Programming Paradigms Fall 2022 — Problem Sets

by Nikolai Kudasov

September 7, 2022

1 Problem set №1 (solutions)

1. For each of the following λ-terms write down an α-equivalent term where all variables have different names:

   (a) $\lambda x.(\lambda y.x\ y)\ x$
   Solution. In this term, none of the λ-abstractions introduces a name shadowing, so renaming is not necessary. Still, we could rename $x \mapsto a$ and $y \mapsto b$ to get $\lambda a.(\lambda b.a\ b)\ a$.

   (b) $\lambda x.(\lambda x.x)\ x$
   Solution. In this term, nested λ-abstraction shadows outer x variable. We will rename inner variable $x \mapsto y$ to get $\lambda x.(\lambda y.y)\ x$

   (c) $\lambda x.\lambda y.x\ y$
   Solution. In this term, none of the λ-abstractions introduces a name shadowing, so renaming is not necessary. Still, we could rename $x \mapsto a$ and $y \mapsto b$ to get $\lambda a.\lambda b.a\ b$.

   (d) $\lambda x.x\ (\lambda x.x)$
   Solution. In this term, nested λ-abstraction shadows outer x variable. We will rename inner variable $x \mapsto y$ to get $\lambda x.x(\lambda y.y)$

   (e) $\lambda x.(\lambda x.x)\ x$
   Solution. Accidentally, this is exactly the same as (b).

   (f) $(\lambda x.\lambda y.y)\ z\ x$
   Solution. In this term, λ-abstraction $\lambda x.$ shadows free variable x. We will rename the bound variable $x \mapsto w$ to get $(\lambda w.\lambda y.y)\ z\ x$. Note that free variables z and x cannot be renamed (only bound variables can be renamed).

2. Write down evaluation sequence for the following λ-terms:

   (a) $(\lambda x.\lambda y.x)\ y\ z$
   Solution. Note that we need to rename bound varialbe y before we can proceed with β-reduction:

$$
\begin{align}
& (\lambda x.\lambda y.x)\ y\ z \tag{1}\\
& \stackrel{\alpha}{=} (\lambda x.\lambda b.x)\ y\ z \tag{2}\\
& \mapsto (\lambda b.y)\ z \tag{3}\\
& \mapsto y \tag{4}
\end{align}
$$

   (b) $(\lambda x.\lambda y.x)\ (\lambda z.y)\ z\ w$
   Solution. Note that we need to rename bound varialbe y before we can proceed with β-reduction (because $y \in FV(\lambda z.y)$):

$$
\begin{align}
& (\lambda x.\lambda y.x)\ (\lambda z.y)\ z\ w \tag{5}\\
& \stackrel{\alpha}{=} (\lambda x.\lambda b.x)\ (\lambda z.y)\ z\ w \tag{6}\\
& \mapsto (\lambda b.(\lambda z.y))\ z\ w \tag{7}\\
& \stackrel{\alpha}{=} (\lambda b.(\lambda c.y))\ z\ w \tag{8}\\
& \mapsto (\lambda c.y)\ w \tag{9}\\
& \mapsto y \tag{10}\\
& \tag{11}
\end{align}
$$

   (c) $(\lambda b.\lambda f.\lambda t.b\ t\ f)\ (\lambda f.\lambda t.t)$
   Solution.

$$
\begin{align}
& (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (\lambda f.\lambda t.t) \tag{12}\\
& \mapsto \lambda f.\lambda t.(\lambda f.\lambda t.t)\ t\ f \qquad \text{(call-by-value and call-by-name stop here)} \tag{13}\\
& \mapsto \lambda f.\lambda t.(\lambda t.t)\ f \tag{14}\\
& \mapsto \lambda f.\lambda t.f \qquad \text{(full β-reduction stops here)} \tag{15}
\end{align}
$$

(d) $(\lambda s.\lambda z.s\ (s\ z))\ (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (\lambda f.\lambda t.t)$

Solution. First, let us introduce the following aliases:

$$A = \lambda s.\lambda z.s\ (s\ z)$$
$$B = \lambda b.\lambda f.\lambda t.b\ t\ f$$
$$C = \lambda f.\lambda t.t$$

