

COMP 1130 Principles of Programming II
Project 3 – File Conversion
Maximum Possible Points: 20

Due Date:

The source code and the program execution screenshot need to be submitted by October 18th.

Objectives:

To gain experience with arrays.

To gain experience with passing I/O streams to functions.

To gain experience with pointers and dynamic memory allocation.

To gain experience with C-strings.

Overview:

Internet browsers usually allow you to save an html file as a text file. You will have to write a utility program that will read an html file and convert it into a text file. A sample html file can be found at

<http://www2.fairmontstate.edu/users/mhossain/test.html>. If you open this file in a browser and click on View ->Source, you will see all the html tags used in the file. All the html tags are written inside the tag markers < >. Also, HTML uses the symbol
 for starting a new line. The corresponding text file can be found at <http://www2.fairmontstate.edu/users/mhossain/test.txt>. If you open this file in a browser and click on View ->Source, you will not see any of the html tags.

Once you download the html file, you will have to open it in text mode, store the content in an array, remove all the newline characters ('\n'), replace
 by the newline character, remove everything that appears between the tag markers < and >, and display the text content or write the results back to a .txt file. This converter will not be a very smart one, e.g., an html file can have more sophisticated tags. You will not use the string class in your program, i.e., **YOU WILL NOT include the <string> header file.**

Your program should display the following menu:

HTML/TXT Converter

1) Display TXT to Console

2) Write TXT to file

Your program should have the following functions in addition to main:

The **readHtmlFile** function should accept the html filename (an array of characters) and another array of characters for storing the html file content. It should open the file and read the html file contents into the second array. The function should return an integer that states how large the file is, in number of characters.

The **convertHtmlFile** function should accept an array of characters for holding the html contents, another array of characters for holding the txt contents, and the number of characters in the html file. This function should be responsible for reading the characters from the first string and store the characters without html tags in the second array. This function should return the number of characters in the converted txt file.

The **outputTxtFile** function should accept an output stream, an array of characters holding the .txt contents, and the number of characters in the text file. It should write the txt file content either to the console or to an output file as desired by the user. It's important to note that the output stream needs to be passed as a reference i.e., the function parameter for the output stream should be a reference type.

The **main function** should be responsible for using the other functions to perform the overall task. Two arrays of characters should be declared inside main -- one for the original html content and one for the converted txt content. You may use a predefined physical size of 600 for these arrays. Use a size of 20 for declaring the input and output file names. Once the html content is converted, you will need to pass either the standard output stream or a file output stream to the outputTxtFile function based on the user choice. Two sample executions of the program are shown at the end.

Instructions:

This will be an individual programming project.

Write your code in a file called prog3.cpp.

Use meaningful variable names, helpful comments, and a consistent coding style.

Your program file should have the appropriate comment block at the top:

// Name: Your Name

// File Name: prog3.cpp

// Date: Day Month, Year

// Program Description: brief description of the program

All functions should have a header comment block describing the purpose of the function, input, and output:

```
/******
```

```
* Name: Name of the function.
```

```
* Description: Description of why the function exists and
```

```
* how it accomplishes its goal.
```

```
* Input: Parameter list with descriptions
```

```
* Output: Return value with description
```

```
*****/
```

Deliverables:

You will need to submit the following in Blackboard by 3:00 PM on October 18th:

The source file prog3.cpp and one screenshot of your program execution.

Grading:

This project is worth 30 points distributed as follows:

Demo of Part I (5 pts)

- Implementation of the readHtmlFile function (2.5 pts)
- Implementation of the outputTxtFile function (2.5 pts)

Part II (15 pts)

- Implementation of main (5 pts)
- Implementation of convertHtmlFile function (10 pts)

Program Style (5 pts)

- Meaningful variable names (1 pt)
- Proper indentation (1 pt)
- Sufficient comments (3 pts)

Report (5 pts)

- Project specification (1 pt)
- Solution summary (1 pt)
- Testing documentation (3 pts)

You will lose 50% of your points if your program does not compile!

Sample Execution I:

HTML/TXT Converter

1) Display TXT to Console

2) Write TXT to file

Enter your choice: 1

Enter html filename: test.html

A novice asked the master: "I have a program that sometime runs and sometimes aborts. I have followed the rules of programming, yet I am totally baffled.

What is the reason for this?"

The master replied: "You are confused because you do not understand Tao. Only a fool expects rational behavior from his fellow humans. Why do you expect it from a machine that humans have constructed? Computers simulate determinism; only Tao is perfect.

Sample Execution II:

HTML/TXT Converter

1) Display TXT to Console

2) Write TXT to file

Enter your choice: 2

Enter html filename: test.html

Enter txt filename: test.txt

Output written to txt file.

** Anything typed in blue indicates a user input.