Christian-Albrechts-Universität zu Kiel
AG Programmiersprachen und Übersetzerkonstruktion
Prof. Dr. M. Hanus

Christian-Albrechts-Platz 4
24118 Kiel
Tel.: 0431 / 880-7271

6. Übung zur Vorlesung „Prinzipien von Programmiersprachen“

Wintersemester 2008/2009

Abgabe: 16. Dezember 2008 in der Vorlesung

Aufgabe 20 (Präsenzaufgabe)

In dieser Aufgabe vergleichen wir Unterklassenbildung, wie sie in der Vorlesung definiert wurde (mit Ko- und Kontravarianz) mit Vererbung in Java. Seien die Klassen `A`, `B`, `C` und `D` wie folgt definiert:

```
public class A {
    public C cc( C x ) { System.out.print( "A" ); return null; }
    public C cd( D x ) { System.out.print( "A" ); return null; }
    public D dc( C x ) { System.out.print( "A" ); return null; }
    public D dd( D x ) { System.out.print( "A" ); return null; }

    public static void main( String[] args ) {
        B b = new B();

        b.cc( new C() ); b.cc( new D() );
        b.cd( new C() ); b.cd( new D() );
        b.dc( new C() ); b.dc( new D() );
        b.dd( new C() ); b.dd( new D() );
    }
}

class B extends A {
    public C cc( C x ) { System.out.print( "B" ); return null; }
    public D cd( C x ) { System.out.print( "B" ); return null; }
    public C dc( D x ) { System.out.print( "B" ); return null; }
    public D dd( C x ) { System.out.print( "B" ); return null; }
}

class C {}
class D extends C {}
```

a) Was ist die Ausgabe von `A.main()`?

b) Welche Methoden von `B` sind von `A` (im Sinne der Vorlesung und in Java) geerbt?

c) Wieviele Methoden hat die Klasse `B`? Welche sind überladen?

Aufgabe 21
Definieren Sie eine abstrakte Klasse `SearchTree`. Ein Objekt dieser Klasse soll einen geordneten binären Baum repräsentieren. Die Klasse soll die beiden abstrakten Methoden

```
boolean contains( java.lang.Comparable x )
SearchTree insert( java.lang.Comparable x )
```

besitzen. Konkretisieren Sie den Suchbaum mittels zweier Klassen `Branch` und `Leaf`. Ein Objekt der Klasse `Branch` besitzt drei Attribute: Ein Objekt, welches `java.lang.Comparable` implementiert und die Attribute `left` und `right` vom Typ `SearchTree`. Ein Objekt der Klasse `Leaf` besitzt kein Attribut.

Aufgabe 22
In dieser Aufgabe sollen Sie sich mit dem Hugs-System vertraut machen.

a) Verwenden Sie hierzu zunächst das in der Vorlesung vorgestellte Programm zur Fakultätsberechnung.

   Untersuchen Sie wieviele Schritte zur Berechnung von `fac 10` notwendig sind. Die Ausgabe der Schrittzahl kann in Hugs mit `:set +s` eingeschaltet werden.

b) Implementieren Sie die Fibonacci-Funktion in Haskell.

$$\begin{aligned} &fib(0) = 0 \\ &fib(1) = 1 \\ &fib(n) = fib(n-1) + fib(n-2) \quad \forall n > 1 \end{aligned}$$

   Bestimmen Sie anhand der benötigten Reduktionsschritte die Zeitkomplexität Ihrer Implementierung (liegt in $O(??)$). Die einfachste Methode ist eine graphische Auswertung.

   Können Sie Ihre Funktion entscheidend optimieren?

Wichtige Interpreter-Kommandos des Hugs-Systems:

- `:l` *filename*

  Löschen aller vorhandenen Funktionsdefinitionen (außer Prelude) und Laden des Programms *filename*.

- `:r`

  Wiederholt die letzte Ladeoperation, somit werden alle Funktionsdefinitionen aktualisiert.

- `:q`

  Verlassen des Hugs-Systems.

- `:?`

  Kommandoliste wird angezeigt.

- *Ausdruck* (z.B. `fac 3`)

  Start einer Funktionsberechnung.