Analysis of Algorithms 1

HW Report

HW 3

Ahmet Göktuğ SEVİNÇ

150140120

25.11.2017

a)

	Insertion	Lookup
Dictionary	0,619sec	2,192sec
List	0,064sec	47,796sec

b) In list implementation, we basically insert objects in the order shown in input file, in that case insertion becomes fast, but to search an object from list, we have to search all data until finding the correct object, so this process slows down our code's performence. For dictionary implementation, we insert object according to our key determined by hashing function and when this function leads us a non-empty row we are probing list with another function. This operations slows our insertion slightly but for looking up, now we have a clue about where our object could be, just like insertion, for look up we first look the place where our hashing function leads, and if the object is not there we probe list with probing function. So we don't have to look all data in our dictionary. That accelaretes the performance visibly.

c) Average number of collisions increase exponentially. Because as we insert more and more data, hash table fills and empty spaces decrase. Since our hashing function starts to lead us to non-empty places probing function takes role and after adding to dictionary many data, this probing function also starts to lead us to filled places and keeps searching for an available row for our object. So, average number of collisions increase quickly as we insert more data.

d) If we get same key for all objects, then we had to call our probing function for all data to find correct object, that is the worst case for looking up for dictionary. Corresponding time complexity for this situation is O(n). In worst case since keys are same, hashing function will give us same keys, if the data is not in that row correspondig to the key, we will call probing function and also this function will always lead us specific rows in our table, so it will start to work like a list.

Note: In my code there is a function called strtok_single. strtok function does not work on empty data in input file, this function fixes this situation. It is windows version of strsep function in linux. I took that code from stackoverflow. Link is in my code.