

# Progetto Compilatori

A.A. 2020/2021

Gaetano Antonucci

Alessio Romano

19 gennaio 2021

## Indice

<b>1</b>	<b>Regole di Type Checking implementate</b>	<b>2</b>
1.1	Tipi Primitivi . . . . .	2
1.2	Dichiarazioni di Variabili . . . . .	2
1.3	Operazioni Unarie . . . . .	2
1.4	Operazioni Binarie . . . . .	2
1.5	Chiamata a Procedura . . . . .	2
1.6	Statement . . . . .	2
1.6.1	if-then . . . . .	2
1.6.2	if-then-else . . . . .	2
1.6.3	if-then-elif-else . . . . .	2
1.6.4	while . . . . .	2
1.6.5	while-return . . . . .	3
1.6.6	readln . . . . .	3
1.6.7	write . . . . .	3
1.6.8	assign . . . . .	3

# 1 Regole di Type Checking implementate

## 1.1 Tipi Primitivi

$$\begin{array}{l} \Gamma \vdash \text{null} : \text{null} \quad \Gamma \vdash \text{true} : \text{boolean} \quad \Gamma \vdash \text{false} : \text{boolean} \\ \Gamma \vdash \text{int} : \text{int} \quad \Gamma \vdash \text{float} : \text{float} \quad \Gamma \vdash \text{string} : \text{string} \quad \Gamma \vdash \text{bool} : \text{boolean} \end{array}$$

## 1.2 Dichiarazioni di Variabili

$$\frac{(x : \tau) \in \Gamma}{\Gamma \vdash x : \tau}$$

## 1.3 Operazioni Unarie

$$\frac{\Gamma \vdash e : \tau_1 \quad \text{optype1}(op, \tau_1) = \tau}{\Gamma \vdash (op \ e) : \tau}$$

## 1.4 Operazioni Binarie

$$\frac{\Gamma \vdash e_1 : \tau_1 \quad \Gamma \vdash e_2 : \tau_2 \quad \text{optype2}(op, \tau_1, \tau_2) = \tau}{\Gamma \vdash (e_1 \ op \ e_2) : \tau}$$

## 1.5 Chiamata a Procedura

$$\frac{\Gamma \vdash f : \tau_i^{i \in 1 \dots n} \rightarrow \tau_j^{j \in 1 \dots m} \quad \Gamma \vdash e_i : \tau_i^{i \in 1 \dots n}}{\Gamma \vdash f(e_i^{i \in 1 \dots n}) : \tau_j^{j \in 1 \dots m}}$$

## 1.6 Statement

### 1.6.1 if-then

$$\frac{\Gamma \vdash e : \text{boolean} \quad \Gamma \vdash \text{stmt}}{\Gamma \vdash \text{if } e \text{ then } \text{stmt} \text{ fi}}$$

### 1.6.2 if-then-else

$$\frac{\Gamma \vdash e : \text{boolean} \quad \Gamma \vdash \text{stmt}_1 \quad \Gamma \vdash \text{stmt}_2}{\Gamma \vdash \text{if } e \text{ then } \text{stmt}_1 \text{ else } \text{stmt}_2 \text{ fi}}$$

### 1.6.3 if-then-elif-else

$$\frac{\Gamma \vdash e_j^{j \in 1 \dots m} : \text{boolean} \quad \Gamma \vdash \text{stmt}_i^{i \in 1 \dots 3}}{\Gamma \vdash \text{if } e_1 \text{ then } \text{stmt}_1 \text{ (elif } e_j^{j \in 2 \dots m} \text{ then } \text{stmt}_2)_t^{t \in 1 \dots k} \text{ else } \text{stmt}_3 \text{ fi}}$$

### 1.6.4 while

$$\frac{\Gamma \vdash e : \text{boolean} \quad \Gamma \vdash \text{stmt}}{\Gamma \vdash \text{while } e \text{ do } \text{stmt} \text{ od}}$$

### 1.6.5 while-return

$$\frac{\Gamma \vdash e : \textit{boolean} \quad \Gamma \vdash \textit{stmt}_1 \quad \Gamma \vdash \textit{stmt}_2}{\Gamma \vdash \textbf{while } \textit{stmt}_1 \textbf{--> } e \textbf{ do } \textit{stmt}_2 \textbf{ od}}$$

### 1.6.6 readln

$$\frac{(x_i^{i \in 1 \dots n} : \tau_i^{i \in 1 \dots n}) \in \Gamma}{\Gamma \vdash \textbf{readln}(x_i^{i \in 1 \dots n})}$$

### 1.6.7 write

$$\frac{\Gamma \vdash e : \tau}{\Gamma \vdash \textbf{write}(e : \tau)}$$

### 1.6.8 assign

$$\frac{(x : \tau) \in \Gamma \quad \Gamma \vdash e : \tau}{\Gamma \vdash x := e}$$