

CS 5363: Type Rules for TL05

Jeffery von Ronne
September 15, 2005

Rules

Top-level and variable declarations

$$\text{PROG} \quad \frac{\Sigma \vdash \mathcal{C}}{\Sigma \vdash \text{PROGRAM } name \text{ BEGIN } \mathcal{C} \text{ END}}$$

(where *name* is any legal identifier)

$$\text{DECLVAR} \quad \frac{\Sigma \mid (\ell \rightarrow \tau) \vdash \text{PROGRAM } name \text{ } \mathcal{D} \text{ BEGIN } \mathcal{C} \text{ END}}{\Sigma \vdash \text{PROGRAM } name \text{ VAR } \ell \text{ AS } \tau \text{ ; } \mathcal{D} \text{ BEGIN } \mathcal{C} \text{ END}}$$

$$\text{SEQ} \quad \frac{\Sigma \vdash \mathcal{C}_1 \quad \Sigma \vdash \mathcal{C}_2}{\Sigma \vdash \mathcal{C}_1 \mathcal{C}_2}$$

Statements

$$\text{IF} \quad \frac{\Sigma \vdash e : bool \quad \Sigma \vdash \mathcal{C}}{\Sigma \vdash \text{IF } e \text{ THEN } \mathcal{C} \text{ END ;}}$$

$$\text{IFELSE} \quad \frac{\Sigma \vdash e : bool \quad \Sigma \vdash \mathcal{C}_1 \quad \Sigma \vdash \mathcal{C}_2}{\Sigma \vdash \text{IF } e \text{ THEN } \mathcal{C}_1 \text{ ELSE } \mathcal{C}_2 \text{ END ;}}$$

$$\text{WHILE} \quad \frac{\Sigma \vdash e : bool \quad \Sigma \vdash \mathcal{C}}{\Sigma \vdash \text{WHILE } e \text{ DO } \mathcal{C} \text{ END ;}}$$

$$\text{ASGN} \quad \frac{\Sigma \vdash \ell : \mathcal{R}(\tau) \quad \Sigma \vdash e : \tau}{\Sigma \vdash \ell := e}$$

$$\text{READINT} \quad \frac{\Sigma \vdash \ell : \mathcal{R}(int)}{\Sigma \vdash \ell := \text{READINT ;}}$$

$$\text{WRITEINT} \quad \frac{\Sigma \vdash e : int}{\Sigma \vdash \text{WRITEINT } e ;}$$

$$\text{WRITELN} \quad \Sigma \vdash \text{WRITELN ;}$$

Expressions and Operators

$$\text{OP2} \quad \frac{\Sigma \vdash e_1 : int \quad \Sigma \vdash e_2 : int}{\Sigma \vdash e_1 \text{ OP2 } e_2 : int}$$

$$\text{OP3} \quad \frac{\Sigma \vdash e_1 : int \quad \Sigma \vdash e_2 : int}{\Sigma \vdash e_1 \text{ OP3 } e_2 : int}$$

$$\text{OP4} \quad \frac{\Sigma \vdash e_1 : int \quad \Sigma \vdash e_2 : int}{\Sigma \vdash e_1 \text{ OP4 } e_2 : bool}$$

Variables and Literals

$$\text{VARREF} \quad \Sigma \mid (\ell \rightarrow \tau) \vdash \ell : \mathcal{R}(\tau)$$

$$\text{DEREF} \quad \frac{\Sigma \vdash \ell : \mathcal{R}(\tau)}{\ell : \tau}$$

$$\text{INDEX} \quad \frac{\Sigma \vdash \ell : \mathcal{R}(\text{ARRAY } \iota \text{ OF } \tau) \quad \Sigma \vdash e : int}{\Sigma \vdash \ell [e] : \mathcal{R}(\tau)}$$

$$\text{FALSE} \quad \Sigma \vdash \text{FALSE} : bool$$

$$\text{TRUE} \quad \Sigma \vdash \text{TRUE} : bool$$

$$\text{NUM} \quad \Sigma \vdash \textit{number} : int$$

(where *number* is literal integer from -2147483648 through 2147483647)

Conventions

\mathcal{C} is a metavariable over sequences of zero or more statements (including the semicolons).

\mathcal{D} is a metavariable over sequences of zero or more variable declarations (including the last semicolon).

e is metavariable over r-values.

τ is a meta-variable over types.

ι is some non-negative integer literal.

$\mathcal{R}(\tau)$ is a type denoting a reference to a memory cell whose contents is of type τ .

ℓ is some program variable name (name of a location in memory).