Notice that all these subterms are closed, which means that substituting them into the bodies of λ-abstractions will not cause a name conflict, when passing any of these as argument (so renaming is not necessary). Now, we can write the given term more compactly:

$$(\lambda s.\lambda z.s\ (s\ z))\ (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (\lambda f.\lambda t.t) \tag{16}$$
$$\stackrel{\text{def}}{=} (\lambda s.\lambda z.s\ (s\ z))\ B\ C \tag{17}$$
$$\mapsto (\lambda z.B\ (B\ z))\ C \tag{18}$$
$$\mapsto B\ (B\ C) \tag{19}$$
$$\stackrel{\text{def}}{=} (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (B\ C) \tag{20}$$
$$\mapsto \lambda f.\lambda t.(B\ C)\ t\ f \qquad \text{(call-by-value and call-by-name stop here)} \tag{21}$$
$$\stackrel{\text{def}}{=} \lambda f.\lambda t.((\lambda b.\lambda f.\lambda t.b\ t\ f)\ C)\ t\ f \tag{22}$$
$$\mapsto \lambda f.\lambda t.(\lambda f.\lambda t.C\ t\ f)\ t\ f \tag{23}$$
$$\stackrel{\alpha}{=} \lambda f_1.\lambda t_1.(\lambda f_2.\lambda t_2.C\ t_2\ f_2)\ t_1\ f_1 \tag{24}$$
$$\mapsto \lambda f_1.\lambda t_1.(\lambda t_2.C\ t_2\ t_1)\ f_1 \tag{25}$$
$$\mapsto \lambda f_1.\lambda t_1.C\ f_1\ t_1 \tag{26}$$
$$\stackrel{\text{def}}{=} \lambda f_1.\lambda t_1.(\lambda f.\lambda t.t)\ f_1\ t_1 \tag{27}$$
$$\mapsto \lambda f_1.\lambda t_1.(\lambda t.t)\ t_1 \tag{28}$$
$$\mapsto \lambda f_1.\lambda t_1.t_1 \qquad \text{(full } \beta\text{-reduction stops here)} \tag{29}$$

(e) $(\lambda s.\lambda z.s\ (s\ z))\ (\lambda s.\lambda z.s\ (s\ z))\ (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (\lambda f.\lambda t.t)$

Solution. First, let us introduce the following aliases:

$$A = \lambda s.\lambda z.s\ (s\ z)$$
$$B = \lambda b.\lambda f.\lambda t.b\ t\ f$$
$$C = \lambda f.\lambda t.t$$

Notice that all these subterms are closed, which means that substituting them into the bodies of λ-abstractions will not cause a name conflict, when passing any of these as argument (so renaming is not necessary). Now, we can write the given term more compactly:

