Program 7: File Conversion – Remove Control Blocking data from Backups

**Problem Description:**

* We are given a text file which contains added on corrupt data. This data was added on at the time that the file was backed up using some backup program. The corrupt data that was added on is the control blocking characters ‘CONTROL C’ and ‘CONTROL B’, and on top of this, carriage returns were added into the file. We were then to write programs in Perl, Python, and Java which remove these values from the file.
* In all three language implementations I read the file and used a flag to determine whether to print the current looked at character to the output file or not. I set the flag initially to true and then once I reached a CTRL C I set to false in which I stopped printing the characters until I reached a CTRL B where I set the flag to true and allowed printing to resume. This method proved to be successful in all three languages.

**Perl Code & Output:**

#!/usr/bin/perl

# Name: Tony Maldonado

# Date: October 31, 2020

# Input: A txt file which has corrupt data in it.

# Output: A new txt file with the corrupt data removed.

# Preconditions: The txt file should exist and be named 'control-char.txt'.

# Postconditions: The file is wanted with only printable characters.

# e.g., no control blocking chars and no carriage return

use strict;

use warnings;

# Variable declarations:

# These are the input and output files

my $inputFile = "control-char.txt";

my $outputFile = "newPerlFile.txt";

# String will store the final string

my $fileString = "";

# Flag to determine whether we should print

my $flag = 1;

# Read and char are for the file reader

my $read;

my $char;

# Open the file in read mode

open FH, '<', $inputFile or die "No such file, can't open! $!";

# Use an integer flag to determine what gets printed to the new file.

# Initially flag is true, so it starts off printing fine. Once it

# encounters a CTRL C it will set the flag to False and not print until

# we encounter a CTRL B and set the flag to true afterwords

while ($read = read FH, $char, 1) {

# CTRL C sets flag to 0 and doesn't print

if (ord($char) == 3) {

$flag = 0;

}

# CTRL B sets flag to 1

if (ord($char) == 2) {

$flag = 1;

}

# Skip to next iteration

next if (ord($char) == 2);

if ($flag ==1){

$fileString .= $char;

}

}

# Create the file for the output and print to it

open FH, '>', $outputFile or die "No such file, can't open! $!";

print FH $fileString;

print "New file created successfully: $outputFile\n";



A picture containing text

Description automatically generated

**Java Code & Output:**

/\*

\* Name: Tony Maldonado

\* Date: November 01, 2020

\* Input: A txt file which has corrupt data in it

\* Output: A new txt file with the corrupt data removed.

\* Preconditions: The text file should exist and be named 'control-char.txt'.

\* Postconditions: The file is wanted with everything in between CTRL C and

\* CTRL B removed. All other characters stay in place.

\*/

/\* Sources:

\* https://stackoverflow.com/questions/811851/how-do-i-read-input-character-by-character-in-java

\* https://www.baeldung.com/java-write-to-file

\* https://stackoverflow.com/questions/5868369/how-can-i-read-a-large-text-file-line-by-line-using-java

\*/

import java.io.\*;

import java.util.\*;

import java.nio.file.\*;

public class controlRemoval {

public static void main (String [] args) throws IOException {

// First read the input file stream

FileInputStream fstream = new FileInputStream("control-char.txt");

BufferedReader br = new BufferedReader(new InputStreamReader(fstream));

// Then create the new output file

BufferedWriter writer = new BufferedWriter(new FileWriter("newJavaFile.txt"));

// Use an integer flag to determine what gets printed to the new file.

// Initially flag is true, so it starts off printing fine. Once it

// encounters a CTRL C it will set the flag to False and not print until

// we encounter a CTRL B and set the flag to true afterwords

int flag = 1;

int r;

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

char ch = (char) r;

if (ch == 0x3)

flag = 0;

if (ch == 0x2){

flag = 1;

continue;

}

if (flag == 1)

writer.write(ch);

}

System.out.println("New file successfully created: newJavaFile.txt");

/// Close both files

fstream.close();

writer.close();

}

}

**Text

Description automatically generated**

**A picture containing text

Description automatically generated**

**Python Code & Output:**

# Name: Tony Maldonado

# Date: October 31, 2020

# Input: A txt file which has corrupt data in it.

# Output: A new txt file with the corrupt data removed.

# Precondition: The txt file should exist and be named 'control-char.txt'.

# Postcondition: The file is wanted with everything in between CTRL C and

# CTRL B removed. All other characters stay in place.

import string

# Open the txt file and assign it to a string

file1 = open("control-char.txt")

# New file for writing the modified contents to

file2 = open("newPythonFile.txt", "w")

# Use a boolean flag to determine what gets printed to the new file.

# Initially flag is true, so it starts off printing fine. Once it

# encounters a CTRL C it will set the flag to False and not print until

# we encounter a CTRL B and set the flag to true afterwords

flag = True

for lines in file1:

for char in lines:

if char == chr(3):

flag = False

if char == chr(2):

flag = True

continue

if flag == True:

file2.write(char)

# New modified file created successfully

print("New file created successfully: ")

print(file2.name)

# Close both files

file1.close()

file2.close()



A picture containing text

Description automatically generated