

Programming Project 4

Due: Thursday, 9/19/13 at 11:59 pm

Find and Replace

For this lab, you are to write a C program that will perform a "find and replace" of characters in a string.

Input will be given in the following manner:

```
#target1#replacement1#
originalString1.1
originalString1.2
...
originalString1.N
#target2#replacement2#
originalString2.1
originalString2.2
...
originalString2.M
...
###
```

When the first character on the line is the # symbol, you have a new **target-replacement** pair. For the original strings that follow, you are to find all occurrences of the target substring in the original string and replace the target with the replacement substring. Note that the lengths of the target and the replacement do not need to match. The target substring must be of length of 1 or greater, but the replacement substring can be of zero length. If a zero length target is given, print an error message, ignore the current line and continue processing the input.

Your program is to process each "original string" line using the last valid target-replacement pair. If the input does not begin with a target-replacement pair, your program should do zero substitutions until a valid target-replacement pair is given.

Your program should quit when it sees the input of ###

The input is to come from standard input. This file may be given if redirection is used. An example input file can be found at: [proj4.txt](#)

The output of the program is to be given to standard output. When a new target-replacement pair is read in by your program, you are to print out some message showing the new values of both the target and replacement substrings. When an "original string" is read in by your program, your program is to output (in this order!):

1. the original string,
2. the total number of substitutions made and
3. the final string.

This information is to be formatted in some readable manner.

Any error messages are to be written to standard error. About the only real error is an invalid target/replacement string (i.e. a target string of length zero).

You may not assume a maximum length of the strings to earn full points for this assignment.

Your program must read in each line of input completely before processing it.

The [strncmp \(\)](#) library function may be useful for this program depending on your approach.

Program Submission

You are to submit the programs for this lab via the Assignments Page in [Blackboard](#).

To help the TA, name your file with your net-id and the assignment name, like:

- ptroy1LabX.c

Examples for the Project

Assume the input file contains the following:

```
#h#j#
hello
happy birthday
#ra#ar#
radar
are you ready
pirate radio rating
##This is an Error – Target String has length of zero#
this line should still use the target string of ra
###
```

The above begins with one target/replacement pair followed by two original strings, then has a second target/replacement pair followed by 3 strings. The first target/replacement pair tells your program to replace all occurrences of h in an original string with a j. Thus the first original string of **hello** becomes **jello** and the second original string of **happy birthday** becomes **jappy birtjday**. The second target/replacement pair tells your program to replace all occurrences of ra with ar. Thus the original string of **radar** becomes **ardar**, the original string of **are you ready** becomes **are you ready** (no replacements are done since the target string does not exist), and the original string of **pirate radio rating** becomes **pirate ardio arting**.