$$(\lambda s.\lambda z.s\ (s\ z))\ (\lambda s.\lambda z.s\ (s\ z))\ (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (\lambda f.\lambda t.t) \tag{30}$$
$$\stackrel{\text{def}}{=} (\lambda s.\lambda z.s\ (s\ z))\ A\ B\ C \tag{31}$$
$$\mapsto (\lambda z.A\ (A\ z))\ B\ C \tag{32}$$
$$\mapsto A\ (A\ B)\ C \tag{33}$$
$$\stackrel{\text{def}}{=} (\lambda s.\lambda z.s\ (s\ z))\ (A\ B)\ C \tag{34}$$
$$\mapsto (\lambda z.(A\ B)\ ((A\ B)\ z))\ C \tag{35}$$
$$\mapsto (A\ B)\ ((A\ B)\ C)) \tag{36}$$
$$\stackrel{\text{def}}{=} ((\lambda s.\lambda z.s\ (s\ z))\ B)\ (A\ B\ C) \tag{37}$$
$$\mapsto (\lambda z.B\ (B\ z))\ (A\ B\ C) \tag{38}$$
$$\mapsto B\ (B\ (A\ B\ C)) \tag{39}$$
$$\stackrel{\text{def}}{=} (\lambda b.\lambda f.\lambda t.b\ t\ f)\ (B\ (A\ B\ C)) \tag{40}$$
$$\mapsto \lambda f.\lambda t.(B\ (A\ B\ C))\ t\ f \qquad \text{(call-by-value and call-by-name stop here)} \tag{41}$$
$$\stackrel{\text{def}}{=} \lambda f.\lambda t.((\lambda b.\lambda f.\lambda t.b\ t\ f)\ (A\ B\ C))\ t\ f \tag{42}$$
$$\mapsto \lambda f.\lambda t.(\lambda f.\lambda t.(A\ B\ C)\ t\ f)\ t\ f \tag{43}$$
$$\stackrel{\alpha}{=} \lambda f_1.\lambda t_1.(\lambda f_2.\lambda t_2.(A\ B\ C)\ t_2\ f_2)\ t_1\ f_1 \tag{44}$$
$$\mapsto \lambda f_1.\lambda t_1.(\lambda t_2.(A\ B\ C)\ t_2\ t_1)\ f_1 \tag{45}$$
$$\mapsto \lambda f_1.\lambda t_1.(A\ B\ C)\ f_1\ t_1 \tag{46}$$
$$\stackrel{\alpha}{=} \lambda f_1.\lambda t_1.(A\ B\ C)\ f_1\ t_1 \tag{47}$$
$$\mapsto^* \lambda f_1.\lambda t_1.(\lambda f_3.\lambda t_3.t_3)\ f_1\ t_1 \qquad \text{(from exercise 2d)} \tag{48}$$
$$\mapsto \lambda f_1.\lambda t_1.(\lambda t_3.t_3)\ t_1 \tag{49}$$
$$\mapsto \lambda f_1.\lambda t_1.t_1 \qquad \text{(full } \beta\text{-reduction stops here)} \tag{50}$$

3. Recall that with Church booleans we have the following encoding:

$$\begin{aligned} \mathsf{tru} &= \lambda t.\lambda f.t \\ \mathsf{fls} &= \lambda t.\lambda f.f \end{aligned}$$

(a) Using only bare λ-calculus (variables, λ-abstraction and application), write down a λ-term for logical implication implies of two Church booleans.
Solution. Recall that *"A implies B"* is true when B is true or A is false. We can encode that as *"if A then B else* tru*"*. Recall definition of test from the lab:

$$\mathsf{test} = \lambda c.\lambda t.\lambda f.c\ t\ f$$

With these definitions, we can now define implies as follows:

$$\begin{aligned} \mathsf{implies} &= \lambda a.\lambda b.\mathsf{test}\ a\ b\ \mathsf{tru} \\ &= \lambda a.\lambda b.(\lambda c.\lambda t.\lambda f.c\ t\ f)\ a\ b\ (\lambda t.\lambda f.t) \end{aligned}$$

Note, that $\mathsf{test}\ a\ b\ \mathsf{tru} \mapsto^* a\ b\ \mathsf{tru}$, so we can simplify our definition slightly. Unfolding definition of tru we get

$$\mathsf{implies}' = \lambda a.\lambda b.a\ b\ (\lambda t.\lambda f.t)$$

(b) Verify your implementation of implies by writing down evaluation sequence for the term implies fls tru.
Solution.

$$\mathsf{implies\ fls\ tru} \tag{51}$$
$$\stackrel{\text{def}}{=} (\lambda a.\lambda b.(\lambda c.\lambda t.\lambda f.c\ t\ f)\ a\ b\ (\lambda t.\lambda f.t))\ \mathsf{fls\ tru} \tag{52}$$
$$\mapsto (\lambda b.(\lambda c.\lambda t.\lambda f.c\ t\ f)\ \mathsf{fls}\ b\ (\lambda t.\lambda f.t))\ \mathsf{tru} \tag{53}$$
$$\mapsto (\lambda c.\lambda t.\lambda f.c\ t\ f)\ \mathsf{fls\ tru}\ (\lambda t.\lambda f.t) \tag{54}$$
$$\mapsto (\lambda t.\lambda f.\mathsf{fls}\ t\ f)\ \mathsf{tru}\ (\lambda t.\lambda f.t) \tag{55}$$
$$\mapsto (\lambda f.\mathsf{fls\ tru}\ f)\ (\lambda t.\lambda f.t) \tag{56}$$
$$\mapsto \mathsf{fls\ tru}\ (\lambda t.\lambda f.t) \tag{57}$$
$$\stackrel{\text{def}}{=} (\lambda t.\lambda f.f)\ \mathsf{tru}\ (\lambda t.\lambda f.t) \tag{58}$$
$$\mapsto (\lambda f.f)\ (\lambda t.\lambda f.t) \tag{59}$$
$$\mapsto \lambda t.\lambda f.t \tag{60}$$
$$\stackrel{\text{def}}{=} \mathsf{tru} \tag{61}$$

