Andreas Podelski
Fachbereich 14 – Informatik
Universität des Saarlandes,
Max-Planck-Institut für Informatik

Programmiersprachen (WS 1996/97)

9. Januar 1997

Übungsblatt 10

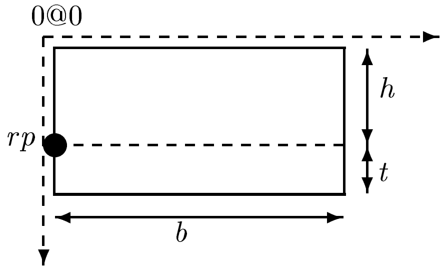
Abdelwaheb Ayari
Hubert Baumeister

Aufgabe 1: (2 Punkte)
Implementiere arithmetische Ausdrücke mit Hilfe der Klassen `Expression`, `Constant`, `Difference`, `Sum`, `Division`, `Product` und `Sqrt`. Instanzen der Klassen, geeignet zusammengesetzt, entsprechen Ausdrücken der Art $1+\sqrt{\frac{4-2}{5*6}}$. Die Objekte der Klassen sollen die Nachricht `value` verstehen, die ihren Wert liefert. Zum Beispiel soll der Ausdruck $\sqrt{1+3}$ durch das Objekt

```
Expression o = new Sqrt(new Sum(new Constant(1),new Constant(3)))
```

dargestellt werden. Wenn `o` die Nachricht `value` geschickt wird, wird 2 zurückgeliefert.

Aufgabe 2: (5 Punkte)
Um die Ausdrücke aus der vorherigen Ausgabe graphisch darstellen zu können, sollen Boxen implementiert werden. Eine `Box` definiert eine rechteckige Fläche mit Breite b, Tiefe t, Höhe h und Referenzpunkt rp. Der Nullpunkt einer `Box` soll die obere linke Ecke sein.

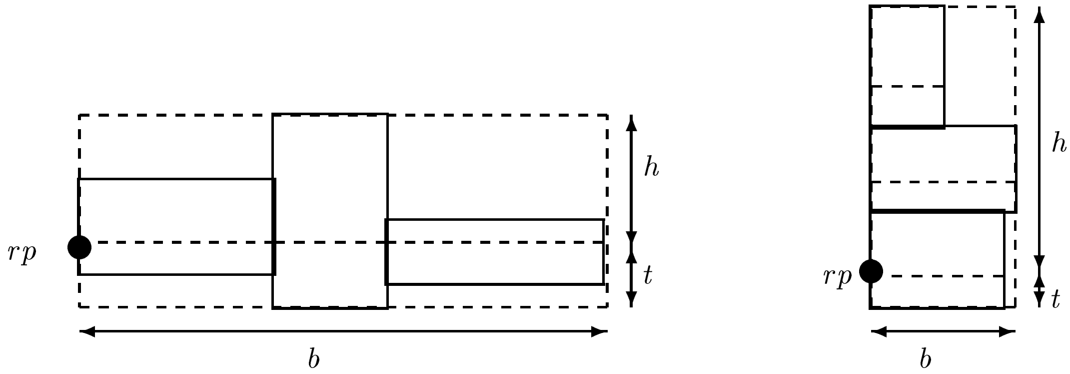


Außerdem sollen Boxen die Nachricht `paint(Graphics g)` und `paint(Graphics g, int x, int y)` verstehen, die eine Box mit Hilfe eines Objekts von Typ `Graphics` zeichnen. Objekte vom Typ `Graphics` (aus dem Paket `java.awt`) können zum Zeichnen verwendet werden. Zum Beispiel kann man Polygone mit der Nachricht

```
drawPolygon(int xPoints[], int yPoints[], int nPoints)
```

zeichnen.
Subtypen von `Box` definieren was in dem von der Box definierten Rechteck dargestellt wird. Eine `TextBox` stellt einen Text dar. Um die Höhe, Breite und Tiefe einer TextBox berechnen zu können braucht man Informationen über den Zeichensatz mit dem der Text in der `paint` Methode dargestellt wird. Diese Informationen stellt ein Objekte der Klasse `FontMetrics` bereit

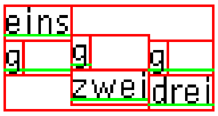
(aus dem Paket `java.awt`). Das `FontMetrics` Objekt für den üblichen Zeichensatz erhält man indem man die Nachricht `getFontMetrics()` an ein Objekt vom Typ `Graphics` schickt.
Eine Instanz der Klasse `HBox` besteht aus einer Liste von `Boxen`. Bei der Darstellung einer `HBox` (mittels `paint(Graphics g)`) werden die einzelnen Boxen nebeneinander dargestellt, so daß sie sich gerade nicht mehr überlappen und ihre Referenzpunkte auf einer horizontalen Linie liegen.



