Harvard School of Engineering and Applied Sciences — CS 152: Programming Languages

Products and Sums; Recursion; References; Polymorphism; Records; Subtyping Section and Practice Problems

Week 7: Tue Mar 6–Fri Mar 10, 2023

---

1 Products and Sums

For these questions, use the lambda calculus with products and sums (Lecture 13§1.1).

(a) Write a program that constructs two values of type $\textbf{int} + (\textbf{int} \to \textbf{int})$, one using left injection, and one using right injection.

(b) Write a function that takes a value of type $\textbf{int} + (\textbf{int} \to \textbf{int})$ and if the value is an integer, it adds 7 to it, and if the value is a function it applies the function to 42.

(c) Give a typing derivation for the following program.

$$\lambda p\!:\!(\textbf{unit} \to \textbf{int}) \times (\textbf{int} \to \textbf{int}).\ \lambda x\!:\!\textbf{unit} + \textbf{int}.\ \mathsf{case}\ x\ \mathsf{of}\ \#1\ p \mid \#2\ p$$

(d) Write a program that uses the term in part (c) above to produce the value 42.

2 Recursion

(a) Use the $\mu x.\ e$ expression to write a function that takes a natural number n and returns the sum of all even natural numbers less than or equal to n. (You can assume you have appropriate integer comparison operators, and also a modulus operator.)

(b) Try executing your program by applying it to the number 5.

(c) Give a typing derivation for the following program. What happens if you execute the program?

$$\mu p\!:\!(\textbf{int} \to \textbf{int}) \times (\textbf{int} \to \textbf{int}).\ (\lambda n\!:\!\textbf{int}.\ n + 1, \#1\ p)$$

3 References

(a) Give a typing derivation for the following program.

$$\begin{array}{l}\mathsf{let}\ a\!:\!\textbf{int ref} = \mathsf{ref}\ 4\ \mathsf{in} \\ \mathsf{let}\ b\!:\!(\textbf{int} \to \textbf{int})\ \textbf{ref} = \mathsf{ref}\ \lambda x\!:\!\textbf{int}.\ x + 38\ \mathsf{in} \\ !b\ !a\end{array}$$

(b) Execute the program above for 4 small steps, to get configuration $\langle e, \sigma\rangle$. What is an appropriate Σ such that $\emptyset, \Sigma \vdash e\!:\!\tau$ and $\Sigma \vdash \sigma$?

(c) Consider a store $\sigma = [\ell_1 \mapsto 42, \ell_2 \mapsto \lambda n\!:\!\textbf{int}.\ n + 1]$. What is the domain of σ?

Now consider a store type $\Sigma = [\ell_1 \mapsto \textbf{int}, \ell_2 \mapsto \textbf{int} \to \textbf{int}]$. Note that $\mathrm{dom}(\sigma) = \mathrm{dom}(\Sigma)$.

Show that $\emptyset, \Sigma \vdash \sigma$.

4 Parametric polymorphism

(a) For each of the following System F expressions, is the expression well-typed, and if so, what type does it have? (If you are unsure, try to construct a typing derivation. Make sure you understand the typing rules.)

- $\Lambda A.\, \lambda x\!:\!A \rightarrow \textbf{int}.\, 42$
- $\lambda y\!:\!\forall X.\, X \rightarrow X.\, (y\ [\textbf{int}])\ 17$
- $\Lambda Y.\, \Lambda Z.\, \lambda f\!:\!Y \rightarrow Z.\, \lambda a\!:\!Y.\, f\ a$
- $\Lambda A.\, \Lambda B.\, \Lambda C.\, \lambda f\!:\!A \rightarrow B \rightarrow C.\, \lambda b\!:\!B.\, \lambda a\!:\!A.\, f\ a\ b$

(b) For each of the following types, write an expression with that type.

- $\forall X.\, X \rightarrow (X \rightarrow X)$
- $(\forall C.\, \forall D.\, C \rightarrow D) \rightarrow (\forall E.\, \textbf{int} \rightarrow E)$
- $\forall X.\, X \rightarrow (\forall Y.\, Y \rightarrow X)$

5 Records and Subtyping

(a) Assume that we have a language with references and records.

(i) Write an expression with type

$$\{\ \mathit{cell} : \textbf{int ref}, \mathit{inc} : \textbf{unit} \rightarrow \textbf{int}\ \}$$

such that invoking the function in the field *inc* will increment the contents of the reference in the field *cell*.

(ii) Assuming that the variable y is bound to the expression you wrote for part (i) above, write an expression that increments the contents of the cell twice.

(b) The following expression is well-typed (with type **int**). Show its typing derivation. (Note: you will need to use the subsumption rule.)

$$(\lambda x\!:\!\{\mathit{dogs} : \textbf{int}, \mathit{cats} : \textbf{int}\}.\, x.\mathit{dogs} + x.\mathit{cats})\ \{\mathit{dogs} = 2, \mathit{cats} = 7, \mathit{mice} = 19\}$$

(c) Suppose that Γ is a typing context such that

$$\Gamma(a) = \{\mathit{dogs}\!:\!\textbf{int}, \mathit{cats}\!:\!\textbf{int}, \mathit{mice}\!:\!\textbf{int}\}$$
$$\Gamma(f) = \{\mathit{dogs}\!:\!\textbf{int}, \mathit{cats}\!:\!\textbf{int}\} \rightarrow \{\mathit{apples}\!:\!\textbf{int}, \mathit{kiwis}\!:\!\textbf{int}\}$$

Write an expression e that uses variables a and f and has type $\{\mathit{apples} : \textbf{int}\}$ under context Γ, i.e., $\Gamma \vdash e\!:\!\{\mathit{apples}\!:\!\textbf{int}\}$. Write a typing derivation for it.

(d) Which of the following are subtypes of each other?

(a) $\{\mathit{dogs}\!:\!\textbf{int}, \mathit{cats}\!:\!\textbf{int}\} \rightarrow \{\mathit{apples}\!:\!\textbf{int}\}$
(b) $\{\mathit{dogs}\!:\!\textbf{int}\} \rightarrow \{\mathit{apples}\!:\!\textbf{int}\}$
(c) $\{\mathit{dogs}\!:\!\textbf{int}\} \rightarrow \{\mathit{apples}\!:\!\textbf{int}, \mathit{kiwis}\!:\!\textbf{int}\}$
(d) $\{\mathit{dogs}\!:\!\textbf{int}, \mathit{cats}\!:\!\textbf{int}, \mathit{mice}\!:\!\textbf{int}\} \rightarrow \{\mathit{apples}\!:\!\textbf{int}, \mathit{kiwis}\!:\!\textbf{int}\}$
(e) $(\{\mathit{apples}\!:\!\textbf{int}\})\ \textbf{ref}$
(f) $(\{\mathit{apples}\!:\!\textbf{int}, \mathit{kiwis}\!:\!\textbf{int}\})\ \textbf{ref}$
(g) $(\{\mathit{kiwis}\!:\!\textbf{int}, \mathit{apples}\!:\!\textbf{int}\})\ \textbf{ref}$

For each such pair, make sure you have an understanding of *why* one is a subtype of the other (and for pairs that aren't subtypes, also make sure you understand).