

Algorithmen und Datenstrukturen im WS 2005/2006

Prof. Dr. W. Kowalk,
Abteilung Rechnernetze,
Department Informatik, Fakultät II, Universität Oldenburg

10.1	Große Zahlen in Java (50 Punkte)
	<p>1. Schreiben Sie ein Programm in Java, welches die Zahl Pi auf beliebig viele Stellen genau berechnet.</p> <pre> BigDecimal PI(int scale) { // arcsin(x) = x + 1/2 * x^3/3 + 1*3/(2*4) * x^5/5 + // 1*3*5/(2*4*6) * x^7/7 + ... // arcsin(0.5) = Pi/6 // 3.141592653589793238462643383279502884197169399375 // 1058209749445923078164062862089986280348253421170680 </pre> <p>b) Die Leibnizsche Formel für Pi lautet:</p> $\frac{\pi}{4} = \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots \pm \frac{1}{2n-1}$ <p>Warum ist diese Formel für die obige Reihenentwicklung nicht geeignet?</p>
10.2	3AA Programmierung (50 Punkte)
	<p>Schreiben Sie ein 3AA-Programme, welches eine im Speicher stehende Folge von Zahlen sortiert. Verwenden Sie das Verfahren: Sortieren durch Einfügen.</p> <p>Um die Richtigkeit des Programms abstrakt zu zeigen, argumentieren Sie über die Sortierte Menge, welche zunächst nur aus einem Element besteht.</p> <ol style="list-style-type: none"> 1. die Sortierte Menge aus nur einem Element; 2. die äußere Schleife, welche jeweils ein Element der Sortierten Menge hinzufügt; 3. die innere Schleife, welche benachbarte Element der Sortierten Menge vom rechten bis zum linken vertauscht; 4. die Abbruchbedingung, welche die innere Schleife beendet, sobald das eingefügte Element an der richtigen Stelle steht. <p>Argumentieren Sie, warum Ihr Programm</p> <ol style="list-style-type: none"> 1. stabil ist, 2. ordnungsverträglich, 3. terminiert mit $O(N^2)$

Hinweise:

Die Übungen sind bis Mittwoch, 18.1.2006, 10.00 Uhr in den Kästen der jeweiligen Tutoren abzugeben.