F28DA COURSEWORK I

Gd21



1. Specification of The program

The Program that we were asked to develop was a spellchecker implemented with an HashMap and a LinkedList.

The spellcheck takes all the words from a dictionary file and uploads them into the map structure.

The program then takes other words from a second file, and tries to find them in the dictionary, if it cannot find the words in the dictionary then it tries to compute corrections of the word, applying substitution, omission, insertion and swap to the letters in the word and comparing corrected words to the dictionary.

The main Program Takes place in the class SpellCheck.java.

The java class has 3 fields:

The StringHashCode is used as a parameter to create the maps,

Then the IMaps are required, one to contain the dictionary, one for the corrections.

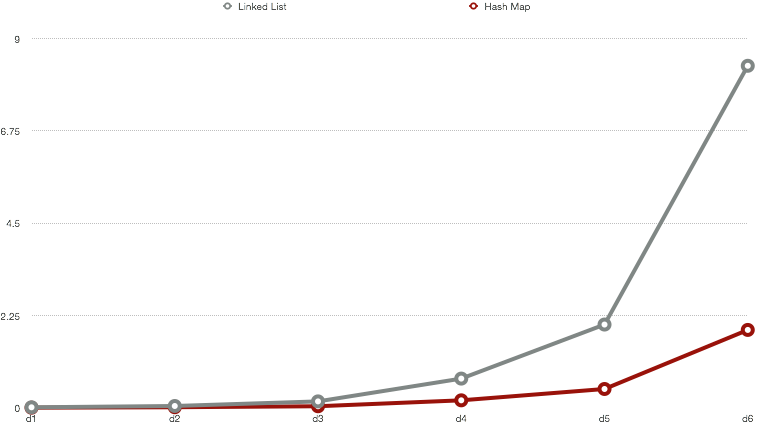
**public** SpellCheck(IMap Dictionary, IMap Corrections) {

SpellCheck.*M* = Dictionary;

SpellCheck.*S* = Corrections;

}

1. LinkedList Vs HashMap

I have created a class Compare.java to compare the linked list and hash Map implementations

The graph, shows for every dictionary file (d1…d6) the time in seconds spent to spellcheck by the hash Map (red) and the LinkedList (Grey) . It turns out that with the number of words in the dictionary going up the linked List become slower and slower compared to the Hash Map. That is because the linked list runs on quadratic time whereas the hash Map does run on linear time.

Giovanni D’amico