; CSC104 Winter 2020 — Exercise #4 — Print out and fill in by hand, then hand in to the TA at the start of your quiz. 

; UTorID (login ID):

; Surname :

; Given Name :

; Part I.

; Define sesqui so that it behaves as shown:

(set (sesqui 1903)) ; produces the steps ...
(+ 1903 150)
2053

(set (sesqui 2020)) ; produces the steps ...
(+ 2020 150)
2170

; Beside each of these two expressions write its value:

sesqui

(sesqui 1815)

; Show, with standard underlining, the following steps ...

(set (map sesqui (list 1815 1906 1903)))

(set (hide sesqui) (map sesqui (list 1815 1906 1903)))

; Define ! so that it behaves as shown:

(step (! "wow")) ; produces the steps ... (text-join "wow" "!"
"wow!"

(step (! "whatever")) ; produces the steps ... (text-join "whatever" "!"
"whatever!"

; Beside each of these two expressions write its value:

! (! "buddy")

; Show, with standard underlining, the steps for:

(step (hide !) (map ! (list "wow" "whatever" "buddy")))
: Define born-1906? so that it behaves as shown ...

(step (born-1906? (list 1906 "Goedel" "Kurt")))  ; ... produces the steps ...
(same? (first (list 1906 "Goedel" "Kurt")) 1906)  ; ... produces the steps ...
(same? 1906 1906)  ; ... produces the steps ...
#true

(step (born-1906? (list 1815 "Lovelace" "Ada")))

: Beside each of these two expressions write its value ...

born-1906?  ; (born-1906? (list 1906 "Church" "Alonzo"))

: Define text-first so that it behaves as shown :

(step (text-first "ruby"))  ; ... produces the steps ...
(first (text->list "ruby"))  ; ... produces the steps ...
(first (list "r" "u" "b" "y"))  ; ... produces the steps ...
"r"

(step (text-first "jade"))
(first (text->list "jade"))
(first (list "j" "a" "d" "e"))
"j"

: Beside each of these two expressions write its value ...

text-first  ; (text-first "onyx")

: Show, with standard underlining, the steps for ...

(step (hide text-first)  ; (step (map text-first (list "ruby" "jade" "onyx")))
   (map text-first (list "ruby" "jade" "onyx")))

: Part II. Assume the following definitions have been entered/run ...

(define R (random 1000000))
(define (r _) (random 1000000))
(define (Rf _) R)

: ... then under each of these expressions write its value ...

(same? (random 1000000) (same? R R) (same? (r "hmm") (same? (Rf "hmm") (random 1000000) (r "hmm")) (Rf "hmm")) r Rf
: Part III.

Based on this definition ...

\[
\begin{align*}
\text{(define (A n)} & \\
& \triangleleft \\
& \quad \text{(if (same? n 0))} \\
& \quad \text{else (above (A (- n 1))} \\
& \quad \text{(wide (A (- n 1))))})
\end{align*}
\]

... show the steps, with standard underlining, for ...

\[
\begin{align*}
\text{(step (A 0))}
\end{align*}
\]

\[
\begin{align*}
\text{(step (hide (A 0))} & \\
& \quad \text{(A 1))}
\end{align*}
\]

\[
\begin{align*}
\text{(step (hide (A 1))} & \\
& \quad \text{(A 2))}
\end{align*}
\]
Part IV.

Based on these definitions...

\[
\begin{align*}
\text{(define } C \text{ (circle 20))} \quad & \text{(define } \text{ (arrange an-image)} \\
\text{beside } C \text{ (tall an-image) } C) \\
\end{align*}
\]

... show the steps, with standard underlining, for...

\[
\begin{align*}
\text{(step (arrange } C)) \\
\end{align*}
\]

Beside each of these two expressions write its value:

\[
\begin{align*}
C \quad & \text{arrange} \\
\end{align*}
\]

Based on this definition...

\[
\begin{align*}
\text{(define } B \text{ k)} \\
\text{if (same? } k \text{ 0) } C \\
\text{else (arrange } B \text{ (~ k 1)))} \\
\end{align*}
\]

... show the steps, with standard underlining, for...

\[
\begin{align*}
\text{(step } B \text{ 0))} \\
\text{(step (hide } B \text{ 0)} \\
\text{arrange)} \\
\text{(B 1))} \\
\end{align*}
\]

Show the value of:

\[
\begin{align*}
\text{(B 2)} \\
\end{align*}
\]