Exercises marked ⋆ are more advanced. Please try all unstarred exercises before the tutorial meeting.

1. **Comparing large-step and small-step derivations**
   Write both large-step and small-step derivations for the following expressions. For the small-step derivations, construct the derivations of each $e \rightarrow e'$ step explicitly.
   (a) $(\lambda x. x + 1) \ 42$
   (b) $(\lambda x. \text{if } x == 1 \ \text{then } 2 \ \text{else } x + 1) \ 42$

2. **Small-step derivations that go wrong**
   For each of the following expressions, show the small-step evaluation leading to the point where evaluation becomes stuck due to a dynamic type error. (There is no need to show the derivations of each step.)
   (a) $((\lambda x. \lambda y. \text{let } z = x + y \ \text{in } z + 1) \ 42) \ \text{true}$
   (b) $(\lambda x. \text{if } x \ \text{then } x + 1 \ \text{else } x + 2) \ \text{true}$

3. **Small-step rules for $L_{\text{Data}}$**
   Recall that we defined the semantics for $L_{\text{Data}}$ using big-step rules, as follows:

   \[
   e \downarrow v
   \]

   \[
   \begin{array}{llll}
   e_1 \downarrow v_1 & e_2 \downarrow v_2 & e \downarrow (v_1, v_2) & e \downarrow (v_1, v_2) \\
   (e_1, e_2) \downarrow (v_1, v_2) & \text{fst } e \downarrow v_1 & \text{snd } e \downarrow v_2 & \\
   e \downarrow v & e \downarrow \text{left}(v_1) & e_1[v_1/x] \downarrow v & \\
   \text{left}(e) \downarrow \text{left}(v) & \text{case } e \text{ of } \{ \text{left}(x) \Rightarrow e_1; \text{right}(y) \Rightarrow e_2 \} \downarrow v & \\
   e \downarrow v & e \downarrow \text{right}(v_2) & e_2[v_2/x] \downarrow v & \\
   \text{right}(e) \downarrow \text{right}(v) & \text{case } e \text{ of } \{ \text{left}(x) \Rightarrow e_1; \text{right}(y) \Rightarrow e_2 \} \downarrow v & \\
   \end{array}
   \]

   (a) For each construct, write out equivalent small-step rules. Are there any design choices in translating the big-step rules to small-step rules?
   (b) (⋆) Construct small-step derivations reducing the following expressions to values:
   i. $(\lambda p. (\text{snd } p, \text{fst } p + 2)) \ (17, 42)$
   ii. $(\lambda x. \text{case } x \text{ of } \{ \text{left}(y). y + 1; \text{right}(z). z \}) \ (\text{left}(42))$
4. (⋆) **Type soundness for nondeterminism**

This question builds on the *nondeterministic choice* construct mentioned in an earlier tutorial, with the following typing rules:

\[
\Gamma \vdash e : \tau
\]

\[
\frac{\Gamma \vdash e_1 : \tau \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash e_1 \land e_2 : \tau}
\]

and small-step evaluation rules:

\[
e \rightarrow e'
\]

\[
e_1 \land e_2 \rightarrow e_1 \quad e_1 \land e_2 \rightarrow e_2
\]

(a) State the *preservation* property. Outline how we could prove the cases of preservation for nondeterministic expressions.

(b) State the *progress* property. Outline how we could prove the cases of progress for nondeterministic expressions.