

Aufgabe 5

Wir betrachten Terme über die Rechenarten $+$, $-$, \times , $/$, die rekursiv definiert sind:

- Jedes Literal ist ein Term, z.B. „4“.
- Jedes Symbol ist ein Term, z.B. „x“.
- Ist t ein Term, so ist „(t)“ ein (geklammerter) Term.
- Sind t_1 , t_2 Terme, so ist „ t_1 Op t_2 “ ebenso ein Term.

Beispiele für gültige Terme sind also „ $A + 8$ “, „ $Ax \times$ “ oder „ $A + (8 \times x)$ “.

- Welches Design-Pattern eignet sich hier am besten zur Modellierung dieses Sachverhalts?
- Nennen Sie drei wesentliche Vorteile von Design-Pattern im Allgemeinen.
- Modellieren Sie eine Klassenstruktur in UML, die diese rekursive Struktur von Termen abbildet. Sehen Sie mindestens einzelne Klassen für die Addition und Multiplikation vor, sowie weitere Klassen für geklammerte Terme und Literale, welche ganze Zahlen repräsentieren. Gehen Sie bei der Modellierung der Klassenstruktur davon aus, dass eine objektorientierte Programmiersprache wie Java zu benutzen ist.
- Erstellen Sie ein Objektdiagramm, welches den Term $t := 4 + (3 \times 2) + (12 \times y / (8 \times))$ entsprechend Ihres Klassendiagramms repräsentiert.
- Überprüfen Sie, ob das Objektdiagramm für den in Teilaufgabe d) gegebenen Term eindeutig definiert ist. Begründen Sie Ihre Entscheidung.
- Die gegebene Klassenstruktur soll mindestens folgende Operationen unterstützen:
 - das Auswerten von Termen,
 - das Ausgeben in einer leserlichen Form,
 - das Auflisten aller verwendeten Symbole. Welches Design-Pattern ist hierfür am besten geeignet?
- Erweitern Sie Ihre Klassenstruktur um die entsprechenden Methoden, Klassen und Assoziationen, um die in Teilaufgabe f) genannten zusätzlichen Operationen gemäß dem von Ihnen genannten Design Pattern zu unterstützen.