

**Übung zu Algorithmen und Programmieren III, WS 2001/2**

## Übung 6

Ausgabe: 20.11.01

Abgabe: 29.11.01 bis 14.00 Uhr

**Aufgabe 1 ( 6 P)**

Für die Flugbuchung braucht man einen abstrakten Datentyp `flug`. Der ADT `flug` soll spezifiziert werden, indem in Haskell ein entsprechender Typ `flug` mit entsprechenden Operationen wie `buchen`, `stornieren` u.ä. angegeben wird:

```

...      } Typen

typ flug = ... } Modell

...      } Funktionsdefinitionen für
           die Semantik der
           Operationen

```

Berücksichtigen Sie die partiellen Funktionen. Machen Sie die Partialität der Funktionen explizit, indem Sie für einen undefinierten Funktionswert eine geeignete Fehlermeldung vorsehen.

**Aufgabe 2 ( 6 P)**

Überlegen Sie sich eine Repräsentation für den ADT `flug` und schreiben Sie die entsprechende Java-Klasse und eine zugehörige Testklasse.

**Aufgabe 3 ( 12 P)**

Es soll ein ADT `NMenge` für die natürlichen Zahlen  $N = \{0, 1, \dots, n-1\}$  entwickelt werden.

1. Wählen Sie ein Modell und spezifizieren Sie einen Satz von Operationen (mindestens: ein Element hinzufügen, entfernen, Vereinigung, Durchschnitt, Differenz, Test auf Enthaltensein). Abstrakte Invariante, Voraussetzungen, Effekte nicht vergessen.
2. Implementieren Sie den ADT `NMenge` mit Hilfe der Klasse `BitSet` (siehe Java-Dokumentation). Zur Implementierung gehören Abstraktionsfunktion, Repräsentationsinvariante, die konkreten Voraussetzungen und Effekte und eine Testklasse mit einer Ausgabefunktion.

**Aufgabe 4 ( 4 Punkte)**

1. Definieren Sie: Vererbung, Multiple Vererbung, Superklasse, Subklasse, Typ, Typumwandlung (type cast).
2. Wie sinnvoll ist ein "type cast" von einer Subklassen-Referenz auf eine Superklassenreferenz?
3. Warum ist ein "type cast" von einer Superklassen-Referenz auf eine Subklasse potentiell gefährlich?
4. Die Verwendung von `protected` wird von manchen abgelehnt, weil dies das "Information hiding"-Prinzip verletzen kann. Diskutieren Sie diese Haltung.