Design and Development of Plagiarism Detection Software in C++

A plagiarism detection software is a **Software** system that takes a submitted text as input, and compares the text against a set of publicly available and privately held documents, resulting in a similarity report. The similarity report includes marking of similar or identical text, hyperlinks or other references to sources that match it, and an overall similarity report. Such software is developed specifically to identify cases where someone's work is presented totally or partially without giving credit to its owner and without applying proper citation practices. The idea behind this project is to develop a simple program that can check plagiarism of a simple text file by comparing it with 5 different source files.

Problem specification:

Your designed program should meet the following conditions and rules.

1) Should take the name of the Test file and 5 source files from the user as input.
2) Match the test file with source files 1 by 1 and keeps a track of the number of words and phrases/clauses found similar.
3) It should keep track of and print similar words in the following fashion

Total Number of similar words found=143
33 from file 1
10 from file 2
50 from file 3
25 from file 4
25 from file 5

Note: The given word count is only for example, the actual word count will depend upon the test and source files.

4) Should generate a Similarity index report both at the end of the file and on the console in the following formatting.

Similarity Index= __%
Source 1 = __%
Source 2 = __%
Source 3 = __%
Source 4 = __%
Source 5 = __%

Note: Count the number of words that are shared between the test and source files. Count the total number of words in both files (shared and un-shared). Divide the number of shared words by the total number of words. Multiply the number you found by 100.0.

$$\text{Similarity Index} = \frac{\text{Number of Shared Words}}{\text{Total Number of Words in all files}} X100$$

Use the same formula to generate a similarity index from every source file i.e. from the Test file and 5 source files.

Considering the above example suppose the test file contains 200 words and total number of words of 5 source files is 200, 300, 150, 500, and 300 respectively. Then Similarity index will be calculated as follows:

$$\text{Total Similarity Index} = \frac{\text{Number of Shared Words}}{\text{Total Number of Words in all files}} X100$$

$$= \frac{143}{200+300+150+500+300+200} X100 = 8.6\%$$

$$\text{From Source 1} = \frac{\text{Number of Shared Words between } test\ and\ source\ file\ 1}{\text{Total Number of Words in test and source file 1}} X100$$

$$= \frac{33}{200+200} X100 = 8.25\%$$

$$\text{From Source 2} = \frac{\text{Number of Shared Words between } test\ and\ source\ file\ 2}{\text{Total Number of Words in test and source file 2}} X100$$

$$= \frac{10}{200+300} X100 = 2\%$$

You can similarly calculate the similarity index from the remaining files too.

5) Total number of phrases found similar.
Then it should print the similar phrases found at the end of the test file as follows:
Total Number of Similar Phrases= 32

Similar Phrases/Clauses	Source File
a) In the woods	1
b) Living in Pakistan	5
c) The amazing Spiderman	3

Note: Count the group of 2 or more words a phrase/Clause.

Final Project Report

The final project report should be formatted as a two-column, 4-6 page IEEE conference paper, with appropriate references in IEEE format. The template file can be found here: https://www.ieee.org/content/dam/ieee-org/ieee/web/org/conferences/conference-template-a4.docx. The emphasis should be on analysis, interpretation, and validation of the choice of method(s) used and any underlying assumptions with critical discussion on conclusions. The report should have the following four mandatory sections:

• Introduction

• Methods

• Results

• Discussion and Summary

You may also use Google Scholar to see sample of reports written in IEEE Conference Paper format.

General Instructions:

You must abide by the following general instructions.

1) You can make a group of at most 5 people.
2) You must include comments in your program.
3) Submission Deadline is the last week before ESEs in your respective lab timeslots.
4) Presentation and individual viva will be taken along with IEEE formatted report and working code.
5) Copied report or code will be marked 0

Assessment Rubric:

Sr. No.	Attribute	[1-3] Below Expectations	[4-7] Meeting Expectations	[8-10] Exceeding Expectations
1	**Word Count**	Code is unable to count the total number of words.	Code only works on samples provided by the programmer.	Code works on every sample provided.
2	**Phrase/Clause Count**	Code is unable to count and print the total number of phrases/clauses.	Code only works on samples provided by the programmer i.e. it counts and prints all the phrases found similar.	Code works on every sample provided i.e. it counts and prints all the phrases found similar.
3	**Program output formations**	Program output was not formatted properly.	Program output was well-formatted and followed the instructions provided.	Program output was well-formatted and followed the instructions provided using manipulators.
4	**Comments**	Comments were not included in the program	Comments were included in the program.	Comments were included in the program including the smallest of the details possible.
5	**C++ Libraries and Data Structures**	Produces programming solutions that use existing libraries and built in functions only.	Organizes programming solutions that incorporate appropriate data structures existing as well as programmer-defined functions.	Organizes programming solutions that incorporate appropriate data structures existing as well as programmer-defined functions and classes.
6	**Report**	The report is unstructured. The majority of the sections are missing. Figures are not properly formatted	The report is unstructured. The majority of the sections are missing. Figures are not properly formatted	All sections are included and properly formatted as per the given format. Figures and tables are properly formatted, have captions, and are referred in the text.
7	**Viva**	The student knows about the program working but was unable to link the working with the program structure.	The student knows about the program working and was able to link the working with the program structure.	The student knows the proper use of C++ language i.e. not only for this particular project.