**Problem Description**

For this problem, given a text file, I was to remove the control characters using Java, Python, and Perl. This was a bit tricky, because I had to ensure that when I read in the files in each language, it would preserve all the control characters. To verify this, after I read the file into a string in each program, I wrote it directly back to a new file and checked it in Vim on the lab computers (I removed the code that did this for my final submission). All my programs preserved the control characters correctly, so I proceeded to elimination. To do this, I used a system of flags. For Perl and Python, the file is read one character at a time. That character is stored in a variable, and is converted to its ASCII value, then it’s compared to the ASCII value of the ctrl-b and ctrl-c characters. If we meet a ctrl-c, we know not to print that and anything after it until we hit a ctrl-b. So, the flag is set to 0. Once we hit a ctrl-b, we can set the flag to 1 again, but we need to use a next/continue to make sure that we skip to the next iteration of the loop, so we don’t print the ctrl-b. Finally, the loop reaches the last conditional, and will print the character to the file if the print flag is 1. The only difference with Java is the buffered reader, which automatically reads the character as an integer. This means we don’t have to cast the character to an integer, otherwise it functions the same as the other two.

**Java Code and Output**

/\*Joseph Camacho-Terrazas

\*10/26/2020

\*Input: The backup text file

\*Output: Resulting string after eliminating control characters

\*Preconditions: The file must exist and filename must be valid

\*Postconditions: Print the resulting string after eliminating all control characters

\*/

import java.util.\*;

import java.io.\*;

public class ControlChar {

    public static void main(String[] args) {

        try {

            //Declare the input and output files

            File input = new File("control-char.txt");

            File output = new File("javaoutput.txt");

            //Create a file reader, file writer, and buffered reader

            FileReader fr = new FileReader(input);

            FileWriter fw = new FileWriter(output);

            BufferedReader br = new BufferedReader(fr);

            //c stores value from buffered reader

            int c = 0;

            //Print flag allows us to print when it's 1

            int printFlag = 1;

            //Loop through the file using buffered reader

            while ((c = br.read()) != -1) {

                //Buffered reader gives us ints, so compare ascii values

                //If c is ctrl-c, printFlag is 0

                if(c == 3) {printFlag = 0;}

                //If c is ctrl-b, printFlag is 1 and skip to next iteration

                if(c == 2) {printFlag = 1; continue;}

                //If printFlag is 1, write c to the file

                if(printFlag == 1) {

                    fw.write(c);

                }

            }

            //close all open readers and writers

            fr.close();

            fw.close();

            br.close();

        }

        catch (IOException e) {

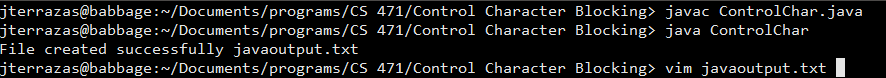
            //Catch any IO Exceptions

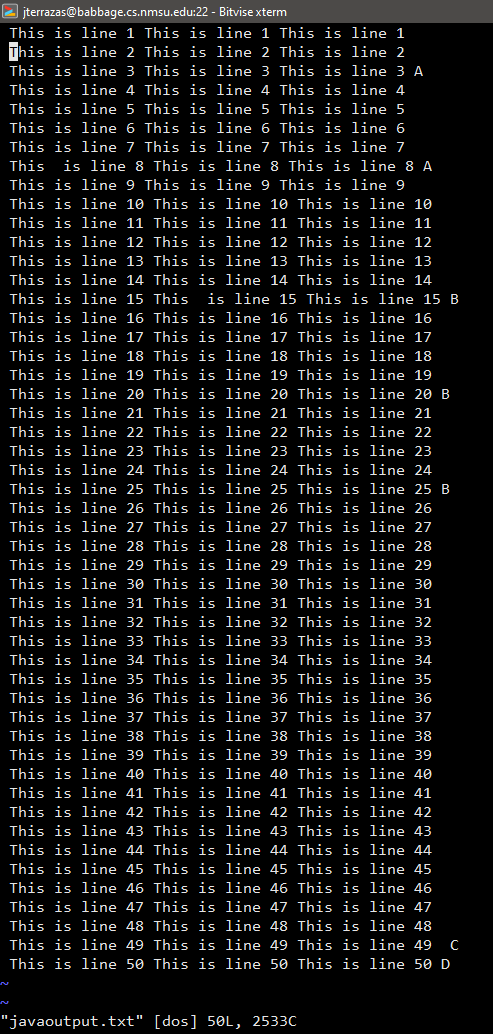
            System.out.println("There was an error" + e);

