# Test Plan for “DIFF C++ Program”

1. Defined specification
2. Logistics
3. Unit Testing
4. Test Cases
5. Goals

## DEFINED SPECIFICATION

* Development a DIFF C++ Program which compares two input text files and shows the difference.
* Specification using engineering sense:
  + if numbers of lines differ: program will show this in message and end program.
  + if numbers of lines are matched but numbers of words differ: program will print the line of both texts
  + if numbers of lines are matched and numbers of words are matched but word differ: program will print the word of both texts.
  + if there is no difference: program will show this in message and end program.
* Special Modes using engineering sense:
  1. Ignore case mode: ignore capital and small letters differences.
  2. Ignore blank lines mode: ignore any blank lines differences.
  3. Not ignore spaces mode: not ignore numbers of spaces differences.
  4. Ignore lines mode: ignore number of lines differences when the words are identical

## LOGISTICS

* Machine: 1 Laptop
* Software: Codeblocks, notepad++

## UNIT TESTING

* Main Units:
  + Compare files word by word: tested using Ignore lines mode by input same inputs but in different numbers of lines or input different words
  + Count numbers of lines: tested by input multi lines texts and count numbers of lines
  + Compare files line by line: tested by input 2 texts with identical numbers of lines but with different numbers of words or input different words

## TEST CASES

* Case 1
  + Description: input 2 identical texts
  + Expected Output: message “both texts are identical”
* Case 2
  + Description: input 2 texts with different numbers of lines
  + Expected Output: message “both texts are identical”
* Case 3
  + Description: input 2 texts with identical numbers of lines but with different numbers of words
  + Expected Output: message “both texts are identical”
* Case 4
  + Description: input 2 texts with identical numbers of lines and identical numbers of words but with different words
  + Expected Output: message “both texts are identical”
* Case 5
  + Description: input 2 texts with different case
  + Expected Output: message “both texts are identical”
* Case 6
  + Description: input 2 texts with different case and use ignore case mode
  + Expected Output: message “both texts are identical”
* Case 7
  + Description: input 2 texts with different blank lines numbers and use ignore blank lines mode
  + Expected Output: message “both texts are identical”
* Case 8
  + Description: input 2 texts with different spaces numbers
  + Expected Output: message “both texts are identical”
* Case 9
  + Description: input 2 texts with different spaces numbers and use not ignore spaces mode
  + Expected Output: message “both texts are identical”
* Case 10
  + Description: input 2 texts with different numbers of lines but the same words and use ignore lines mode
  + Expected Output: message “both texts are identical”

## GOALS

* 100 % of required test passed.