4. Recall that with Church numerals we have the following encoding:

$$\begin{aligned}
\mathsf{c}_0 &= \lambda s.\lambda z.z \\
\mathsf{c}_1 &= \lambda s.\lambda z.sz \\
\mathsf{c}_2 &= \lambda s.\lambda z.s\ (s\ z) \\
\mathsf{c}_3 &= \lambda s.\lambda z.s\ (s\ (s\ z)) \\
&\dots
\end{aligned}$$

(a) Using only bare λ-calculus (variables, λ-abstraction and application), write down a single λ-term for each of the following functions on natural numbers:

i. $n \mapsto 2n + 1$

ii. $n \mapsto n^2 + 1$

iii. $n \mapsto 2^n + 1$

iv. $n \mapsto 2^{n+1}$

Solution. Recall definitions from lab:

$$\begin{aligned}
\mathsf{c}_0 &= \lambda s.\lambda z.z \\
\mathsf{c}_1 &= \lambda s.\lambda z.s\ z \\
\mathsf{c}_2 &= \lambda s.\lambda z.s\ (s\ z) \\
\mathsf{inc} &= \lambda n.\lambda s.\lambda z.s\ (n\ s\ z) \\
\mathsf{plus} &= \lambda n.\lambda m.\lambda s.\lambda z.m\ s\ (n\ s\ z) \\
\mathsf{times} &= \lambda n.\lambda m.n\ (plus\ m)\ c_0
\end{aligned}$$

Similarly to times, we define exponentiation:

$$\mathsf{exp} = \lambda n.\lambda m.m\ (\times n)\ c_1$$

With these definitions we define the required functions as follows

$$\begin{aligned}
\mathsf{f}_{\mathrm{i}} &= \lambda n.\mathsf{inc}\ (\mathsf{times}\ c_2\ n) \\
\mathsf{f}_{\mathrm{ii}} &= \lambda n.\mathsf{inc}\ (\mathsf{times}\ n\ n) \\
\mathsf{f}_{\mathrm{iii}} &= \lambda n.\mathsf{inc}\ (\mathsf{exp}\ c_2\ n) \\
\mathsf{f}_{\mathrm{iv}} &= \lambda n.\mathsf{inc}\ (\mathsf{exp}\ c_2\ (\mathsf{inc}\ n))
\end{aligned}$$

Note, that by unfolding definitions and applying η-equivalence ($\lambda x.f\ x =_\eta f$), we could, theoretically, achieve smaller definitions:

$$\begin{aligned}
\mathsf{f}'_{\mathrm{i}} &= \lambda n.\lambda s.\lambda z.s\ (n\ s\ (n\ s\ z)) \\
\mathsf{f}'_{\mathrm{ii}} &= \lambda n.\lambda s.\lambda z.s\ (n\ (n\ s)\ z)) \\
\mathsf{f}'_{\mathrm{iii}} &= \lambda n.\lambda s.\lambda z.s\ (n\ c_2\ s\ z) \\
\mathsf{f}'_{\mathrm{iv}} &= \lambda n.\lambda s.\lambda z.s\ (n\ c_2\ s\ (n\ c_2\ s\ z))
\end{aligned}$$

(b) Verify each one of your implementations of the functions above by writing down evaluation sequence for each of them, when applied to c_2

