methode auf. Es entsteht folgende Ausgabe (Beispielwerte – können anders sein):</p> <pre>[[6], [5], [7], [10], [13], [18], [20], [12], [7], [7], [6], [3], [17], [3], [16], [7], [7], [10], [12], [13]]</pre>	2 Pkt.
<p>Wenden Sie jeweils die Methoden System.out.println() sowie kleinste() und summe (und evtl. toString()) so an, dass folgende Ausgabe entsteht (Beispielwerte – können anders sein):</p> <pre>kleinste PosZahl : [3] Summe : [199]</pre>	3 Pkt.
<p>Erzeugen Sie ein neues PosZahl-Objekt namens schranke mit dem Wert 10. Wenden Sie die Methoden System.out.println() und print() für schranke so an, dass folgende Ausgabe entsteht:</p> <pre>schranke = [10]</pre> <p>Wenden Sie die Objektmethode elementeGroesserAls(schranke) an (übergeben Sie schranke als Parameter). Rufen Sie die statische Methode printPosZahlArray() so auf, dass das von elementeGroesserAls(schranke) zurückgegebene Array ausgegeben wird, z.B. (Beispielwerte – können anders sein):</p> <pre>[[13], [18], [20], [12], [17], [16], [12], [13]]</pre>	3 Pkt.
<p>Rufen Sie die Objektmethoden sortieren() und print() auf, so dass folgende Ausgabe entsteht (Beispielwerte – können anders sein):</p> <pre>[[3], [3], [5], [6], [6], [7], [7], [7], [7], [7], [10], [10], [12], [12], [13], [13], [16], [17], [18], [20]]</pre>	1 Pkt.

Zur Kontrolle : Ungefähre Ausgabe auf der Konsole (in Teil 2 Beispielwerte):

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

```

z1 = [8]
z2 = [17]
z3 = [8]
z1 kleiner z2 ? : true
z1 kleiner z3 ? : false
z1 gleich z2 ? : false
z1 gleich z3 ? : true
z1 + z2 = [25]
z1 - z2 = [9]

```

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

```

[ [6], [5], [7], [10], [13], [18], [20], [12], [7], [7], [6], [3],
[17], [3], [16], [7], [7], [10], [12], [13] ]
kleinste PosZahl : [3]
Summe : [199]
schranke = [10]
[ [13], [18], [20], [12], [17], [16], [12], [13] ]
[ [3], [3], [5], [6], [6], [7], [7], [7], [7], [7], [10], [10],
[12], [12], [13], [13], [16], [17], [18], [20] ]

```

Klassendiagramme:

PosZahl	
- wert	: int
+ PosZahl(int)	
+ summe(PosZahl)	: PosZahl
+ differenz(PosZahl)	: PosZahl
+ istKleiner(PosZahl)	: boolean
+ istGleich(PosZahl)	: boolean
+ toString()	: String
+ print()	

- private
+ public
_ static

PosZahlArray	
- pza	: PosZahl[]
+ PosZahlArray(int)	
+ toString()	: String
+ print()	
+ kleinste()	: PosZahl
+ summe()	: PosZahl
+ elementeGroesserAls(PosZahl)	: PosZahl[]
+ <u>printPosZahlArray(PosZahl[])</u>	
+ sortieren()	

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