

Computer Language Processing

Exercise Sheet 06

November 3, 2022

Welcome to the sixth exercise session of CS320!

Exercise 1

Consider a simple programming language with integer arithmetic, boolean expressions and user-defined functions.

```
T := Int | Bool | (T1, ..., Tn) => T  
t := true | false | c  
    | t1 == t2 | t1 + t2 | t1 && t2  
    | if (t1) t2 else t3  
    | f(t1, ..., tn) | x
```

Where **c** represents integer literals, == represents equality (between **Int**, as well as between **Bool**), + represents the usual integer addition and && represents conjunction.

The meta-variable **f** refers to names of user-defined function and **x** refers to names of variables. You may assume that you have a fixed environment **e** which contains information about user-defined functions (i.e. the function arguments, their types, the function body and the result type).

Write down the typing rules for this language. (We give you an example for **true**)

$$\frac{}{\Gamma \vdash \mathbf{true} : \mathbf{Bool}}$$

Exercise 2

Consider the following typing rules.

$\frac{\text{VARIABLE } v : T \in \Gamma}{\Gamma \vdash v : T}$	$\frac{\text{INT LITERAL } i \text{ is an integer literal}}{\Gamma \vdash i : \text{Int}}$	$\frac{\text{STRING LITERAL } s \text{ is a string literal}}{\Gamma \vdash s : \text{String}}$	$\frac{\text{UNIT}}{\Gamma \vdash () : \text{Unit}}$
$\frac{\text{BOOLEAN LITERAL } b \in \{\text{true}, \text{false}\}}{\Gamma \vdash b : \text{Boolean}}$	$\frac{\text{ARITH. BIN. OPERATORS } \Gamma \vdash e_1 : \text{Int} \quad \Gamma \vdash e_2 : \text{Int} \quad op \in \{+, -, *, /, \%\}}{\Gamma \vdash e_1 \text{ op } e_2 : \text{Int}}$		
$\frac{\text{ARITH. COMP. OPERATORS } \Gamma \vdash e_1 : \text{Int} \quad \Gamma \vdash e_2 : \text{Int} \quad op \in \{<, <=\}}{\Gamma \vdash e_1 \text{ op } e_2 : \text{Boolean}}$			$\frac{\text{ARITH. NEGATION } \Gamma \vdash e : \text{Int}}{\Gamma \vdash -e : \text{Int}}$
$\frac{\text{BOOLEAN BIN. OPERATORS } \Gamma \vdash e_1 : \text{Boolean} \quad \Gamma \vdash e_2 : \text{Boolean} \quad op \in \{\&\&, \ \}}{\Gamma \vdash e_1 \text{ op } e_2 : \text{Boolean}}$			$\frac{\text{BOOLEAN NEGATION } \Gamma \vdash e : \text{Boolean}}{\Gamma \vdash !e : \text{Boolean}}$
$\frac{\text{STRING CONCATENATION } \Gamma \vdash e_1 : \text{String} \quad \Gamma \vdash e_2 : \text{String}}{\Gamma \vdash e_1 ++ e_2 : \text{String}}$		$\frac{\text{EQUALITY } \Gamma \vdash e_1 : T \quad \Gamma \vdash e_2 : T}{\Gamma \vdash e_1 == e_2 : \text{Boolean}}$	
$\frac{\text{SEQUENCE } \Gamma \vdash e_1 : T_1 \quad \Gamma \vdash e_2 : T_2}{\Gamma \vdash e_1 ; e_2 : T_2}$		$\frac{\text{LOCAL VARIABLE DEFINITION } \Gamma \vdash e_1 : T_1 \quad \Gamma, n : T_1 \vdash e_2 : T_2}{\Gamma \vdash \text{val } n : T_1 = e_1 ; e_2 : T_2}$	
$\frac{\text{FUNCTION/CLASS CONSTRUCTOR INVOCATION } \Gamma \vdash e_1 : T_1 \quad \dots \quad \Gamma \vdash e_n : T_n \quad \Gamma \vdash f : (T_1, \dots, T_n) \Rightarrow T}{\Gamma \vdash f(e_1, \dots, e_n) : T}$			
$\frac{\text{IF-THEN-ELSE } \Gamma \vdash e_1 : \text{Boolean} \quad \Gamma \vdash e_2 : T \quad \Gamma \vdash e_3 : T}{\Gamma \vdash \text{if } (e_1) \{e_2\} \text{ else } \{e_3\} : T}$			$\frac{\text{ERROR } \Gamma \vdash e : \text{String}}{\Gamma \vdash \text{error}(e) : T}$

Assuming an empty initial environment, type check the following expressions. Write down the derivation trees.