Solution.

$$
\begin{aligned}
&\mathsf{f}_{\mathrm{i}}\ c_2 && (62)\\
&\stackrel{\text{def}}{=} (\lambda n.\mathsf{inc}\ (\mathsf{times}\ c_2\ n))\ c_2 && (63)\\
&\mapsto \mathsf{inc}\ (\mathsf{times}\ c_2\ c_2) && (64)\\
&\stackrel{\text{def}}{=} (\lambda n.\lambda s.\lambda z.s\ (n\ s\ z))\ (\mathsf{times}\ c_2\ c_2) && (65)\\
&\mapsto \lambda s.\lambda z.s\ (\mathsf{times}\ c_2\ c_2\ s\ z) \qquad \text{(call-by-value and call-by-name stop here)} && (66)\\
&\mapsto^* \lambda s.\lambda z.s\ (c_4\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_2 \mapsto^* c_4) && (67)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda s_2.\lambda z_2.s_2\ (s_2\ (s_2\ (s_2\ z_2))))\ s\ z) && (68)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda z_2.s\ (s\ (s\ (s\ z_2))))\ z) && (69)\\
&\mapsto \lambda s.\lambda z.s\ (s\ (s\ (s\ (s\ z)))) && (70)\\
&\stackrel{\text{def}}{=} c_5 && (71)
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{f}_{\mathrm{ii}}\ c_2 && (72)\\
&\stackrel{\text{def}}{=} (\lambda n.\mathsf{inc}\ (\mathsf{times}\ n\ n))\ c_2 && (73)\\
&\mapsto \mathsf{inc}\ (\mathsf{times}\ c_2\ c_2) \qquad (\text{from here it goes exactly like in } \mathsf{f}_{\mathrm{i}}\ c_2) && (74)\\
& && (75)\\
&\stackrel{\text{def}}{=} (\lambda n.\lambda s.\lambda z.s\ (n\ s\ z))\ (\mathsf{times}\ c_2\ c_2) && (76)\\
&\mapsto \lambda s.\lambda z.s\ (\mathsf{times}\ c_2\ c_2\ s\ z) \qquad \text{(call-by-value and call-by-name stop here)} && (77)\\
&\mapsto^* \lambda s.\lambda z.s\ (c_4\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_2 \mapsto^* c_4) && (78)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda s_2.\lambda z_2.s_2\ (s_2\ (s_2\ (s_2\ z_2))))\ s\ z) && (79)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda z_2.s\ (s\ (s\ (s\ z_2))))\ z) && (80)\\
&\mapsto \lambda s.\lambda z.s\ (s\ (s\ (s\ (s\ z)))) && (81)\\
&\stackrel{\text{def}}{=} c_5 && (82)
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{f}_{\mathrm{iii}}\ c_2 && (83)\\
&\stackrel{\text{def}}{=} (\lambda n.\mathsf{inc}\ (\mathsf{exp}\ c_2\ n))\ c_2 && (84)\\
&\mapsto \mathsf{inc}\ (\mathsf{exp}\ c_2\ c_2) && (85)\\
&\stackrel{\text{def}}{=} (\lambda n.\lambda s.\lambda z.s\ (n\ s\ z))\ (\mathsf{exp}\ c_2\ c_2) && (86)\\
&\mapsto \lambda s.\lambda z.s\ (\mathsf{exp}\ c_2\ c_2\ s\ z) \qquad \text{(call-by-value and call-by-name stop here)} && (87)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda n.\lambda m.m\ (\mathsf{times}\ n)\ c_1)\ c_2\ c_2\ s\ z) && (88)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda m.m\ (\mathsf{times}\ c_2)\ c_1)\ c_2\ s\ z) && (89)\\
&\mapsto \lambda s.\lambda z.s\ (c_2\ (\mathsf{times}\ c_2)\ c_1\ s\ z) && (90)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda s_2.\lambda z_2.s_2\ (s_2\ z_2))\ (\mathsf{times}\ c_2)\ c_1\ s\ z) && (91)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda z_2.(\mathsf{times}\ c_2)\ (\mathsf{times}\ c_2\ z_2))\ c_1\ s\ z) && (92)\\
