*Your submission for this assignment **must be commented** and **must include both your name** and your **student number** as a comment at the top of **every source file you submit**. Each of your submitted files must use a **file name beginning 'comp3007_w23_#########_assignment_04'** (replacing the number signs with your own student number) and any **submissions that crash** (i.e., terminate with an error) on execution will automatically **receive a mark of 0**.*

Officially, the Due Date for this Assignment is:
Friday, March 3rd, 2021, *at* **11:59pm EST**.

Late Submissions are Accepted Without Penalty Until Sunday, March 5th, by 11:59pm EST. Submissions received after that will not be accepted and will receive a mark of 0.

The objective of this assignment is to allow you to practice with recursive design and the recursive definition of lists in Haskell by designing and implementing functions for determining the divisors for an integer and whether or not that integer is a prime number.

If you do not recall the definition of a prime number, you are directed to first take a few minutes and read the article at https://mathworld.wolfram.com/PrimeNumber.html.

For an additional example of process for determining whether or not a positive integer is prime, consider the following:

A list of the divisors of the positive integer 12, for instance, are [12, 6, 4, 3, 2, 1].
As this list contains elements other than 12 and 1, the positive integer 12 is NOT prime.

A list of the divisors of the positive integer 13, for instance, are [13, 1].
As this list contains only elements 13 and 1, the positive integer 13 IS prime.

For this assignment:

- you must write a recursive function that provides (as a return value) a list of every divisor of the positive integer specified. This function will not be called directly, but will instead be called by a non-recursive function you will also create, using type declaration Integer -> [Integer]. To clarify, the non-recursive function will be called with the integer argument, and this function will call the recursive function and then return the resulting list of Integer divisors.

- you may use the built-in "mod" function for modulation.

- your program will be penalized if this list of divisors is not constructed in descending order (i.e., from largest to smallest), but you are not permitted to apply any sort algorithm to the result.

- you must write another function using type declaration Integer -> Boolean that uses your other functions to determine whether or not the Integer argument is a prime number.

- your function for determining whether or not a number is prime must use the "trace" and "show" functions to justify your answer (by printing the list of divisors). You have used the "trace" function for the previous assignment, and the "show" function takes an argument of any type and produces a string representation of it as a return value.

- you may assume that the integer arguments you receive during testing will always be positive.