Julian Ullrich
Heinrich-Heine-Universität Düsseldorf

Kommentar zur HK W20/21 NKT Programmierung WS 20/21

Bitte bedenkt, dass es sich hier lediglich um einen kommentierten Lösungsvorschlag der Tutoren des Nachklausurtutoriums und nicht um eine Musterlösung handelt. Insbesondere können wir nicht garantieren, dass der Lösungsvorschlag komplett fehlerfrei ist, bzw. dass jede der Antworten die volle Punktzahl erbringen würde. Solltet ihr an einer Stelle einen Fehler finden, oder der Meinung sein eine Antwort wäre z.B. nicht umfangreich genug, dann sagt uns gerne im Rahmen des Tutoriums Bescheid, sodass wir das Dokument anpassen können. Bei Aufgaben mit einem **markierten** Bereich ist dieser besonders wichtig und möglichlicherweise auch ausreichend für die vollen Punkte, wobei wir persönlich um auf nummer sicher zu geben mehr hinschreiben würden, jedoch das fett geschriebene auf keinen Fall fehlen sollte.

(2) a) **int, double, char.**

Weitere möglichkeiten wären: byte, short, long, float, boolean

b) $(a \wedge b) \vee \neg b$ bzw. (a && b) || !b

Da a=true und b = false vorgegeben ist, evaluiert $(a \wedge b)$ zu false, und $\neg b$ zu true, da wir die Ausdrücke mit einem oder verküpft haben, evaluiert auch der gesamte Ausdruck zu true.

c) Die beiden Schlüsselwörter für Schleifen in Java sind **for** und **while**.

d) Die Aufgabe eines Konstruktors ist es, ein Objekt seiner Klasse zu erzeugen und ggf. zu initialisieren.

e) Anmerkung: Zu bedenken ist, dass length anders als bei z.B. einem int-Array hier eine Methode ist, und unsere Indexierung in einem String bei 0 beginnt.

i. **str.length()**

ii. **str.charAt(1)**

f) **Eine Rekursive Methode ist eine Methode, welche sich selbst aufruft.** Dabei wird in der Regel durch ändern der Übergabeparameter das zu lösende Problem bis zum erreichen einer gewissen Abbruchbedingung in kleinere Teilprobleme zerlegt, um die Lösung danach aus deren Teillösungen zusammenzusetzen.

g) Eine ArrayIndexOutOfBoundsException tritt dann auf, wenn versucht wird, mit einem unzulässigen Index auf ein Array zuzugreifen. Das ist der Fall, wenn der Index entweder negativ, gleich, oder größer als die Länge des Arrays ist.

```
int[] x = new int[13];
int a = x[-3]; // Index negativ
int b = x[13]; // Index zu hoch (nur 0-12 erlaubt)
int c = x[1232] // Index zu hoch (nur 0-12 erlaubt)
```

(3) Prumzahlen - Siehe Klasse Prumzahlen.

(4) Schleifen und Rekursion - Siehe Klasse Schleifen

(5) Coderätsel:

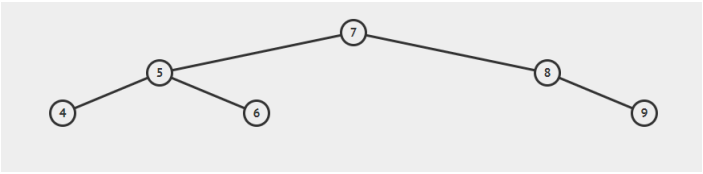
a) Die Ausgabe lautet: 2 3 4

b) Es wird zunächst geprüft, ob genügend Werte (mindestens 4) übergeben wurden. Danach wird ein double Array entsprechender Länge erzeugt, in welchem durch umparsen die Kommandozeilenparameter als double gespeichert werden. Anschließend übergeben wir dieses Array an die Methode mysteryMethod, speichern die von der Methode zurückgegebene Referenz und geben danach die Werte davon mit Leerzeichen getrennt über die Standardausgabe aus.

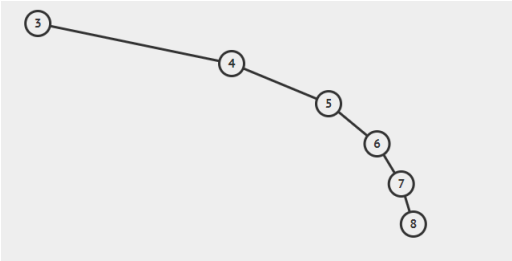
c) **In mysteryMethod werden Mittelwerte für unsere Einträge im übergebenen Array berechnet, und zwar so, dass der Eintrag im Rückgabearray an Index i dem arithmetischen Mittel der 3 Werte im übergebenen Array an den Stellen i, i+1 und i+2 entspricht.** Es wird nur ein Mittelwert berechnet, wenn es noch 2 weitere Elemente gibt, sodass das zurückgegebene Array um 2 kürzer ist.

(6) Binäre Suchbäume

a) Zahlenfolge (7,5,8,4,6,9)



b) Zahlenfolge (3,4,5,6,7,8)



c) Wie gut in einem binären Suchbaum, gesucht werden kann, hängt von der Tiefe des Baumes ab. Da der Suchbaum aus a) ausbalanciert ist, können wir hier schneller suchen als in Suchbaum b), welcher zu einer verketteten Liste entartet ist.

d) siehe Klasse Btree

(7) Siehe Klasse Klammern

(8) Siehe Klassen/Inferfaces im package Post