&\mapsto \lambda s.\lambda z.s\ ((\mathsf{times}\ c_2)\ (\mathsf{times}\ c_2\ c_1)\ s\ z) && (93)\\
&\mapsto^* \lambda s.\lambda z.s\ (\mathsf{times}\ c_2\ c_2\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_1 \mapsto^* c_2) && (94)\\
&\mapsto^* \lambda s.\lambda z.s\ (c_4\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_2 \mapsto^* c_4) && (95)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda s_2.\lambda z_2.s_2\ (s_2\ (s_2\ (s_2\ z_2))))\ s\ z) && (96)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda z_2.s\ (s\ (s\ (s\ z_2))))\ z) && (97)\\
&\mapsto \lambda s.\lambda z.s\ (s\ (s\ (s\ (s\ z)))) && (98)\\
&\stackrel{\text{def}}{=} c_5 && (99)
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{f}_{\mathrm{iv}}\ c_2 && (100)\\
&\stackrel{\text{def}}{=} (\lambda n.\mathsf{inc}\ (\mathsf{exp}\ c_2\ (\mathsf{inc}\ n)))\ c_2 && (101)\\
&\mapsto \mathsf{inc}\ (\mathsf{exp}\ c_2\ (\mathsf{inc}\ c_2)) && (102)\\
&\stackrel{\text{def}}{=} (\lambda n.\lambda s.\lambda z.s\ (n\ s\ z))\ (\mathsf{exp}\ c_2\ (\mathsf{inc}\ c_2)) && (103)\\
&\mapsto \lambda s.\lambda z.s\ (\mathsf{exp}\ c_2\ (\mathsf{inc}\ c_2)\ s\ z) \qquad \text{(call-by-value and call-by-name stop here)} && (104)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda n.\lambda m.m\ (\mathsf{times}\ n)\ c_1)\ c_2\ (\mathsf{inc}\ c_2)\ s\ z) && (105)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda m.m\ (\mathsf{times}\ c_2)\ c_1)\ (\mathsf{inc}\ c_2)\ s\ z) && (106)\\
&\mapsto \lambda s.\lambda z.s\ (\mathsf{inc}\ c_2\ (\mathsf{times}\ c_2)\ c_1\ s\ z) && (107)\\
&\mapsto^* \lambda s.\lambda z.s\ (c_3\ (\mathsf{times}\ c_2)\ c_1\ s\ z) && (108)\\
&\stackrel{\text{def}}{=} \lambda s.\lambda z.s\ ((\lambda s_2.\lambda z_2.s_2\ (s_2\ (s_2\ z_2)))\ (\mathsf{times}\ c_2)\ c_1\ s\ z) && (109)\\
&\mapsto \lambda s.\lambda z.s\ ((\lambda z_2.(\mathsf{times}\ c_2)\ ((\mathsf{times}\ c_2)\ (\mathsf{times}\ c_2\ z_2)))\ c_1\ s\ z) && (110)\\
&\mapsto \lambda s.\lambda z.s\ (\mathsf{times}\ c_2\ ((\mathsf{times}\ c_2)\ (\mathsf{times}\ c_2\ c_1))\ s\ z) && (111)\\
&\mapsto^* \lambda s.\lambda z.s\ (\mathsf{times}\ c_2\ (\mathsf{times}\ c_2\ c_2)\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_1 \mapsto^* c_2) && (112)\\
&\mapsto^* \lambda s.\lambda z.s\ (\mathsf{times}\ c_2\ c_4\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_2 \mapsto^* c_4) && (113)\\
&\mapsto^* \lambda s.\lambda z.s\ (c_8\ s\ z) \qquad (\text{since } \mathsf{times}\ c_2\ c_4 \mapsto^* c_8) && (114)\\
&\mapsto^* \lambda s.\lambda z.s\ (s\ (s\ (s\ (s\ (s\ (s\ (s\ (s\ z)))))))) && (115)\\
&\stackrel{\text{def}}{=} c_9 && (116)
\end{aligned}
$$