1. Write a record type definition for a record type called \texttt{rgb-color}, with three fields named \texttt{red}, \texttt{green}, and \texttt{blue}. None of the fields should be mutable. Name the constructor \texttt{%make-rgb-color} to indicate that it will be only a helper procedure—the real \texttt{make-rgb-color} constructor will be defined in the next exercise.

2. Each of the components of an \texttt{rgb-color} should be an exact integer in the range from 0 up to, but not including, 256. Using \texttt{%make-rgb-color} as a helper procedure, define a three-argument constructor \texttt{make-rgb-color} that returns an \texttt{rgb-color} color record if all three of its arguments are integers in that range, and otherwise crashes with an appropriate error message. (You can use the \texttt{assert} procedure from the \texttt{(discrete utilities)} library to enforce the preconditions.)

3. Write a record type definition for a record type called \texttt{pixel}, with three fields named \texttt{x-coordinate}, \texttt{y-coordinate}, and \texttt{color}. The \texttt{x-coordinate} and \texttt{y-coordinate} fields should be mutable and so your record definition should be written in such a way that modifier procedures for them are provided automatically.

4. The constructor for the \texttt{pixel} type should ensure that its first two fields are real numbers and that its third field is a record of the \texttt{rgb-color} type. Using the technique illustrated in the second preceding exercise, arrange for the \texttt{make-pixel} constructor to enforce these preconditions.

5. Define, write, and test a Scheme procedure \texttt{move-left} that takes two arguments, a pixel and a real number, and modifies the given pixel by subtracting the given real number from its \texttt{x-coordinate}.

6. Define, write, and test a Scheme procedure \texttt{green-only} that takes a list of pixels as its argument and returns a similar list of pixels with the same \texttt{x}– and \texttt{y}-coordinates, but with each color replaced by an \texttt{rgb-color} that has the same green component as the original color but 0 for its red and blue components.