CS 2500/Accelerated Exam 0—Fall 2017

Matthias Felleisen

September 20, 2017

- The exam is a **one-hour** exam.
- We will not answer any questions during the exam. If you believe a problem statement is ambiguous, choose *any* non-trivial interpretation.
- Write down the answers in the space provided, including the back of the given spaces.
- You may use the paper copy of the book or your notes.
- You may *not* use any electronic gadgets (for example, watches, google glasses, phones, tablets, laptops). Any use of an electronic gadget will lead to immediate expulsion from the exam and class.
- You may use all the definitions, expressions, and functions found BSL, especially those suggested in hints. Define everything else.
- Some basic test taking advice: Before you start answering any problems, read *every* problem, so your brain can be thinking about the harder problems in the background while you knock off the easy ones.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Max. Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>30</strong></td>
</tr>
</tbody>
</table>
Problem 1 Design the function \texttt{adder}, which consumes a \texttt{Posn} and computes the sum of its fields. Show all steps of the design recipe.
Problem 2

(define-struct counting (time-left))
(define-struct going (x y))

; A CD is one of:
; -- "hold"
; -- (make-counting PositiveNumber)
; -- (make-going Number Number)

; interpretation It represents the state of a rocket-launch countdown.

Design the function cd-as-string. It consumes a CD and forms a sentence that describes the current state of a countdown. The sentence starts with "The countdown is " and continues with an appropriate word, depending on which state is given. For example, when given a going structure, the function may add "over." When a counting structure is given, the sentence must involve the number in the time-left field.
intentionally left blank
Problem 3 Design the function inflate. It consumes a menu item and an inflation rate. Its result is a menu item whose price has been inflated at the given rate. The restaurant that you are working for represents menu items as follows:

\[
\text{(define MP "market price")}
\]

\[
\text{(define-struct item (title price))}
\]

; A MenuItem is (make-menu String Price)

; A Price is one of:
; -- Number
; -- MP

Domain knowledge Inflation is the rate at which the price for certain items increases over time. Typical items include phones, cars, food, fuel, and work. If an economy experiences a 10% inflation and a restaurant wishes to keep up with it, it must raise its prices by 10%.

(If prices trend downward, economists speak of a deflation.)