Objektorientierte Programmierung (OOP)
Lösungsblätter zur Klausur, WS 2019/20

WICHTIG

Bitte nutzen Sie für Ihre Antworten ausschließlich diese ausgehändigten Lösungsblätter!
Sie geben am Ende der Klausur nur die Lösungsblätter ab!

Vor- und Nachname	Matrikelnummer	Note

Prüfungsdauer: 90 Minuten

Wertung: Insgesamt können 100 Punkte erreicht werden. Ab 50 Punkten ist die Klausur „bestanden“. Die pro Aufgabe erzielbaren Punkte sind angegeben.

Hilfsmittel: Es sind keine Hilfsmittel erlaubt.

Hinweise:

- Bitte verwenden Sie keinen Rotstift oder Bleistift.
- Tragen Sie in das obige Feld Ihren Namen und Ihre Matrikelnummer ein. Unterschreiben Sie das Deckblatt unten.
- Sie erhalten je einen getrennten Satz an Aufgaben- und Lösungsblättern.
- Schreiben Sie Ihre Antworten bitte ausschließlich auf die Lösungsblätter und dort in die dafür vorgesehenen Abschnitte.
- Geben Sie dieses Deckblatt zusammen mit allen Lösungsblättern ab. Die Aufgabenblätter sind für den Verbleib bei Ihnen bestimmt.
- Überprüfen Sie die Klausur auf Vollständigkeit. Bitte beanstanden Sie Mängelexemplare umgehend.
- Es werden nur leserliche Klausurlösungen bewertet!
- Programmcode hat sich in Art und Form an den in der Veranstaltung verwendeten Konventionen zu orientieren. Schreiben Sie syntaktisch korrekten Code.

Datum	Unterschrift
6. Februar 2020	

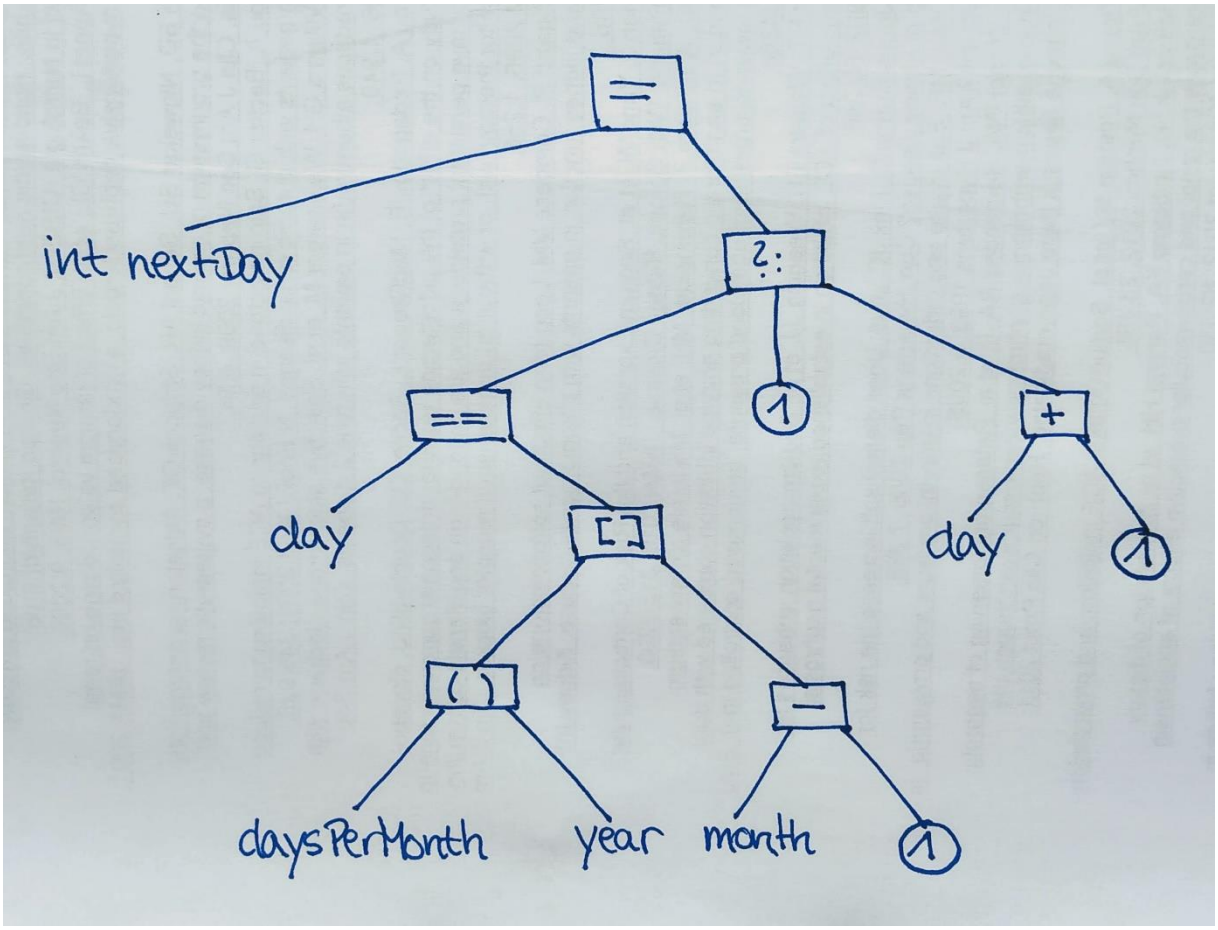
1. Fragen (16 x 2 = 32 Punkte)

Tragen Sie hier bitte durch deutliches Ankreuzen Ihre Lösungen ein. Die Spalte „Punkte“ bitte leer lassen.

