CSE 341 — Haskell Mini-Exercises # 1

These are questions for discussion in class. (You don’t need to hand in anything.) The solutions are on the class web page.

1. Write a Haskell function to find the cube of a Double. What is the type of this function?

2. Write a Haskell function to find the sum of three Doubles. What is the type of this function?

3. Write a Haskell function to find the value of the quadratic expression $ax^2 + bx + c$ for parameters $a$, $b$, $c$, and $x$. What is the type of this function?

4. Write a Haskell function to reverse a list. What is the type of this function?

5. Write a function `my_map2` that is analogous to `map` but works for functions of two arguments rather than one. What is its type? For example,
   ```haskell
   map2 (+) [1,2,3] [10,11,12]
   ```
   should evaluate to `[11,13,15]`

6. Give a recursive definition of a variable `doubles` whose first element is 10, and whose $n^{th}$ element is twice the $n-1^{st}$, i.e. [10, 20, 40, 80, 160, 320, ...].

7. Give an alternate non-recursive definition of `doubles` using the built-in function `iterate` from the Haskell prelude. This is defined as follows:
   ```haskell
   iterate :: (a -> a) -> a -> [a]
   iterate f x = x : iterate f (f x)
   ```

8. Define a Haskell variable `dollars` that is the infinite list of amounts of money you have every year, assuming you start with $100 and get paid 5% interest, compounded yearly. (Ignore inflation, deflation, taxes, bailouts, the possibility of total economic collapse, and other such details.) So `dollars` should be equal to `[100.0, 105.0, 110.25, ...]`.

9. Suppose that the following Haskell script has been filed in.
   ```haskell
   my_const c x = c
   append [] ys = ys
   append (x:xs) ys = x : append xs ys
   my_map f [] = []
   my_map f (x:xs) = f x : my_map f xs
   ```

What is the type of each of the following Haskell expressions? (Some may give an error.)

(a) `my_const`
(b) `my_const True`

(c) `append []`

(d) `append [True, False]`

(e) `append [3] ['a', 'b']`

(f) `append "squid" ['a', 'b']`

(g) `append "octopus" ['a', 'b']`

(h) `my_map`

(i) `my_map (my_const True)`

What is the value of each of the following Haskell expressions?

(a) `my_const 5 "octopus"

(b) `my_map (my_const "squid") [1 ..]`

(c) `my_map sqrt [1, 2, 100]`