CS 2341 – Assignment 4

Write a program that reads in a string from the command line and a dictionary of words (given below) and checks whether it is a "strong" password.
Assume "strong" means that it
(i) is at least 8 characters long,
(ii) is not a word in the dictionary,
(iii) is not a word in the dictionary followed by a digit 0-9 (e.g., hello5)

https://www.mit.edu/~ecprice/wordlist.10000

You will implement the dictionary by using hash tables. For each word, associated value is the line number (a: 1, aaron: 4, abandoned: 6).

1. Insert the words into hash table using separate chaining and use below hash functions. Assume M=1000 (fixed size).

First consider the following `hashCode()` implementation for `String`, which was used in early versions of Java:

```
public int hashCode() {
   int hash = 0;
   int skip = Math.max(1, length() / 8);
   for (int i = 0; i < length(); i += skip)
      hash = (hash * 37) + charAt(i);
   return hash;
}
```

Then, consider the current implementation:

```
public int hashCode() {
   int hash = 0;
   for (int i = 0; i < length(); i++)
      hash = (hash * 31) + charAt(i);
   return hash;
}
```

2. Insert the words into hash table using linear probing and report the results for above hash functions. Assume M=20000 (fixed size).

Input:
Password

Output:
Display if the password is strong.
Display the cost of the search (number of comparisons) for both hasCode() functions for separate chaining and linear probing.

Show the results for below passwords:
account8
accountability
9a$D#qW7!uX&Lv3zT
B@k45*W!c$Y7#zR9P
X$8vQ!mW#3Dz&Yr4K5

Grading rubric:
Read input file: 10
Implement strong password rules: 10
Hash table with separate chaining: 40*
Hash table with linear probing: 40*

* Each hashCode() function is 20 points.

Submission:

All results, concatenated into README file.
Source file(s)

All files will be uploaded to Github and you will submit the link to your GitHub repository through Canvas. If you work with another teammate include all team members' names and IDs in the README file.