Written Homework 3

Due: Wednesday, November 11, 9:00 PM

For this homework, use the data file `words.dat`. We want to investigate questions regarding the design of a good hash table for the data.

1. Consider the following hash function `hash()`:

```
int hash(char key[5])
{
  int i;
  long long x;

  x=0;
  for (i=0; i<4; i++) {
    x=x+key[i];
    x=x<<8;
  }
  x=x+key[4];
  return x%HASH_PRIME;
}
```

   When we put the 5757 five-letter words into a hash table, how many collisions occur for the following values for `HASH_PRIME`?

$$M_1 = 7000, \qquad M_3 = 12000, \qquad M_5 = 22000,$$
$$M_2 = 6997, \qquad M_4 = 11117, \qquad M_6 = 22307.$$

   Note that M_1, M_3, M_5 are composite numbers and M_2, M_4, M_6 are prime numbers. For this question, you probably need to write a computer program.

   For a hash prime M, the hash function h has a value between 0 and $M-1$. Let c_i be the number of words w, among 5757 words in `words.dat`, such that $h(w) = i$. Then the number of collisions is defined to be

$$\sum_{i=0}^{M-1} \max\{(c_i - 1), 0\}.$$

2. Supose that we use "chaining" for our hash table, using `hash()` and `HASH_PRIME` $= M_4$. After all the words are inserted in the hash table, if we search each and every word in `words.dat` "exactly once," how many key comparisons are made in total? Hint: you do not need to actually implement a hash table but only need to figure out the lengths of chains for each hash values.

3. (Optional) Supose that we use "open addressing" for our hash table, again, using `hash()` and `HASH_PRIME` $= M_4$. After all the words are inserted, if we search each and every word in `words.dat` "exactly once," how many key comparisons are made in total?