

Semestralprüfung

Die Semestralprüfung wird mündlich sein. Die 20minütigen Prüfungen werden am Mittwoch, den 18. Juli, von 12 – 13:30 Uhr stattfinden. Hinweise zur Anmeldung folgen rechtzeitig.

Aufgabe 1 (H) (*Das Typsystem von Jinja*)

- Geben Sie ein Programm P , eine Typumgebung E und einen wohlgetypten Ausdruck e an, der eine If-Anweisung enthält. Leiten Sie den Typ von e gemäß der Typisierungsregeln von Jinja her. Geben Sie den Ableitungsbaum an.
- Zeigen Sie, dass der Typ, der einem Ausdruck e für ein Programm P und eine Typumgebung E zugeordnet wird, eindeutig ist.

Hinweis: Führen Sie den Beweis mit struktureller Induktion über den Ausdruck. Zur Vereinfachung betrachten Sie nicht alle Ausdrücke, sondern nur Val, binäre Operation, Sequenz, If und Methodenaufruf.

Aufgabe 2 (Ü) (*Subsumption*)

In den Typisierungsregeln von Jinja kommt an mehreren Stellen die Teiltypen-Relation \leq vor. Die Regeln sollen nun vereinfacht werden. Hierzu wird die Subsumptions-Regel

$$[e : T; T \leq T'] \implies e : T'$$

zu den Typisierungsregeln hinzugenommen. Diese Regel besagt, dass wann immer einem Ausdruck der Typ T zugeordnet wird, ihm auch jeder Obertyp T' von T zugeordnet wird.

- Vereinfachen Sie die Typisierungsregeln für Variablenzuweisung, Feldzuweisung, If, Binäre Operation und Typcast.
- Diskutieren Sie, welche Änderungen sich hierdurch für das Typsystem insgesamt und für bestimmte Ausdrücke ergeben.

Aufgabe 3 (Ü) (*Überladen von Methoden*)

Im Gegensatz zum Überschreiben (*overriding*) sind beim Überladen (*overloading*) mehrere Methoden mit gleichem Namen aber verschiedenen Paramertypen gleichzeitig sichtbar. Die tatsächlich aufgerufene Methode wird statisch, das heisst vom Compiler, ermittelt.

In dieser Aufgabe soll Jinja um Überladen erweitert werden. Gehen Sie dazu in mehreren Schritten vor.

- Beschreiben Sie ein Prädikat, das die Deklarationsinformation aus dem Programm extrahiert.

- b) Der Methodenaufruf $e.M(es)$ erfolgt in einem Klassenausdruck e . Welche der sichtbaren Methoden sind aus Compilersicht anwendbar? Definieren Sie, welche Methoden anwendbar sind.
- c) Wenn mehrere Methoden anwendbar sind, soll die Methode aufgerufen werden, die am besten „passt“. Hierbei wird z.B. davon ausgegangen, dass eine Methode, die nur auf einem Teiltyp definiert ist, auf diesem Teiltyp besser passt, weil das die „spezifischere“ Definition der Methode ist. Definieren Sie den Begriff „spezifischer“ zwischen zwei Methoden, der diesen Sachverhalt zum Ausdruck bringt.
- d) Geben Sie die Typisierungsregel für den Methodenaufruf an. Damit zur Laufzeit die richtige Methode aufgerufen werden kann, wird der Methodenaufruf in der Semantik mit den Argumenttypen parametrisiert. Das heisst,

$$e.M\{Ts\}(es)$$

ruft im Objekt e die Methode M mit den Argumenttypen Ts und den Argumenten es auf. Dabei ist Ts der Argumenttyp der spezifischsten, anwendbaren Methode. Dieser wird durch die Typisierungsregeln bestimmt.

Aufgabe 4 (P) (Big-Step-Semantik von Jinja)

Implementieren Sie der Regeln der Big-Step-Semantik (ohne Exceptions) von Jinja in Prolog. Sie finden einen Rahmen zu dieser Aufgabe im Verzeichnis

`/usr/proj/semantik/prog/prolog/jinja-big/`.

Ergänzen Sie die Regelrümpfe in der Datei `semantik.pl` und geben Sie Ihre Datei ab. Einige Testbeispiele finden Sie wieder in `dialog.txt`.