



Total Points: 100

## COP4342 – UNIX Tools

### Assignment #2: Custom Simplified Diff

Instructor: Dr. Robert Van Engelen Teaching Assistant: Joydeep Das

Spring 2018

**Description:** The diff command line utility on UNIX and UNIX-based platforms compares two text files line by line and yields the difference if the files are not identical. The syntax is as follows :-

**diff** [-options] file1 file2

The output of this utility typically outlines the minimal changes, in the form of three different type of instructions, that are to be made in file1 to make it the same as file2. For example, if the contents of the files are as shown :-

#### *file1*

*This is line one of file1  
welcome to cop4342  
code should execute on linprog  
This is a sample test case  
Florida State, Tallahassee  
GNU is Not UNIX  
Linux is based on UNIX*

#### *file2*

*This is line one of file2  
welcome to UNIX tools  
code should execute on linprog  
12345678987654321  
This is a sample test case*

The output of '*diff file1 file2*' would be :-

```
1,2c1,2
< This is line one of file1
< welcome to cop4342
---
> This is line one of file2
> welcome to UNIX tools
3a4
> 12345678987654321
5,7d5
< Florida State, Tallahassee
< GNU is Not UNIX
< Linux is based on UNIX
```

#### **Key Things to note here:**

[1]. '<' indicates the lines of file1 and '>' indicates the lines of file2.

[2]. 'c', 'd', 'a' refer to change, delete and add respectively. These are the instructions that diff outlines. Numbers to the left and right of these instructions point to a line or range of lines from file1 and file2 respectively.

[3]. '3a4' here means add line 4 of file2 after line 3 of file1. '5,7d5' means delete lines 5 through 7 from file1 to get a match up to line 5 of file2.

Notice that even if identical lines do not appear on corresponding line numbers of file1 and file2, diff proposes adjustments to their position in file1 through minimal changes, to match that of file2.

**Assignment:** For this assignment, a simplified version of the diff utility is to be implemented. This custom utility (which we will name *csdiff*) should function by only comparing the corresponding lines of the two text files provided as arguments (that is, without having to scan the files ahead to search and match lines). Rather than reporting a single line each time there is a difference, you should report deletions, additions, and changes grouped together like the diff utility.

Hence, if executed with the above files, output of *'./csdiff file1 file2'* should look like:-

```
1,2c1,2
< This is line one of file1
< welcome to cop4342
---
> This is line one of file2
> welcome to UNIX tools
4,5c4,5
< This is a sample test case
< Florida State, Tallahassee
---
> 12345678987654321
> This is a sample test case
6,7d5
< GNU is Not UNIX
< Linux is based on UNIX
```

**Assumptions and Restrictions:**

[1]. There will be at most 100 characters per line and at most 50 lines in the test case files of this assignment.

[2]. All file operations should be implemented with the help of the relevant UNIX API calls.

If the order of the files is reversed and provided to the custom *csdiff* utility, the change instructions remain similar, only the last instruction becomes:

```
5a6,7
> GNU is Not UNIX
> Linux is based on UNIX
```

It should be clear that for this custom implementation the addition and deletion instructions are only relevant when one of the files contains more lines than the other.

**Command line options:** The following options for the custom diff utility are to be implemented as described :-

- d** The presence of this option would require the implementation to delete the least recently modified file in case the two provided files are identical.
- o file** The presence of this option implies that the output of the utility is to be written/redirected to the file instead of standard output.

**Exit Code:** The *csdiff* utility should exit with `EXIT_SUCCESS` if the files are identical and `EXIT_FAILURE` otherwise (files could not be opened or they differ).

**Submission Requirements:** A tar file named 'yourCSusername\_asg2.tar' containing the following files:-

[1]. The C or C++ implementation of the above described *custom diff* utility *csdiff*. Do not include object files or binaries.

**Grading Policy:** Submitted code must compile with the '-Wall -ansi -pedantic' flags without any warning messages to conform to the ANSI ISO standard. Code that does not compile or generates errors (Segmentation Fault etc.) on execution will receive zero grade points. Points distribution :-

[1]. Code Readability	(20 points)
[2]. Test Cases	(80 points)

Individual parts of the implementation will not be graded separately for correctness. There will be several cases to test the implementation logic as a whole. Also, keep in mind that your code will be tested on *linprog*. Students should test their code thoroughly on the *linprog* server before submission.

**Miscellaneous:**

Honor code policy will be strictly enforced. Write the code by yourself and protect your submission.

**Key Concepts:** *System Calls, File Descriptor, File Meta-Data, File Manipulation, String Operations, Command Options Parsing.*

**UNIX API Calls:** *open(), read(), write(), stat(), dup2(), unlink().*