15-110: Principles of Computing

HOMEWORK 02

Due: 14th September, 2020 at 11:59pm

- You need to submit your python file to Autolab.

- You must solve the tasks **individually**.

- There are 50 points.

1. (10 points) **Circle Area**

   The area of a circle of radius r is given by the following formula:

   $$A = \pi \times r^2$$

   Complete the function `circleArea(r)` that returns the area of a circle with radius r.

   Since there is no way to represent a real π in a machine with finite memory, you can use the approximation 3.141592.

2. (10 points) **Hotdog Purchase**

   Hot dogs are an American tradition. Each year, Americans eat up to 20 Billion hot dogs. A classic hot dog is made up of two components: A frank (the meat) and a bun. Yet, for reasons that mystify mankind, the franks are typically sold in packs of ten and the buns in packs of eight. And, of course, you must buy full packages.

   Write the function `hotdogPurchase(numHotdogs)` that takes as input the total number of hot dogs you want to make, and returns the number of packages of franks and the number of packages of buns you need to purchase. You may assume that the argument, `numHotdogs`, is a non-negative int and the function returns as ints the smallest number of packages of franks and buns that must be purchased.

   For example:

   ```
   hotdogPurchase (50) == (5, 7) # Meaning 5 packs of franks and 7 packs of buns.
   ```

3. (10 points) **Hotdog Excess**

   Write the function `hotdogExcess(numHotdogs)` that takes the total number of hot dogs you want to make (as a non-negative integer) and returns the number of excess franks and buns you will need to purchase.

   Hint: you may want to use some of the code from `hotDogPurchase`, which you just wrote!

   For example:

   ```
   hotdogExcess (50) == (0, 6)
   ```

4. (20 points) $\mathbf{K}^{th}$ **digit**

   Write the function `getKthDigit(n, k)` that takes a possibly-negative int n and a non-negative int k, and returns the kth digit of n, starting from 0, counting from the right. So:

```
getKthDigit(789, 0) == 9
getKthDigit(789, 1) == 8
getKthDigit(789, 2) == 7
getKthDigit(789, 3) == 0
getKthDigit(-789, 0) == 9
```

You may assume that the digits to the left of the last non-zero digit are all zeros.