        }

    }

}





**Perl Code and Output**

#Joseph Camacho-Terrazas

#10/26/2020

#Input: The backup text file

#Output: Resulting string after eliminating control characters

#Preconditions: The file must exist and filename must be valid

#Postconditions: Print the resulting string after eliminating all control characters

#!/usr/bin/perl

use warnings;

use strict ;

#Declare input and output files

my $inputFile = "control-char.txt";

my $outputFile = "perloutput.txt";

#finalString stores the cleaned string

my $finalString = "";

#Read and char delcared for the reader

my $read;

my $char;

#printFlag will let us print if it's 1

my $printFlag = 1;

#Open the input file using file handler

open FILE, '<', $inputFile or die "Can't open file! $!";

#Read through the file one char at a time

while ($read = read FILE, $char, 1) {

    #Convert char to ascii value and compare to ctrl values

    #If char is ctrl-c printFlag is 0

    if(ord($char) == 3) {

        $printFlag = 0;

    }

    #If char is ctrl-b printFlag is 1

    if(ord($char) == 2) {

        $printFlag = 1;

    }

    #Skip to next iteration on a ctrl-b

    next if(ord($char) == 2);

    #If printFlag is one we add the char to the final string

    if($printFlag == 1) {

        $finalString .= $char;

    }

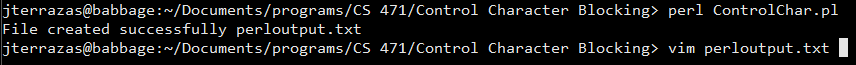
}

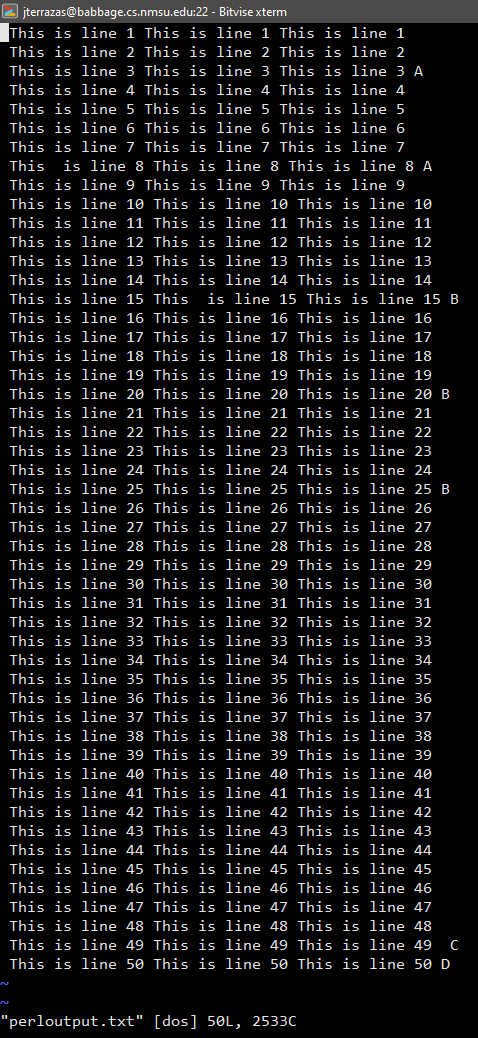
#Open the output file and print the final output string to it

open FILE, ">", $outputFile or die "Can't open the file! $!";

print FILE $finalString;

print "File created successfully $outputFile\n";





**Python Code and Output**

#Joseph Camacho-Terrazas

#10/26/2020

#Input: The backup text file

#Output: Resulting string after eliminating control characters

#Preconditions: The file must exist and filename must be valid

#Postconditions: Print the resulting string after eliminating all control characters

import string

#create a flag that will allows us to print

printFlag = 1

#Open the input and output files

f1 = open("control-char.txt")

f2 = open("pythonoutput.txt", "w")

while True:

    #Read one character of the file into c

    c = f1.read(1)

    #Break out of the loop at the end of the file

    if not c:

        break

    #Convert the chars to ascii values, and compare to control char value

    #If current char is ctrl-c, print flag is 0

    if ord(c) == 3:

        printFlag = 0

    #If current char is ctrl-b, print flag is 1 and skip to next iteration

    if ord(c) == 2:

        printFlag = 1

        continue

    #If print flag is 1, then write it to the output file

    if printFlag == 1:

        f2.write(c)

#User confirmation about the output file creation

print ("File created successfully ")

print (f2.name)

#Close all open files

f1.close()

f2.close()

