CO314: Systems and Network Programming Lab

Project I - Iteration I

Group members

- E10/049
- E10/170
- E/08/406

Description

we had to count the word frequencies of a given text of about 74000 words input ignoring the given stop words. There were about 7000 unique words.we had to give the Results to the standard output.

Our method

What we have done was,

1. Take the characters one by one using getc function from input
2. Make the input characters lowercase.
3. Add them to a temporary string.
4. If we found a punctuation mark then add '\0' to the end of the string and add the temporay string to the hash table.
5. When adding words to the hash table if the word we trying to add already in the hash table, then increase its frequency by one.
6. After get all the words, get the stop words from a file using fscanf.
7. While getting the stopwords, Delete the stop words from hash table.
8. Print the results to the standard output in the given format.

Data structure

We used hash table data structure to store words. To insert delete and add new nodes to the hash table we wrote functions separately.

- Access time was minimum for hashtable compare to other data structures. O(n) when no collisions. Size of the used hash table is 10000.

- Hash function is,

  hash_value=hash_value + data[i]*(383/(i+1)).

  (383 was just a random number. Nearly equal to (hashtable size)/26) which gives more weight to starting characters and less weight to ending characters. That makes the words spread over the hash table.
- Separate chaining method was used to avoid problems from collisions. Collisions were maximum 4- 5 for one bucket. So collisions not affected to the access speed of hashtable for the given number of words.

Benchmark

- It take about 1.8 -0.2 seconds to print the result for given input.
  - 74000 0.18s
  - 100,000 0.22s
  - 500,000 1.9s
  - 1,000,000 4.6s

May be the 10000 size of the hash table is not suitable for large number of words. But for a given input the time of 0.18 is acceptable.