

Leistungsnachweis Java

Name: _____ Klasse: _____

Erreichte Punktzahl: _____ / _____ Note: _____

Hilfsmittel: Alle mitgebrachten ausgedruckten oder geschriebenen Aufzeichnungen, TR!

Aufgabennr/text	Erreicht	Maximal
(1.) Definiert wurde ein 4 x 5 Array u vom Typ double!		
(a.) Schreiben Sie eine Anweisung, welche das Array OHNE Inhalte deklariert und erzeugt!		2
(b.) Schreiben Sie die Namen aller Elemente in der zweiten Spalte von u(mit Komma getrennt!) auf!		2
(c.) Schreiben Sie die Namen aller Elemente in der dritte Zeile von u(mit Komma getrennt!) auf!		2
(d.) Schreiben Sie ein Codefragment mit einer for Wiederholungsanweisung, welche alle Elemente der ersten Spalte ausgibt! Dabei ist die einzelne Ausgabe nicht erlaubt!		4
(e.) Schreiben Sie ein Codefragment mit einer for Wiederholungsanweisung, welche die Summe der Elemente der 2. Zeile ausgibt!		6
(2.) Schreiben Sie eine Methode ReverseArray, welche die Reihenfolge aller Elemente eines eindimensionalen Arrays mit Werten vom Datentyp Integer umkehrt UND den alten Inhalt des Arrays erhält! Die Methode hat zwei Parameter: ArrAlt, ArrNeu und wird mit ReverseArray(myData,myReverseData) aufgerufen! Beispiel: {1,2,3,4} wird zu {4,3,2,1}!		8

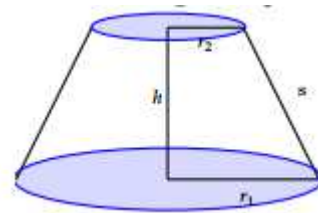
<p>(3.) Sortieren Sie das folgende Array aufsteigend durch Anwendung des Selectionsort-Algorithmus! Geben Sie dazu das Array nach jeder Vertauschungs-Operation(NICHT Durchlauf) an!</p> <table border="1" data-bbox="165 383 643 602"> <tr> <td>7</td> <td>3</td> <td>6</td> <td>2</td> <td>1</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	7	3	6	2	1																	9
7	3	6	2	1																		
<p>(4.) Sortieren Sie das folgende Array aufsteigend durch Anwendung des Insertionsort-Algorithmus. Geben Sie dazu das Array nach jeder Einfügeoperation an!</p> <table border="1" data-bbox="165 763 643 1028"> <tr> <td>7</td> <td>3</td> <td>6</td> <td>2</td> <td>1</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	7	3	6	2	1																	9
7	3	6	2	1																		
<p>(5.) Gegeben ist ein Array mit n Elementen! Wie viele Vertauschungen werden beim Bubblesort Algorithmus maximal und minimal pro Durchlauf benötigt? Maximal: _____ Minimal: _____</p>		4																				
<p>(6.) Bestimmen Sie die Ausgabe des zugehörigen Struktogrammes! Erstellen Sie dazu eine Tabelle, welches die Werte der Variablen in der Wiederholungsanweisung enthält! Sollten Sie eine unendliche Wiederholungsanweisung erhalten, brechen Sie nach der 8. Wiederholung ab!</p> <table border="1" data-bbox="172 1473 874 1951"> <tr> <td>grenze=12</td> </tr> <tr> <td>z=0</td> </tr> <tr> <td>x=1</td> </tr> <tr> <td>y=1</td> </tr> <tr> <td>Ausgabe x,y,z</td> </tr> <tr> <td>z=x+y</td> </tr> <tr> <td>Ausgabe x,y,z</td> </tr> <tr> <td>x=y</td> </tr> <tr> <td>y=z</td> </tr> <tr> <td>y<grenze</td> </tr> </table>	grenze=12	z=0	x=1	y=1	Ausgabe x,y,z	z=x+y	Ausgabe x,y,z	x=y	y=z	y<grenze		6										
grenze=12																						
z=0																						
x=1																						
y=1																						
Ausgabe x,y,z																						
z=x+y																						
Ausgabe x,y,z																						
x=y																						
y=z																						
y<grenze																						

Durchlauf	x	y	z
0.			
1.			
2.			
3.			
4.			
5.			
6.			
7.			
8.			

(7.) Erstellen Sie eine Klasse *CCirculaCone*, die das Volumen ($V = 1/3 * h * \pi * r^2$) eines Kegels in einer Methode *CalcVolume* berechnet. Schreiben Sie dazu die notwendigen Methoden, um die Höhe und den Radius in der Klasse als private Variablen mit sinnvollen Namen zu verkapseln. Erstellen Sie einen Konstruktor, der die notwendigen Variablen einliest und einen, der als Vorgabewerte alle Variablen auf Null setzt.

