Whenever we say to "design a function", we mean that you need to follow the design recipe. Any other time that you need to write a function on this exam (or in this class for that matter) you also need to follow the design recipe. As a reminder the design recipe has the following steps:

1. (a) choose a data definition/representation
   (b) give examples (you need to build some data by hand)
   (c) choose the name of your function
   (d) write the signature of your function
   (e) write the purpose statement of your function
   (f) give some examples (you will write check-expect-like statements)
   (g) now write the header of your function and the appropriate function template
   (h) fill in the template by actually completing the function
   (i) actually write the check-expects for testing your function

Here now are your problems (each one is worth 20 points for a total of 100):

1. Provide a structure type definition and a data definition for representing moments in time since midnight. A moment in time consists of three numbers: hours, minutes and seconds. Provide some examples. Write down the names of the functions (constructors, selectors, predicates) your structure type definition creates. Write the template for a function that processes this type of data definitions.

2. Design the function `time->seconds` which consumes instances of the time structures developed in the previous exercise and produces the number of seconds that have passed since midnight. For example if you are representing 12 hours, 30 minutes, and 2 seconds with one of these structures and if you then apply `time->seconds` to this instance, the correct result is 45002.

3. Determine if \((\text{and} \ x \ (\text{or} \ (\text{not} \ x) \ y))\) is equivalent to \((\text{and} \ x \ y)\) for \(x\) and \(y\) booleans.

4. Design a function `contains?` which determines whether some given string occurs in a list of strings.

5. Design a function `all-true`, which consumes a list of boolean values and determines whether all of them are true. In other words, if there is any false on the list, the function produces false; otherwise it produces true.

In class please write your answers on the paper provided. Then turn your exam in. Then go to lab, where you will be reunited with your paper exam. In lab you can’t write or add or erase anything to/from your paper exam. In lab you type your answers in and fix them if needed, also give yourself a grade, accordingly. Turn your (fixed) code and brief grade report into OnCourse, and return your unmodified paper exam to your lab instructor.

1 And this concludes your reference sheet for this exam.