

Übersetzung $C_0 \rightarrow AM_0$

Übungsblatt 10

ERIC KUNZE — 22. JUNI 2022

Dieses Werk ist lizenziert unter einer [Creative Commons](#) “Namensnennung – Nicht-kommerziell – Weitergabe unter gleichen Bedingungen 4.0 International” Lizenz.



Keine Garantie auf Vollständigkeit und/oder Korrektheit!

Rahmenwerk

```
trans( #include <stdio.h> int main() block) := blocktrans(block)
blocktrans({decl statseq return 0;}) := stseqtrans(statseq, update(decl, tab0), 1)
stseqtrans(stat1stat2...statn, tab, a) :=
    sttrans(stat1, tab, a.1)
    sttrans(stat2, tab, a.2)
    ...
    sttrans(statn, tab, a.n)
update(const id1=z1,...,idn=zn; intid'1,...,id'm; tab) :=
    tab[id1/(const, z1), ..., idn/(const, zn), id'1/(var, 1), ..., id'm/(var, m)]
```

Sequenzen & Zuweisungen

```
sttrans({stat1stat2...statn}, tab, a) := stseqtrans(stat1stat2...statn, tab, a)
sttrans(id = exp;, tab, a) :=
    if tab(id) = (var, n) then simplexprtrans(exp, tab) STORE n;
```

Input/Output

```
sttrans(scanf("%d", &id);, tab, a) := if tab(id) = (var, n) then READ n;
sttrans(sprintf("%d", id);, tab, a) := if tab(id) = (var, n) then WRITE n;
```

Einfache Expressions

```
boolexptrans(se1 rel se2, tab) :=
    simplexprtrans(se1, tab)
    simplexprtrans(se2, tab)
    REL;
wobei (rel, REL) ∈ {(=, EQ), (!=, NE), (<, LT), (>, GT), (<=, LE), (>=, GE)}
simplexprtrans(x + a * 2, [a/(const, 5), x/(var, 1)]) =
    LOAD 1; LIT 5; LIT 2; MUL; ADD;
```

Verzweigungen

$sttrans(\text{if } (exp) \text{ stat}, tab, a) :=$
 $boolexptrans(exp, tab)$
 JMC $a.1$;
 $sttrans(stat, tab, a.2)$
 $a.1 :$
 $sttrans(\text{if } (exp) \text{ stat}_1 \text{ else } stat_2, tab, a) :=$
 $boolexptrans(exp, tab)$
 JMC $a.1$;
 $sttrans(stat_1, tab, a.2)$
 JMP $a.3$;
 $a.1 : sttrans(stat_2, tab, a.4)$
 $a.3 :$

Schleifen

$sttrans(\text{while } (exp) \text{ stat}, tab, a) :=$
 $a.1 : boolexptrans(exp, tab)$
 JMC $a.2$;
 $sttrans(stat, tab, a.3)$
 JMP $a.1$;
 $a.2 :$