CPSC 327

Teams: None, please work individually on this project

References:

1. Pointers Memory lectures and projects
2. Command line parameters, lectures and projects
3. File IO lectures and projects

Sample Code:
See starter project on course website projects folder
See pointer_proj_small_SOLUTION. An application that you can run from a terminal window to see what output should be. You will have to set the permissions on this file after cloning the repo so that it is executable. Type the following line at a terminal window;
chmod 755 pointer_proj_small_SOLUTION

Topics covered by this project;

- Using pointers to manipulate char data
- Dynamic memory allocation
- Namespaces
- Command Line Arguments

Outline
Please create an application that reads a file (inputfile) , replaces all occurrences of a particular string (tag) with a new string (tag_replacement), and writes that new string to a new file (outputfile).

This project has 3 parts;
1. In pointer_proj_small.cpp, main(…) - Commandline parameter processing and expected program flow
2. In stringmanip.cpp - Pointer based char string processing
3. In fileio.cpp – read and write char data to files

In pointer_proj_small.cpp, main(…) Commandline parameter processing and expected program flow
(see constants.h for following constants)

1. If user passes 1 argument, and it is the HELP_CHAR, the program should output HELP_STRING1, HELP_STRING_2 and should then exit returning SUCCESS.
2. Otherwise if the user does not provide EXPECTED_NUMBER_ARGUMENTS the program should output HELP_STRING_2 and should then exit, returning FAIL_WRONG_NUMBER_ARGS.
3. Otherwise the user has entered 4 params;
   an inputfile,
   an ouputfile
   a tag to search for
   a replacement token
4. Open and read inputfile into string1.
5. Determine number of tags in string1.
6. Calculate the amount of memory needed if tag is replaced by tag_replacement in string1 (call it string 2).
7. Allocate a char array on the heap that is large enough to hold string2.
8. Copy string1 to string2, replacing tag with tag_replacement.
9. Save string2 to outputfile
10. Delete any dynamically allocated memory

In stringmanip.cpp - Pointer based char string processing

Please provide implementations for the given functions in stringmanip.h. You must use char * pointers only in this file, you may not use std::string to parse the character arrays. You may use functions defined in <string.h>.

HINT: The const char * pointers that are used make it easy to get a pointer to a strings internal char representation. For instance:
string mys = "abc";
const char *p = mys.c_str();

HINT: Iterate over a const char pointer:

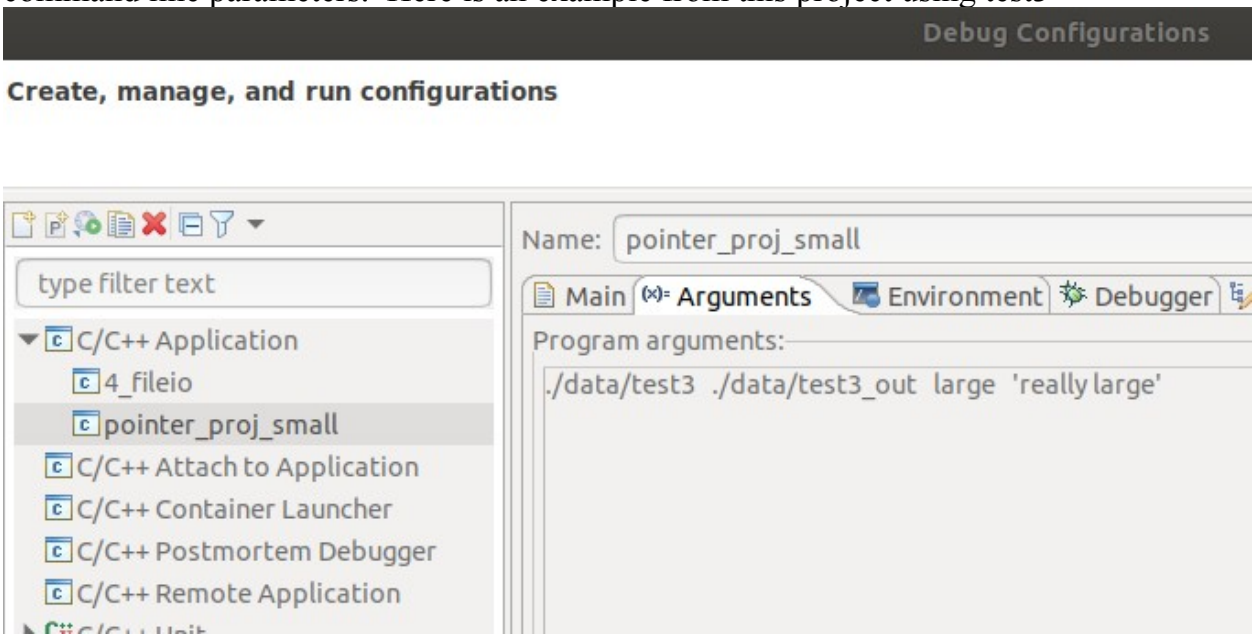
```
void replaceTagWithTagReplacement(const char *src, const char *tag...){
        :
        while (i<len_src){
                :
                if(*(src+i) == tag[0])
                        :
```

In fileio.cpp – read and write char data to files

Please provide implementations for the given functions in fileio.h

Passing paramseters in eclipse

When debugging in eclipse you should set your debug configuration to pass in all command line parameters. Here is an example from this project using test3



Assignment

Please fill in required content in;

- fileio.cpp
- stringmanip.cpp
- pointer_proj_small.cpp

Please submit only the above three files.

Sample runs are in the data directory with input files and it's associated output file (for instance test1 and test1_out).

Please ensure that all parsing in stringmanip.cpp is done with pointers, no algorithms or built in string parsing allowed.

Please run valgrind on your project to ensure there are no memory errors.

Grading:
I will compile and link your solution. I will probably use my own datasets to test regular and edge conditions.

15% Valgrind
15% commandline args
15% main
15% fileio.cpp
40 % stringmanip.cpp

Special cases:
-5 turn in more than 3 files
-100 does not compile