

Create a better diff tool: this tool will compare two text documents and identify the minimum sequence of transformations which can be applied to the first document in order to re-create the second document.

Example:

Draft 1:

An exciting day dawned.

It was a sunny day. The big fat dog jumped over the lazy fox. And then night fell.

Another day dawned.

Draft 2:

It was a sunny day. The big fat frog jumped over the lazy fox. And then night fell.

An exciting day dawned.

Another day dawned.

MINIMUM TRANSFORMATION SEQUENCE:

1. Replace word "dog" with word "frog"
2. Transpose sentence 1, sentence 2

Thus, it should be able to detect transformations, i.e., insertion, deletion, substitution, of blocks of text (eg. paragraphs, sentences, words). Note that transformations may be nested: for example, a paragraph at the beginning of the first draft of a document may have been moved to the end of the document in the second draft, and in addition, the paragraph may have had edits made within it in the second draft.

There are three types of transformations:

1. Deletion:

the quick brown fox

the brown fox

## 2. Addition:

the brown fox

the quick brown fox

## 3. Transposition:

the quick brown fox

the brown quick fox

There are two components to this project:

Program 1, Diff Tool: Find the minimum sequence of transformations

The diff tool will take two command line arguments, which are the names of two documents. The program will print to a file or the standard output the minimum sequence of transformations that turn one document into the other. You can use any programming language (Python preferred).

For example:

```
find_transformations.py file1.txt file2.txt
```

Output:

```
file1_to_file2.transformations (the formatting of this file is up to you)
```

Program 2, Transformation Utility: Verify Correctness

The transformation utility will take one document and the text file produced by the diff tool and will perform the changes to turn one document into the other. The purpose of this utility is to verify that the sequence of transformations produced by the first program can correctly transform the first document into the second.

For example:

```
execute_transformations.py file1.txt file1_to_file2.transformations
```

Output:

file3.txt

Then, if the transformations were identified and applied correctly, the standard UNIX diff tool should show that they are identical.

To Start:

Fork this GitHub repository. This repository contains this document, as well as some example pairs of documents, each pair consisting of an original version and a scrambled version.

<https://github.com/chrisbdf/better-diff-tool>