

## LTXL Abstract Syntax

Typing rule/handwriting friendly version of the LTXL abstract syntax:

$e \rightarrow n$	<i>literal integer</i>
$x$	<i>variable</i>
$\ominus e$	<i>unary operator app.</i>
$e \otimes e$	<i>binary operator app.</i>
<b>if</b> $e$ <b>then</b> $e$ <b>else</b> $e$	<i>conditional expression</i>
<b>let</b> $(T \ x = e)^* \text{ in } e$	<i>let-expression</i>

G53CMP: Handout part I lecture 10 – p.1/4

## LTXL Operator Types

Unary LTXL operator types:

$\backslash$	$\text{bool} \rightarrow \text{bool}$	
$-$	$\text{int} \rightarrow \text{int}$	<i>unary minus</i>

Binary LTXL operator types:

$  , \&\&$	$(\text{bool}, \text{bool}) \rightarrow \text{bool}$
$<, ==, >$	$(\text{int}, \text{int}) \rightarrow \text{bool}$
$+, -, *, /$	$(\text{int}, \text{int}) \rightarrow \text{int}$

G53CMP: Handout part I lecture 10 – p.3/4

## LTXL Types

LTXL type syntax:

$T \rightarrow$	<b>int</b>	<i>integer type</i>
	<b>bool</b>	<i>boolean type</i>
	$(T, T)$	<i>product (pair)</i>
	$T \rightarrow T$	<i>function</i>

G53CMP: Handout part I lecture 10 – p.2/4

## LTXL Typing Rules

$\Gamma \vdash n : \text{int}$  (T-LITINT)

$\frac{x : T \in \Gamma}{\Gamma \vdash x : T}$  (T-VAR)

$\frac{\Gamma \vdash \ominus : T_1 \rightarrow T_2 \quad \Gamma \vdash e_1 : T_1}{\Gamma \vdash \ominus e_1 : T_2}$  (T-UNOPAPP)

$\frac{\Gamma \vdash \otimes : (T_1, T_2) \rightarrow T_3 \quad \Gamma \vdash e_1 : T_1 \quad \Gamma \vdash e_2 : T_2}{\Gamma \vdash e_1 \otimes e_2 : T_3}$  (T-BINOPAPP)

$\frac{\Gamma \vdash e_1 : \text{bool} \quad \Gamma \vdash e_2 : T \quad \Gamma \vdash e_3 : T}{\Gamma \vdash \text{if } e_1 \text{ then } e_2 \text{ else } e_3 : T}$  (T-IF)

$\frac{\Gamma \vdash \bar{e}_1 : \bar{T}_1 \quad \Gamma, \bar{x} : \bar{T}_1 \vdash e : T}{\Gamma \vdash \text{let } \bar{T}_1 \ \bar{x} = \bar{e}_1 \text{ in } e : T}$  (T-LET)

G53CMP: Handout part I lecture 10 – p.4/4