Eine Instanz der Klasse `VBox` stellt eine Liste von Boxen untereinander dar, wobei die Referenzpunkte der Boxen auf einer vertikalen Linie liegen. Der Referenzpunkt der zusammengesetzten Box ist der Referenzpunkt der untersten Box. Es gibt noch zwei weitere Typen von `VBoxes`. Bei dem ersten ist der Referenzpunkt der `VBox` der Referenzpunkt der obersten Box (Klasse `VTop`), beim zweiten ist er die Mitte der zusammengesetzten Box (`VCenter`).
Das folgende Beispiel

```
HBox hb = new HBox();
hb.add(new VBox().add(new TextBox(fm,"eins")).add(new TextBox(fm,"g")));
hb.add((new VCenter()).add(new TextBox(fm,"g")).add(new TextBox(fm,"zwei")));
hb.add((new VTop()).add(new TextBox(fm,"g")).add(new TextBox(fm,"drei")));
hb.paint(g);
```

produziert die Ausgabe, wobei zusätzlich noch die äußere Umrandung der Boxen und die Referenzlinien eingezeichnet sind:



Aufgabe 3: (3 Punkte)
Füge den Klassen für arithmetische Ausdrücke eine Methode `asBox(FontMetrics fm)` hinzu, die als Ergebnis eine `Box` liefert. Es ist sinnvoll noch einige Unterklassen von `Box` für Spezialfälle hinzuzufügen, z.B. für die Darstellung einer Wurzel und der Division.
Probiere die Klassen an folgendem Beispiel aus:

```
Expression e =  new Sum(new Constant(1),
           new Sqrt(new Division(new Difference (new Constant(4), new Constant(2)),
                                 new Product(new Constant(5),new Constant(6))))) ;
```

Die Ausgabe auf dem Bildschirm sollte so aussehen:

$$1 + \sqrt{\frac{4 - 2}{5 * 6}}$$

Tips zur Graphik in Java

Am einfachsten verwendet man Applets um sich die Ergebnisse der Übungsaufgaben auf dem Bildschirm anzusehen. Eine Applet besteht aus einem HTML-File (hier `expression.html`), das auf eine Unterklasse von Applet (hier `ExprApplet`) verweist und der Java-Klasse.
`expression.html`:

```
<HTML>
 <HEAD>
<TITLE>Beispiel: Übungsblatt 10 Aufgabe 3</TITLE>
</HEAD>
 <BODY>
<APPLET CODE="ExprApplet.class" WIDTH=336 HEIGHT=70></APPLET>
</BODY>
 </HTML>
```

`ExprApplet.java`:

```
import expression.*;
import java.applet.Applet;
import java.awt.Graphics;

public class ExprApplet extends Applet {
 public void paint (Graphics g) {
   Expression o =
     new Sum(new Constant(1),
             new Sqrt(new Division(new Difference (new Constant(4), new Constant(2)),
                                   new Product(new Constant(5),new Constant(6))))) ;
   o.asBox(g.getFontMetrics()).paint(g);
 }
}
```

Mit `javac ExprApplet.java` wird das File `ExprApplet.class` erzeugt und mit

`netscape expression.html`

kann man sich das Ergebnis in Netscape anschauen. Die `paint` Methode wird jedesmal ausgeführt, wenn der Browser die Seite oder Teil von der Seite `expression.html` anzeigen muß.
Achtung: der Reload Button in Netscape lädt das Applet nicht neu. Wurde das Applet geändert, so muß Netscape neu gestartet werden.
Eine Anleitung zum Schreiben von Applets und zur Verwendung der Graphik findet man im Java-Tutorial

`file:/usr/local/languages/java/doc/tutorial/index.html`

In der API Dokumentation kann man weitere Hinweise zu den Klassen `Graphics`, `FontMetrics`, `Applet` etc. finden. Die URL ist

`file:/usr/local/languages/java/doc/apibook/index.htm`

Die URLs sind nur gültig auf den Rechnern im Cip-Pool.
Objekte vom Typ `FontMetrics` verstehen die Nachrichten `getAscent()` um die Höhe, `getDescent()` um die Tiefe und `stringWidth(String str)` um die Weite eines Strings in dem aktuellen Zeichensatz zu berechnen.

Beim Zeichnen ist der Ursprung des Koordinatensystems in der oberen linken Ecke. In der Methode `paint(Graphics g, int x, int  y)` verschiebt man am Besten den Koordinatenursprung von `g` mit `g.translate(x,y)`, damit der eigentliche Zeichenalgorithmus für eine Box (definiert in `paint(Graphics g)`) davon ausgehen kann, daß die obere linke Ecke der Box die Koordinaten $(0,0)$ hat.

Abgabe: Da es sinnvoll ist die Aufgaben die Programme in Pakete und die Klassen in einzelne Files zu zerlegen sollten die Lösungen zu dem gesamten Übungsblatt als ein `tar`-File abgegeben werden. Ein `tar`-File wird erzeugt mit

```
tar cvf Susi.Sorglos.ueb10.tar <file1> <file2> <dir1> ...
```