Theorie der Programmierung I
WS 2007/2008

Leitfaden für die Prüfungsvorbereitung

Die folgenden Fragen sollen ein „Leitfaden“ zu Vorbereitung auf die Prüfung sein, das heisst diese Fragen sollte man sich selbst beim Lernen stellen, und folglich in der Prüfung beantworten können. Das bedeutet nicht, dass in der Prüfung nicht auch andere Fragen gestellt werden können, oder diese Fragen in abgewandelter Form. Es handelt sich nicht um einen vollständigen Fragenkatalog, den es gilt, auswendig zu lernen.

Reiner, ungetypter λ-Kalkül

- Wie sind die gültigen Ausdrücke definiert?
- Welche Programme lassen sich im reinen, ungetypten λ-Kalkül schreiben?

Operationelle step Semantik

- Was ist ein Wert v (formale Definition, intuitive Vorstellung)?
- Was versteht man unter gebundener Umbenennung und wozu braucht man sie (Beispiel)?

Small step Semantik

- Wie sieht ein small step aus (formale Definition, Beispiel)?
- Wie sehen small step Regeln aus? Nach welchem Kriterium kann man small step Regeln in zwei Kategorien einteilen? Man sollte hierzu das Prinzip der Regeln verstanden haben, so dass man nicht alle Regeln auswendig lernt, sondern viel mehr diese anhand des Prinzips konstruieren kann.
- Welches Konzept liegt der Auswertung zu Grund (Erklärung, intuitive Vorstellung)? Stichwort: *term rewriting*.
- Wie spielen die unterschiedlichen Regeln zusammen? Zum Beispiel die vier Regeln für die Applikation.
- Wie äussert sich das *call-by-value* Prinzip? Wie müsste man das Regelwerk abändern um eine *call-by-name* Semantik zu erhalten?
- Welche Ausdrücke müssen aus Sicht der Semantik in der Kernsyntax vorhanden sein? Welche können als syntaktischer Zucker eingeführt werden?
- Nennen Sie Beispiele für abgeleitete small step Regeln für den syntaktischen Zucker. Begründen Sie, warum es sich um abgeleitete Regeln handelt.
- Kann man jedem (wohlgetypten) Ausdruck einen Wert zuordnen? Begründen Sie Ihre Antwort.
- Nennen Sie die vier Möglichkeiten für Berechnungen. Geben Sie jeweils einen Beispielausdruck an.
- Welche wichtige Eigenschaft besitzt unsere Semantik? (Determinismus) Geben Sie auch die Beweisidee an.

Rekursion

- Wie wird Rekursion in die Sprache eingebaut?
- Geben Sie zwei Beispiele für die Definition der Fakultätsfunktion.

Big step Semantik

- Wie sieht ein big step aus? Nennen Sie big step Regeln.
- Welcher Zusammenhang besteht zwischen big und small step Semantik?
- Wie muss man die Regeln ändern, um eine *call-by-name* Semantik zu erhalten?

Typsystem

- Welche Daseinsberichtigung haben Typsysteme? Warum arbeitet man nicht mit ungetypten Sprachen?
- Was ist eine Typumgebung Γ (formale Definition, intuitive Vorstellung)?
- Was ist ein Typurteil?
- Was bedeutet es für einen Wert v, dass er den Typ **int** hat ($\rightarrow v \in \mathbb{Z}$)?
- Wie sehen die wichtigsten Typregeln aus? Zum Beispiel (APP), (COND), (ABSTR) und (LET).
- Wie kommen die Bindungen in (LET), (ABSTR) und (REC) zum Ausdruck?
- Erläutern Sie, was es heisst, dass die Typregeln *syntaxgesteuert* sind.
- Welche Eigenschaften der Sprache haben wir bewiesen? (Progress, Preservation, Typesafety). Erläutern Sie jeweils auch intuitiv die Bedeutung dieser Eigenschaften, und geben Sie die Beweisidee an.
- Geben Sie Beispiele für Restriktionen durch das Typsystem an. Also Ausdrücke, die zur Laufzeit nicht stecken bleiben, aber trotzdem nicht wohlgetypt sind.
- Welche Möglichkeiten existieren, das Typsystem für eine Implementation brauchbar zu machen?

Typinferenz

- Beschreiben Sie kurz, was Typinferenz ist, und welche Daseinsberechtigung es hat.
- Skizzieren Sie den Unifikationsalgorithmus und beschreiben Sie in eigenen Worten, was Unifikation ist.
- Was versteht man unter der allgemeinsten Lösung? (Intuition und formale Definition)
- Beschreiben Sie die Arbeitsweise des Typinferenzalgorithmus anhand zweier Typregeln Ihrer Wahl. Gehen Sie dabei auch auf die Verbindung zum Unifikationsalgorithmus ein.
- Begründen Sie, warum der Unifikationsalgorithmus korrekt ist.

Nach evtl. Korrekturen ist die aktuellste Version verfügbar im WWW unter:
`https://moodle.uni-siegen.de/course/view.php?id=445`