

1 **CS370 Programming Languages (Zaring)**2 **Spring 2020**3 **Assignment 4**4 **Due by 4:00pm on Tuesday, March 10**

5

6 **Description:**7 In lecture, we introduced the `begin` expression

8

9
$$(\text{begin } \textit{expr}_0 \dots \textit{expr}_{n-1})$$

10

11 and claimed that it's just syntactic sugar. Show that this is indeed the case.

12

13 (a) Explain how every `begin` expression could be rewritten, without using any `begin`
14 expressions at all, into equivalent Scheme code that uses only the Scheme we've seen thus
15 far in lecture and in *TLS*. Your rewriting technique must work for any `begin` expression.16 So, your technique can't depend on the particulars of the subexpressions of the `begin`
17 expression, can't be restricted to working only on `begin` expressions with a limited number
18 of subexpressions, can't be restricted to working only on `begin` expressions with restricted
19 kinds of subexpressions, and so on.20 (b) Show how the `begin` expressions

21

22 (1) `(begin)` *This is indeed permitted; its value is unspecified, according to R6RS.*23 (2) `(begin x_0)`24 (3) `(begin y_0 y_1)`25 (4) `(begin z_0 z_1 z_2)`

26

27 (where the particulars of the x 's, y 's, and z 's are completely unknown) would be rewritten using
28 the technique you explain in (a).

29

30 **What to Hand in:**31 • A PDF of a scholarly-/professional-looking word-processed document with your answers to (a)
32 and (b), submitted using the Assignment 4 item on the Assignments page of the CS37033 Katie course – PDF's consisting of images of hand-written solutions and PDF versions of
34 simple text files are unacceptable.

35 • A printed copy of your PDF