Assignment #7
CS 3060 Programming Languages, Spring 2021
Instructor: S. Roy

Haskell #1

Due Date: Apr 13 @ 11.59 PM.
Total Points: 50 points

Directions: Using the source provided via Gitlab `https://gitlab.com/sanroy/sp21-cs3060-hw/`, complete the assignment below. The process for completing this assignment should be as follows:

1. You already forked the Repository "sanroy/sp21-cs3060-hw" to a repository "yourId/sp21-cs3060-hw" under your username. If not, do it now.
2. Get a copy of hw7 folder in "sanroy/sp21-cs3060-hw" repository as a hw7 folder in your repository "yourId/sp21-cs3060-hw"
3. Complete the assignment, committing changes to git. Each task code should be in a separate file. As an example, task1.hs for Task 1.
4. Push all commits to your Gitlab repository
5. If you have done yet done so, sdd TA (username: prabeshpaudel for CS 3060) as a member of your Gitlab repository

Tasks:

1. **Task #1: (12 points)** Write a Haskell program that prints the string "Hello, NAME, you are hard working." where NAME is your name. Note that you need to compile your .hs file to create an executable, using commands like, "ghc -o myhello prog.hs" and then run the "myhello" executable. Submit a screenshot that shows the above activities (which carries 4 points). *Writing README carries 1 point.*

2. **Task #2: (12 points)** Write a function that accepts a list (lst) of integers as the parameter, and filters out a sub-list (of lst) which contains only the 3's multiples and 5's multiples in lst (if any). As an example, if lst is [34,2,12,25,15,32, 20], then the output is [12,25,15,20]. *Writing README carries 1 point.*

3. **Task #3: (14 points)** Write a function that accepts a list (lst) of integers as the parameter, and returns x where x is the number of perfect squares in the list lst. *Writing README carries 1 point.* As an example, if lst is [34,9,80,16,225,15,1000], then the output is 3 as there are three perfect squares (9, 16, and 225).

4. **Task #4: (12 points)** Write a Haskell function (named toDigitList) that accepts an integer as the parameter, and if the integer is a non-negative integer then it breaks it into its digits and the output is a list of digits. On the other hand, if the input is a negative integer, the function returns an empty list. Use the following examples to test your function:

   (a) toDigitList 1325 gives output [1,3,2,5] (b) toDigitList 0 gives output [0] (c) toDigitList (-32) gives output []

   Writing README carries 2 points.