Kristopher Swartzbaugh
CWID: 890939184
CPSC – 535
03/05/2020

Project 1: Rewriting Greeting Cards

Project Contents:
README – File containing brief project and code description
adv-alg-replace-strings.cpp - File Containing all code and the algorithm
replaceStringsInput.txt - Input text file to be read by the algorithm containing input strings and replacement pairs
Report.pdf – This file. Report containing summary, psuedocode, and screenshots.

Summary:

adv-alg-replace-strings.cpp will accomplish the following:

1. scan the contents of replaceStringsInput.txt
    1. Convert content to usable data structures
2. Use brute force algorithm to scan the input strings for the appropriate text from the replacement pairs, and replace that substring with the respective replacement substring
3. Print the new modified strings to the screen

Input file should be a .txt file of the following format:
My dear Anna, let me congratulate you on the beautiful car that you purchased today. It looks very posh. I hope you got a good deal. Cars are expensive but much needed. Best regards, Naomi.
{{"Anna", "Jovi and Victor"} , { "car", "house"}, {"today", "last week"}, {"posh", "well built"}, {"Cars", "Houses"}};

The first line being the string to be modified, without quotation marks, the following line to be the set of replacement pairs. Following lines can have the same format alternating between string to be modified and replacement pairs as the algorithm can take multiple inputs at the same time.

Psuedocode:

Function getStrings():
    open input file
    read file line by line
        even lines, convert to string to be stored as set of input strings (mystring)
        odd lines, parsed to appropriately convert to sets of input strings (my replace)
    call subStrings1(string mystring, vector<string> & my replace) with the gathered data
    output vector contianing all updated strings

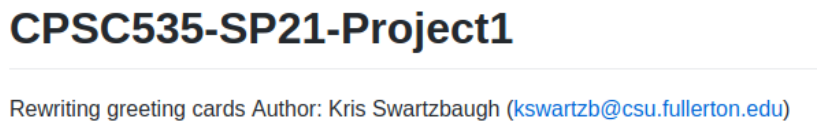
Function replacesubs1(string mystring, vector<string> & my replace)
    cycles through series of replacement pairs, calling replacesubs for each set
    output is updated string

Function replacesubs(string myString, string Find, string replace)
    brute force algorithm
    iterates through myString looking for first character in string Find
        if found checks to see if following characters match rest of Find
            if they do not match, continue iterating through the string
            If found, replace substring of mystring containing Find with replace
        continues searching for any subsequent matches of Find
    returns updated string

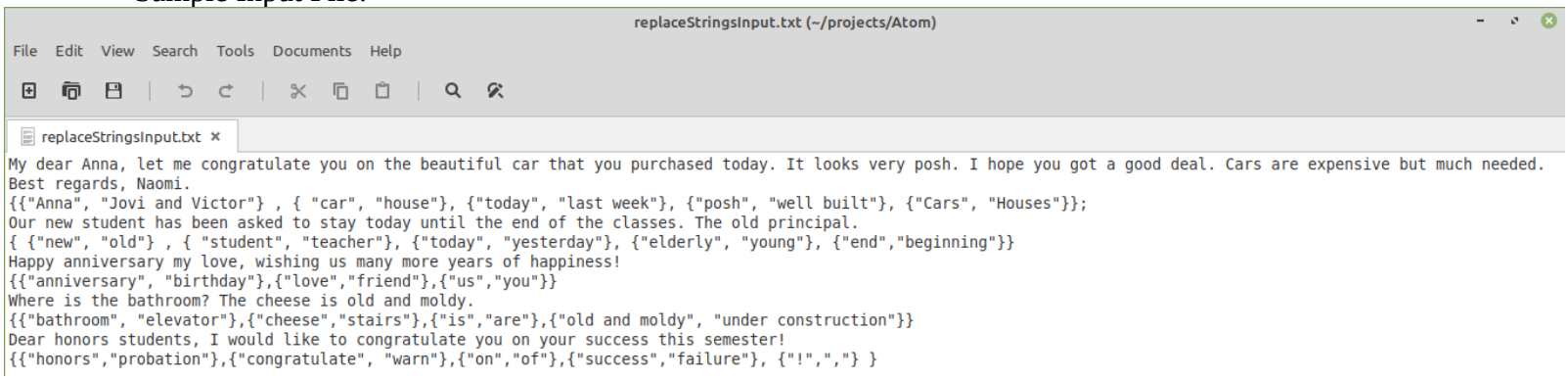
Function printVector(vector<string> &myVect):
    iterate through vector containing all the final modified strings
    print each string to the screen

Screenshots:

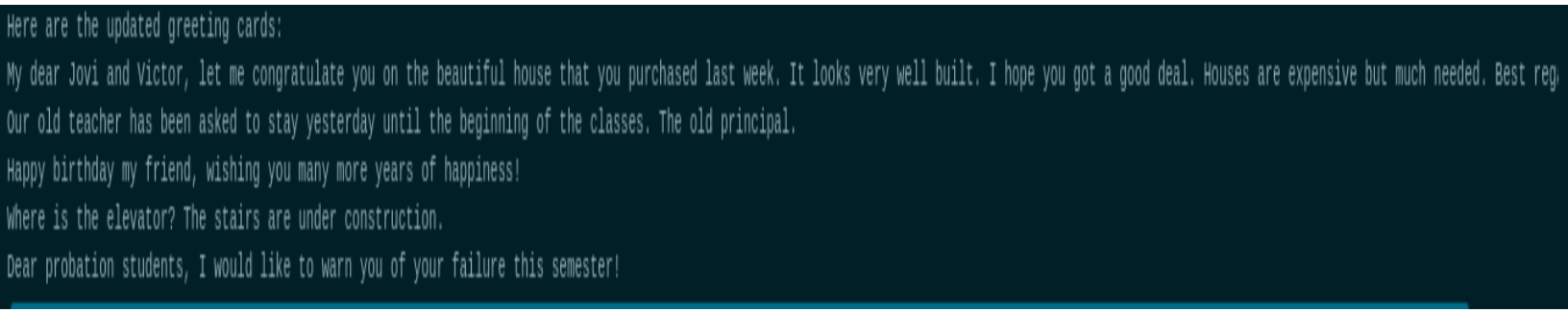
Group Member:



Sample Input File:



Sample Output:



Since this program was built to take an input final containing multiple strings and replacement pairs in a single file no additional screenshots are needed. The above screenshot contains 4 examples of input and output strings