Fundamentals of Programming Languages
Assignment 4
Universal Types

Mestrado em (Engenharia) Informática
Faculdade de Ciências da Universidade de Lisboa

2022/2023

1 Binary Sums in System F

1. type Sum = ∀ T U V . (T -> V) -> (U -> V) -> V

2. inl : ∀ T U . T -> (∀ V . (T -> V) -> (U -> V) -> V)
   inl = \T . \U . \l:T . \V . \t:(T -> V) . \u:(U -> V) . t l

3. inr : ∀ T U . U -> (∀ V . (T -> V) -> (U -> V) -> V)
   inr = \T . \U . \r:U . \V . \t:(T -> V) . \u:(U -> V) . u r

4. cases : ∀ T U V . (∀ V . (T -> V) -> (U -> V) -> V) -> (T -> V) -> (U -> V) -> V
   cases = \T . \U . \V . \v:(∀ V . (T -> V) -> (U -> V) -> V) . \lc:(T -> V) . \rc:(U -> V) . v [V] lc rc

5. cases [T] [U] [T] (inl [T] [U] v) (\ x_1 . t_1) (\ x_2 . t_2) =
   (\V . \t:(T -> V) . \u:(U -> V) . t v) [T] (\ x_1 . t_1) (\ x_2 . t_2) =
   (\t:(T -> T) . \u:(U -> T) . t v) (\ x_1 . t_1) (\ x_2 . t_2) =
   (\u:(U -> T) . (\ x_1 . t_1) v) (\ x_2 . t_2) =
   (\ x_1 . t_1) v

   This result is equivalent to [x_1 -> v]t_1, proving that it reduces to the latter result.
   We can notice that this looks similar to a more generic implementation of what we have in fromL (2.2).

6. cases [T] [U] [U] (inr [T] [U] v) (\ x_1 . t_1) (\ x_2 . t_2) =
   (\V . \t:(T -> V) . \u:(U -> V) . u v) [U] (\ x_1 . t_1) (\ x_2 . t_2) =
   (\t:(T -> U) . \u:(U -> U) . u v) (\ x_1 . t_1) (\ x_2 . t_2) =
   (\u:(T -> U) . u v) (\ x_2 . t_2) =
   (\ x_2 . t_2) v

   This result is equivalent to [x_2 -> v]t_2, proving that it reduces to the latter result.
   We can notice that this looks similar to a more generic implementation of what we have in fromR (2.3).