

**Gebze Technical University**  
**Department of Computer Engineering**  
**CSE 654 / 484**  
**Fall 2020**

**Homework 01**  
**Due date: Nov 16<sup>th</sup> 2020**

In this homework you will implement an enhanced version of edit distance algorithm. The classical edit distance algorithm efficiently finds the distance between given two words using dynamic programming. However, if we want to find the distance between one word against 1000 words, this becomes expensive. In this homework, we will use the following algorithm, which we have discussed during our lectures

- Instead of using 2D array, we will fill in 3D array as our major dynamic programming table. Previously,  $D(I,J)$  gave us the smallest edit distance between the two words up to letters I and J of these two words. Now,  $D(I,J,K)$  will give us the distance between a given word and the Kth word ( $K=1..1000$ ) up to letters I and J.
- To make our algorithm more efficient, will not fill all the 3D elements of  $D(I,J,K)$ . We will fill this table until we find most similar top 5 words out of 1000.
- First we will fill all costs of 1 in all the table elements for all 1000 words. Then costs of 2, then 3, etc. If we have any words finished matching, we will stop for that word. If the number of finished words reaches 5, then we stop filling in the 3D table and report results along with the edit distances.

You will need a dictionary of 1000 words for this homework, which you can get at [https://raw.githubusercontent.com/ncarkaci/TDKDictionaryCrawler/master/s%C3%B6zl%C3%BCkler/Zemberek\\_S%C3%B6zl%C3%BCk\\_Kelime\\_Listesi.txt](https://raw.githubusercontent.com/ncarkaci/TDKDictionaryCrawler/master/s%C3%B6zl%C3%BCkler/Zemberek_S%C3%B6zl%C3%BCk_Kelime_Listesi.txt)

Your program will work like following

- Load the words from a file, the file name will be given to your program. This file will contain 1000 words. Make sure to have at least 5 of these files with random words
- Enter the main word
- Report the results of top 5 words with their edit distances
- Also report the occupied percentage of the dynamic programming table. Lower the percentage, more efficient your algorithm.
- You may assume that, the longest word is 15 characters long. You may ignore rest of the word if the word is longer.

Submit your homework to the moodle page, you will demo your homework after the submission.