

Übungsblatt 8 - Abgabe vor der Vorlesung 13.12.05

Aufgabe 0 (Wiederholung) [optional]

- Spezifikation
- S. von Prozeduren (Eingabe, Zustand, Ausgabe, Bild-, Urbildbereich, Formale Spezifikation, Voraussetzung und Effekt)
- Modellierende Spezifikation (Syntax + Semantik (Modell, Invariante, Operationsspezifikation)
- Funktionale Spezifikation vs. Prädikative Spezifikation
- Korrektheit einer Spezifikation

Aufgabe 1 (Verständnis)

Bestimmen Sie bei folgenden Aussagen, ob sie korrekt oder falsch sind *und* erläutern Sie jeweils mit einem Satz.

1. Spezifikation beantwortet die Frage, *wie* ein System eine Leistung erbringt.
2. Die Spezifikation einer Prozedur oder Methode p lässt sich als Abbildung von Eingabewerten auf Ausgabewerte darstellen.
3. Die folgenden Notationen und Namen werden innerhalb der Vorlesung verwendet: Effekt: P , Post-Condition; Voraussetzung: Q , Pre-Condition; Modell: $model$; Invariante: inv .
4. Die Invariante lässt sich als Verpflichtung interpretieren, zu beweisen, dass jede Operation stets die Invariante impliziert.
5. Im Gegensatz zu funktionaler Spezifikation wird bei prädikativer Spezifikation nur auf die Zustände vor und nach der Operation Wert gelegt, und die genaue Abbildung nicht betrachtet.

Diskutieren Sie auf einer halben Seite die Vor- und Nachteile der Verwendung der folgenden Spezifikationsformen, die sich bezüglich ihrer Formalität unterscheiden:

- Formale Spezifikation
- Formalsprachlich
- Halbformale Spezifikation
- Umgangssprachliche Spezifikation
- Spezifikation anhand von Beispielen

Aufgabe 2 (Spezifikation)

In Aufgabe 3.2 wurde eine einfache Speicherverwaltung simuliert. Zur Erinnerung (hier könnte natürlich auch die Haskell-Typklasse und ein Konstruktor stehen):

```
interface Memory {  
    void init(int size);  
    int allocate(int length) throws NoSpaceException;  
    void deallocate(int start, int end);  
}
```

- a) Gesucht sind nun ein geeignetes Modell und eine zugehörige Invariante.
- b) Für die beiden Operationen sollen umgangssprachlich - aber bezogen auf das Modell - Voraussetzungen und Effekte angegeben werden. Dabei soll für `allocate` bewusst offengelassen werden, welcher Bereich (von eventuell mehreren in Frage kommenden Bereichen) belegt wird.
- c) Wie b), aber formal (in mathematischer Schreibweise).
- d) [optional] Wie b) aber formalsprachlich (in Pseudo-Haskell).
- e) Beweisen Sie die Korrektheit Ihrer Spezifikation aus Teil c)! Hierzu gilt es zu zeigen, dass für alle Operationen (inklusive `init`) die Voraussetzungen erfüllbar sind und eine erfüllte Voraussetzung auch immer einen erfüllten Effekt impliziert (dies war übrigens eine halbformale Spezifikation der Aufgabenstellung ;-).