
3. Übung zu Theoretische Informatik: Berechenbarkeit und Komplexität

Wintersemester 2024/25

zu lösen bis 14. November 2024

Aufgabe 3.1:

Gilt $s[a := b][c := d] = s[c := d][a := b]$? Begründen Sie.

Aufgabe 3.2:

Beweisen Sie:

1. Für jeden goto-Befehl p ist die Relation step_p eine partielle Funktion.
2. Für jedes goto-Programm p ist die Relation $\text{step}_p^* \cap \{(C_1, C_2) \mid C_2 \text{ ist final}\}$ eine partielle Funktion.

(Wiederholung der Begriffe und Notationen für Relationen und partielle Funktionen)

Aufgabe 3.3:

Geben Sie goto-Programme zur Berechnung der folgenden Anweisungen (Unterprogramme) an und zeigen Sie, dass Ihre Programme zum gewünschten Ziel führen.

1. Zuweisung $s(i) := 0$
2. Zuweisung $s(i) := s(j)$
3. Verzweigung `if (s(i) = s(j)) p else q`

Was ist beim Einfügen dieser Unterprogramme zu beachten?

Aufgabe 3.4:

Sei A die Menge aller partiellen Funktionen, die durch ein goto-Programm berechnet werden können, in dem der Befehl Goto nicht vorkommt. Beweisen Sie $\text{GOTO} = A$.

Das sind zwei Inklusionen, die eine ist trivial, für die andere übersetzen Sie einen unbedingten in einen bedingten Sprung.

Aufgabe 3.5:

Sei B die Menge aller partiellen Funktionen, die durch ein goto-Programm berechnet werden können, in dem der Befehl GotoZ nicht vorkommt (nur unbedingte Sprünge),

Geben Sie ein Verfahren an, das entscheidet, ob ein B -Programm eine totale Funktion berechnet.

Aufgabe 3.6:

Sei C die Menge aller partiellen Funktionen, die durch ein goto-Programm berechnet werden können, in dem weder Goto noch GotoZ vorkommen.

Welche dieser Programme berechnen totale Funktionen?

Beweisen Sie $B = C$. (zwei Inklusionen)