CS3460: Data Structures   Spring 2018
Assignment 04: Fun with Hash Functions
Due Date: Monday, February 19; 23:00 hrs
Total Points: 20

This assignment will give practice with understanding hash functions. Specifically, you are given 4 hash functions, and the task is to play the role of a malicious adversary, who having acquired knowledge of these functions, is interested in producing keys that all hash to the same table index. For this problem, we will assume that n keys are hashed into a table of size n. If C is the largest possible key that can be hashed, then $\frac{C}{n}$ is the fraction of keys that can be hashed to the same index. If $\frac{C}{n} \geq n$, then according to the pigeonhole principle, we can find a set of n keys that all hash to the same table index. So we will assume $C = n^2$. Let k be the key being hashed. Then we have the following hash functions.

1. `hash1`$(k) = k \mod n$.

2. `hash2`$(k) = \left\lfloor \frac{n}{C}k \right\rfloor$.

3. `hash3`$(k) = (2917k + 101923) \mod n$.

4. `hash4`(k) is the first number generated from a psuedo-random number generator (the `Random` class in Java) with seed k.

Startup Code: All the hash functions are implemented in the file `HashFunctions.java`. It is not required to modify this file. Some startup code is provided in `Prog.java`. This program takes two arguments n and p from the command line, where n is the input size and p is the hash function number. It then calls the appropriate function (named `prog1` through `prog4`). Please complete these functions so that they output n keys that all hash to the same table index using the corresponding hash functions. A sample execution of the program is shown.

```
$ java Prog 100 2
```

When executing the above command, your code should print 100 keys (one per line) that all give the same hash output using the second hash function. For full credit, your program should run as fast as possible and work for $n \leq 1000$.

Submission: Please submit the files `Prog.java` and all other associated files as a single zip archive `hw4.zip` through ASULearn.

Input/ Output Instructions: For all programs, until and otherwise stated, we will be taking the input from standard input (`System.in`) and will be sending the output to standard output (`System.out`).

Notes on Coding: Please do not include user-defined packages in your code. Your code should run in the Unix/Linux machine using the commands `javac` and `java`.