	1.	2.	3.	4.	*Punkte*		1.	2.	3.	4.	*Punkte*
A	☐	☑	☑	☐		I	☑	☑	☐	☑	
B	☐	☑	☐	☐		J	☐	☑	☐	☐	
C	☐	☐	☑	☐		K	☐	☑	☐	☐	
D	☐	☐	☐	☑		L	☐	☐	☑	☐	
E	☑	☐	☐	☑		M	☑	☐	☐	☑	
F	☐	☐	☐	☑		N	☐	☑	☑	☐	
G	☐	☑	☐	☐		O	☑	☐	☑	☐	
H	☑	☑	☐	☑		P	☐	☐	☐	☑	

Sollten Sie eine Lösung revidieren wollen, streichen Sie die betreffende Reihe durch und notieren Sie hier neben der Tabelle Ihre korrigierte Lösung im Stil von Z13 (die hypothetische Frage Z, Antwort 1 und 3).

2. Syntaxbaum (8 Punkte)



3. Vererbung (max. 12 Punkte)

```
class Speaker {
    String text;
    Speaker(String text) {
        assert next != null;
        this.text = text;
    }
    public String toString() {
        return text;
    }
}

class Repeater extends Speaker {
    Repeater(String text) {
        super(text);
    }
    public String toString() {
        return super.toString() + " " + super.toString();
    }
}
```

4. Boolscher Datentyp selbst gemacht (max. 19 Punkte)

4.1 Interface vervollständigen (2 Punkte)

Tragen Sie bitte nur den fehlenden Code aus dem Interface hier ein.

```
    default MyBool nand(MyBool b) { // 1 Punkt
        return and(b).not();        // 1 Punkt
    }
```

4.2 Wahrheitswerte als Aufzählungstyp (7 Punkte)

```
enum MyBool implements Boolable {                    // 1
    TRUE, FALSE;                                     // 1
    public MyBool not() {                            // 1
        return this == TRUE ? FALSE : TRUE ;   // 1
        // return MyBool.values()[(this.ordinal() + 1) % 2];
    }
    public MyBool and(MyBool b) {                    // 1
        if (this == TRUE && b == TRUE) return TRUE; // 1
        return FALSE;                                // 1
    }
}
```

4.3 Klasse Claim (10 Punkte)

```
class Claim {                                  // 1
    MyBool[] bools;                            // 1
    Claim(MyBool... bools) {                   // 1
        this.bools = bools;                    // 1
    }
    MyBool andAll() {                          // 1
        assert bools.length != 0;              // 1
        MyBool b = bools[0];                   // 1
        for (int i = 1; i < bools.length; i++) // 1
            b = b.and(bools[i]);               // 1
        return b;                              // 1
    }
}
```

5. Bereichszähler (max. 29 Punkte + zusätzlich 11 Bonuspunkte)

5.1 Zahlenbereich (10 P.)

```
class Range { // je Codezeile 1 Punkt (somit max. 10 Punkte)
    double from, to;
    Range(double from, double to) {
        if (from > to) throw new IllegalArgumentException("from <= to");
        this.from = from;
        this.to = to;
    }
    static Range of(double from, double to) {
        return new Range(from, to);
    }
    boolean isIn(double d) {
        return d >= from && d < to;
    }
}
```

5.2 Bereichszähler, 1. Teil (13 P.)

```
class RangeCounter {                                  // 1
    Range[] ranges;                                   // 1
    long[] counter;                                   // 1
    RangeCounter(Range[] ranges) {                    // 1
        this.ranges = ranges;                         // 1
        counter = new long[ranges.length];            // 1
    }
    boolean register(double d) {                      // 1
        boolean rangeFound = false;                   // 1
        for(int i = 0; i < ranges.length; i++) {      // 1
            if (ranges[i].isIn(d)) {                  // 1
                counter[i]++;                         // 1
                rangeFound = true;                    // 1
            }
        }
        return rangeFound;                            // 1
    }
}
```

5.3 Bereichszähler, 2. Teil (6 P.)

```
    public String toString() {
        String s = "\n";
        long max = Arrays.stream(counter).max().getAsLong();
        for(long c : counter) {                                        // 2
            s += "*".repeat((int)(40.0 * c / max)) + " " + c + "\n"; // 2
        }
        return s;                                                      // 2
    }
```

5.4 Bereichszähler, 3. Teil (4 + 7 = 11 Bonuspunkte)

```
    void run(long n, DoubleGenerator dg) {
        while(n-- > 0) register(dg.generate());  // 4
    }
```

```
rc.run(1_000_000,                                                  // 1
       new DoubleGenerator() {                                     // 2
           Random r = new Random();                                // 2
           public double generate() { return r.nextDouble(); }     // 2
       });
```

– Diese Seite wird nur als Lösung berücksichtig, wenn Sie im Lösungsteil ausdrücklich darauf hinweisen. –

– Diese Seite wird nur als Lösung berücksichtig, wenn Sie im Lösungsteil ausdrücklich darauf hinweisen. –

– Diese Seite wird nur als Lösung berücksichtig, wenn Sie im Lösungsteil ausdrücklich darauf hinweisen. –