CS360
Lab 4

For each of these problems, write your Haskell code in a single file. Name it Lab4.hs Please start with something like this, with your name of course.

```
{-            Lab 4:Becka Morgan -}

-- Example
square :: Integer -> Integer
square x = x^2

{- ANSWER: The square of 93240823948293048 is 8693851250556578380656712885130304 -}

-- Problem #1

{- ANSWER:  -}

-- Problem #2

{- ANSWER:  -}

-- Problem #3

{- ANSWER:  -}
```

For most of the questions you'll write the function or functions and then be asked to use it to provide a resulting value for an example input. Put that answer in comments just after the function, as illustrated above.

Questions: A Bunch of Random Things

1. As a good scientist should, I use SI units in everyday life. For example my thermometers are set to Celsius and I record my fuel efficiency in liters per 100km. Some misguided individuals persist in using archaic units. To help them when traveling to sensible countries, write functions necessary to answer the following questions. How many US gallons are there in a certain number of liters? How much have you spent in USD if you spent $x CAD? Now, use these two functions to answer this question: you filled up your gas tank in Canada. You bought 62.3 liters of fuel and paid 78.4 Canadian dollars. What price was that in US dollars per gallon? To get you started on this one, call your first function `gallons`, the second one `usd` and the last `price`.

2. Write a function called `flightDistance` to compute the approximate distance between two locations on the Earth (in nautical miles[1]) given the latitude and longitude of each coordinate as `Double` values. The needed function is

$$d = a\cos^{-1}\left(\cos(\delta_1)\cos(\delta_2)\cos(\lambda_1 - \lambda_2) + \sin(\delta_1)\sin(\delta_2)\right)$$

where d is the flight distance[2] in miles between two points on a sphere of radius a (in miles), roughly 3963 miles for the Earth. Point one has latitude δ_1 and longitude λ_1 while point 2 has latitude δ_2 and longitude λ_2. The latitude and longitude need to be in decimal form, i.e.
+45.58 latitude and -122.6 longitude is basically 45 degrees North latitude and 122.6 West longitude. Note that these values are in *degrees* and that trigonometry functions usually assume values in *radians*. Also, in case you are unfamiliar with the notation, $\cos^{-1}$ means an inverse, or arc cosine. Important: Please write your function using tuples to represent each coordinate; write a helper function to convert from degrees to radians. *Also you must use a where clause.* What is the flight distance between the point at $45^\circ N, 122^\circ W$ to the point $21^\circ N, 158^\circ W$?

3. Use a list comprehension and the sum function to determine the sum of the cubes of all the odd numbers between 1000 and 2000?

4. Write expressions using `map, filter, any, or all to`

   - remove all spaces from a string
   - filter out all even numbers from a list (use the `even` function)
   - double every value in a list
   - tell you True or False if a list contains the number 55
   - tell you True or False if all the values in a list are odd (use the `odd` function)

5. Write a function called `isPrime` that determines if an `Integer` is a prime number (evenly divisible only by itself and one). For reference, here's a list of the primes less than 100:

   ```
   [2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,97]
   ```

   What are the 1000^{th} through 1020^{th} prime numbers? (starting at 2)

6. Write a function called `factor` that will take an `Integer` and determine its prime factors (called prime factorization). The function should take an `Integer` and return a list of `Integers`. This list should be the prime factors of the number, not including multiplicity. For example, the prime factors of 65 are 2 and 7 (even though the full multiplication is $56 = 2^3 \cdot 7 = 2 \cdot 2 \cdot 2 \cdot 7$). What are the prime factors of 175561 and 62451532000?
7. Create a main function that prints all of your function outputs.