1. `3 + 5`
2. `val x: Int = 4; val y: Int = x + x; x * y`

Now, assume that the initial environment is:

`{(x, Boolean), (power, (Int, Int) => Int)}`

type check the following expressions. Write down their derivation trees.

3. `val x: Int = if (x) 1 else 0; x * 3`
4. `val x: Int = 7; if (x < 100) power(x, 10) else error("Too big!")`

Exercise 3

Consider the following typing rules for a simple language with integers, pairs and functions:

$$\frac{n \text{ is an integer literal}}{\Gamma \vdash n : \text{Int}} \quad \frac{\Gamma \vdash e_1 : \text{Int} \quad \Gamma \vdash e_2 : \text{Int}}{\Gamma \vdash e_1 + e_2 : \text{Int}} \quad \frac{\Gamma \vdash e_1 : \text{Int} \quad \Gamma \vdash e_2 : \text{Int}}{\Gamma \vdash e_1 \cdot e_2 : \text{Int}}$$

$$\frac{\Gamma \vdash e_1 : T_1 \quad \Gamma \vdash e_2 : T_2}{\Gamma \vdash (e_1, e_2) : (T_1, T_2)} \quad \frac{\Gamma \vdash e : (T_1, T_2)}{\Gamma \vdash \text{fst}(e) : T_1} \quad \frac{\Gamma \vdash e : (T_1, T_2)}{\Gamma \vdash \text{snd}(e) : T_2}$$

$$\frac{\Gamma \oplus \{(x, T_1)\} \vdash e : T_2}{\Gamma \vdash x \Rightarrow e : T_1 \Rightarrow T_2} \quad \frac{\Gamma \vdash e_1 : T_1 \Rightarrow T_2 \quad \Gamma \vdash e_2 : T_1}{\Gamma \vdash e_1(e_2) : T_2} \quad \frac{(x, T) \in \Gamma}{\Gamma \vdash x : T}$$

Consider the following type derivation, with type variables $\mathbf{T}_1, \dots, \mathbf{T}_5$, where $\Gamma_0 = \emptyset$ and $\Gamma = \{(x, \mathbf{T}_2)\}$:

$$\frac{\frac{\frac{(x, \mathbf{T}_2) \in \Gamma}{\Gamma \vdash x : \mathbf{T}_2}}{\Gamma \vdash \text{fst}(x) : \mathbf{T}_4} \quad \frac{\frac{(x, \mathbf{T}_2) \in \Gamma}{\Gamma \vdash x : \mathbf{T}_2}}{\Gamma \vdash \text{snd}(x) : \mathbf{T}_5}}{\Gamma \vdash \text{fst}(x)(\text{snd}(x)) : \mathbf{T}_3}}{\Gamma_0 \vdash x \Rightarrow \text{fst}(x)(\text{snd}(x)) : \mathbf{T}_1}$$

Circle all the correct answers:

- A. There are no assignments of $\mathbf{T}_1, \dots, \mathbf{T}_5$ such that the resulting derivation is valid.
- B. In all valid derivations, \mathbf{T}_3 is equal to \mathbf{T}_5 .
- C. There does **not** exist valid derivations where \mathbf{T}_1 is Int.
- D. In all valid derivations, \mathbf{T}_2 is equal to $(\mathbf{T}_4, \mathbf{T}_5)$.
- E. In all valid derivations, \mathbf{T}_3 is equal to $\mathbf{T}_2 \Rightarrow \mathbf{T}_1$.

Exercise 4

Infer the type of the following expressions:

1. $x \Rightarrow x + 5$
2. $x \Rightarrow y \Rightarrow x + y$
3. $x \Rightarrow (y \Rightarrow x + 5)$
4. $x \Rightarrow \mathbf{if} (x > 0) x \mathbf{else} -x$
5. $x \Rightarrow \mathbf{if} (x) 1 \mathbf{else} x$
6. $x \Rightarrow \mathbf{if} (x) (y \Rightarrow 0) \mathbf{else} (y \Rightarrow y)$
7. $x \Rightarrow y \Rightarrow x(y) \ \&\& \ y(0)$
8. $x \Rightarrow y \Rightarrow i \Rightarrow x(y) + y(i) + i$