



Hochschule Aalen

*Fakultät Elektronik und Informatik
Studienbereich Informatik*



Objektorientierte Programmierung

Vorlesung im Wintersemester 2023/2024

Prof. Dr. habil. Christian Heinlein

7. Übungsblatt (30. November – 6. Dezember 2023)

Aufgabe 7: Abstrakte Klassen

Implementieren Sie Klassen zur Repräsentation arithmetischer Ausdrücke (abstrakte Klasse `Expr`), die aus Konstanten (konkrete Unterklasse `Const`) und den vier Grundrechenarten (konkrete Unterklassen `Add`, `Sub`, `Mul` und `Div`) bestehen!

Für irgendeinen arithmetischen Ausdruck `x` soll `x.compute()` seinen Wert berechnen; `x.toString()` soll eine Zeichenkettendarstellung des Ausdrucks `x` liefern; `x.equals(y)` soll genau dann `true` liefern, wenn `y` ebenfalls ein arithmetischer Ausdruck ist, der strukturell und inhaltlich mit `x` übereinstimmt.

Definieren Sie zusätzlich eine Hilfsklasse `Bin` mit abstrakten Hilfsmethoden `oper` und `combine`, um Gemeinsamkeiten mehrerer Klassen an einer Stelle zusammenzufassen und so Codeverdopplungen so weit wie möglich zu vermeiden (vgl. UML-Diagramm unten sowie Hinweise in der Übungsstunde)!

Testen Sie Ihre Klassen z. B. mit folgendem Hauptprogramm, das als Ausgabe exakt

```
((2.0*3.0)+4.0) = 10.0
true
false
false
```

liefern muss:

```
// Testprogramm für arithmetische Ausdrücke.
class ExprTest {
    public static void main (String [] args) {
        // Ausdruck 2 * 3 + 4 erzeugen, ausgeben und berechnen.
        Expr x = new Add(new Mul(new Const(2), new Const(3)), new Const(4));
        System.out.println(x + " = " + x.compute());

        // Ausdruck x mit anderen Ausdrücken und Objekten vergleichen.
        Expr y = new Add(new Mul(new Const(2), new Const(3)), new Const(4));
        System.out.println(x.equals(y));

        Expr z = new Add(new Mul(new Const(3), new Const(2)), new Const(4));
        System.out.println(x.equals(z));

        System.out.println(x.equals(x.toString()));
    }
}
```

