

September 06, 2004

Total time: $2\frac{1}{2}$ hours

You are given a list of English words. Write a program to break the list into groups such that the words in each group are anagrams of one another. For example, PAST, PATS, SPAT, and TAPS are all the allowed anagrams from the letters A, P, S and T. Similarly, all allowed anagrams from the letters E, E, L, S are EELS, ELSE, LEES, SEEL (but LESS is not such an anagram).

Part 1: Apply a symmetric hash function**(70%)**

Build a hash table based on a symmetric function of the letters of a word. A recommended hash function is given below. From the symmetry of the hash function it follows that if two words are anagrams of one another, they must collide. However, non-anagrams may also collide.

The following hash function H is recommended: First assign the value 0 to 'A', the value 1 to 'B', and so on. In short, define:

$$\text{val}(\langle \text{letter} \rangle) := \text{ASCII}(\langle \text{letter} \rangle) - \text{ASCII}('A').$$

Let $w = a_1 a_2 \dots a_l$ be a word of length l with each a_i being an upper-case Roman letter ('A' to 'Z'). Call

$$v_i := \text{val}(a_i)$$

for $i = 1, \dots, l$. Finally define

$$H(w) := \left[\sum_{i=1}^l (v_i^4 - 2v_i^3 + 3v_i^2 - 4v_i + 5) \right] \bmod 34543.$$

Report the following results:

- The number of non-empty buckets obtained from the given list.
- The maximum size of a bucket.
- A listing of the elements of a biggest bucket.

Part 2: Break lists into anagram groups**(30%)**

Each non-empty bucket obtained in Part 1 consists of one or more groups of anagrams. Isolate the groups in the biggest bucket found in Part 1. List all the anagram groups separately.

Here is an example. If the bucket contains the following list:

ANGEL ANGLE ASCOT CADDIE CAMPILY COAST COATS COSTA GLEAN TACOS

you should print the following anagram groups:

ANGEL ANGLE GLEAN
 ASCOT COAST COATS COSTA TACOS
 CADDIE
 CAMPILY