



Programming by Contract

SS 2009 – Übungsblatt 6

Ausgabe: 8. Juni 2009

Abgabe: bis spätestens 17. Juni 2009
in der Vorlesung

Aufgabe 1. *name_list*

Erläutern Sie die zweite und dritte Nachbedingung der Methode `name_list::put()` (Materialsammlung Seite 25), indem Sie sie mit Hilfe des (mathematischen) Operators \Rightarrow darstellen.

Erweitern Sie die Vorbedingung auf `true` und ergänzen Sie zwei weitere Nachbedingungen der Form `ENSURE(not_in_list || ...)`.

Aufgabe 2. *Stack*

Modifizieren Sie die denotationale Spezifikation 1.2.2 der Materialsammlung zur Spezifikation eines Stacks.

Aufgabe 3. *Kindklassen*

Können in C++ Kindklassen einzelne Attribute oder Methoden der Elternklasse nicht besitzen? Wenn ja: Wie kann das realisiert werden, und ist es sinnvoll?

Aufgabe 4. *Umgangssprachliche Spezifikation*

Zeigen Sie, dass das Parkplatzproblem genau eine Lösung der in der Vorlesung spezifizierten Art für alle Eingabegrößen besitzt, die die Vorbedingungen erfüllen. Warum sollte der Auftraggeber der Software `Parkplatzproblem` mit den Vorbedingungen einverstanden sein? Wie könnte eine Softwarelösung aussehen, die ohne den obigen Beweis die Einhaltung des Contracts für sich in Anspruch nimmt?