

Abstrakte Maschinen

Sommersemester 2006

10. Übungsblatt

Abgabetermin: Do, 20. Juli 2006 in der Vorlesung

Aufgabe 1: Unifikation

6 Punkte

Überprüfen Sie mit Hilfe der in der Vorlesung angegebenen Funktion *unify*, ob die folgenden Terme unifizierbar sind. Falls ja, geben Sie auch die Substitutionen an, die *unify* liefert.

- $z(a(b(D)), d(e(F)), g(H))$ und $z(H, K, g(F))$
- $p(f(g(X)), Y, X)$ und $p(Z, h(X), i(Z))$
- $f(A, g(A, B))$ und $f(g(B, C), g(g(h(t), B), h(t)))$

Aufgabe 2: Occur-Check

6 Punkte

Unifizieren Sie

$$\begin{aligned} t &\equiv p(X_1, \dots, X_n) \\ s &\equiv p(f(X_0, X_0), f(X_1, X_1), \dots, f(X_{n-1}, X_{n-1})) \end{aligned}$$

Vergleichen Sie den Aufwand für die Unifikation mit und ohne Occur-Check.

Aufgabe 3: Arithmetik

8 Punkte

In Prolog können positive ganze Zahlen als *Nachfolger von 0* definiert werden:

$0 \equiv 0$

$1 \equiv \text{succ}(0)$

$2 \equiv \text{succ}(\text{succ}(0))$

$3 \equiv \text{succ}(\text{succ}(\text{succ}(0)))$

...

- Implementieren Sie Prädikate *greater/2*, *add/3* und *mul/3* entsprechend der gleichnamigen mathematischen Funktionen.
z.B. gilt `greater(succ(succ(succ(0))), succ(0))`
und auch `add(succ(succ(0)), succ(0), succ(succ(succ(0))))`.
- Übersetzen Sie die Klauseln des Prädikats *add* in WiM-Code.