Erstellen Sie ein Hauptprogramm, welches das Volumen von einem Kegelstumpf (siehe Abbildung rechts mit Höhe 4 cm, Radius $r_2 = 10\text{cm}$ und $r_1 = 15\text{cm}$ berechnet)! Verwenden Sie zu dieser Berechnung zwei Instanzen der Klasse *CCirculaCone*!

Hinweis: Sollten Ihnen Werte für die Berechnung fehlen, empfiehlt es sich, eine Skizze des Schnittes mit den Werten zu machen!



20



Leistungsnachweis Java

--	--	--

Viel Erfolg!

Leistungsüberprüfung Java – Lösungen

Aufgabenr/text	Erreicht	Maximal																									
(1.) Definiert wurde ein 4 x 5 Array u vom Typ double!																											
(a.) <code>double u[][] = double int[4][5];</code>		2																									
(b.) <code>u[0][1], u[1][1], u[2][1], u[3][1]</code>		2																									
(c.) <code>u[2][0], u[2][1], u[2][2], u[2][3], u[2][4]</code>		2																									
(d.) <code>for (int i=0;i<u.length;i++) System.out.print(u[i][0]+" ");</code>		4																									
(e.) <code>int sum=0; for (int i=0;i<u[0].length;i++) sum=sum+u[1][i]; System.out.print(sum);</code>		6																									
(2.) <code>static void ReverseArray(int ArrAlt[],int ArrNeu[]) { for (int i=0;i<ArrAlt.length;i++) ArrNeu[i]=ArrAlt[ArrAlt.length-i-1]; }</code>		8																									
(3.) <table border="1" data-bbox="188 1411 667 1630"> <tbody> <tr><td>7</td><td>3</td><td>6</td><td>2</td><td>1</td></tr> <tr><td>1</td><td>3</td><td>6</td><td>2</td><td>7</td></tr> <tr><td>1</td><td>2</td><td>6</td><td>3</td><td>7</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>6</td><td>7</td></tr> </tbody> </table>	7	3	6	2	1	1	3	6	2	7	1	2	6	3	7	1	2	3	6	7		8					
7	3	6	2	1																							
1	3	6	2	7																							
1	2	6	3	7																							
1	2	3	6	7																							
(4.) <table border="1" data-bbox="188 1742 667 2011"> <tbody> <tr><td>7</td><td>3</td><td>6</td><td>2</td><td>1</td></tr> <tr><td>1</td><td>7</td><td>3</td><td>6</td><td>2</td></tr> <tr><td>1</td><td>2</td><td>7</td><td>3</td><td>6</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>7</td><td>6</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>6</td><td>7</td></tr> </tbody> </table>	7	3	6	2	1	1	7	3	6	2	1	2	7	3	6	1	2	3	7	6	1	2	3	6	7		12
7	3	6	2	1																							
1	7	3	6	2																							
1	2	7	3	6																							
1	2	3	7	6																							
1	2	3	6	7																							

<p>(5.) Maximal: $n-1$ Minimal: 0</p>		4																												
<p>(6.)</p> <table border="1" data-bbox="244 360 600 604"> <thead> <tr> <th>Durchlauf</th> <th>x</th> <th>y</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0.</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1.</td> <td>1</td> <td>1</td> <td>2</td> </tr> <tr> <td>2.</td> <td>1</td> <td>2</td> <td>3</td> </tr> <tr> <td>3</td> <td>2</td> <td>3</td> <td>5</td> </tr> <tr> <td>4.</td> <td>3</td> <td>5</td> <td>8</td> </tr> <tr> <td>5.</td> <td>5</td> <td>8</td> <td>13</td> </tr> </tbody> </table>	Durchlauf	x	y	z	0.	1	1	0	1.	1	1	2	2.	1	2	3	3	2	3	5	4.	3	5	8	5.	5	8	13		8
Durchlauf	x	y	z																											
0.	1	1	0																											
1.	1	1	2																											
2.	1	2	3																											
3	2	3	5																											
4.	3	5	8																											
5.	5	8	13																											
<p>(7.)</p> <pre> public class CCirculaCone{ private double radius; private double height; public CCirculaCone(double radius, double height) { this.radius = radius; this.height = height; } public CCirculaCone() { this.radius = 0; this.height = 0; } public double getHeight() { return height; } public void setHeight(double height) { this.height=height; } public double getRadius() { return radius; } public void setRadius(double radius) { this.radius=radius; } public double CalcVolume() { return 3.1415*height*radius*radius/3; } } public class TestCCirculaCone { public static void main(String[] args) { CCirculaCone myCirculaConeBig = new CCirculaCone(15,12); CCirculaCone myCirculaConeLittle = new CCirculaCone(10,8); double VolumeCirculaCone=myCirculaConeBig.CalcVolume()-myCirculaConeLittle.CalcVolume(); System.out.println("Volumen des Kegelstumpfes:" + VolumeCirculaCone); } } </pre>	<p>2</p> <p>2</p> <p>2</p> <p>2</p> <p>2</p> <p>2</p> <p>2</p> <p>4</p> <p>4</p>	8																												

