CSCE 1030 – Homework 4

Due: 11:59 PM on Wednesday, April 8, 2015 CDT

Problem Statement:

The purpose of this programming project is to write a C++ program incorporating functions that use call-by-reference as well as reading and writing from and to I/O streams. In particular, this programming assignment will take a poorly formatted file containing a listing of some recent top boy and girl names, with rank, and write a new file with this information formatted to certain specifications.

Your program's output should initially call a function to display the department and course number, your name, your EUID, and your e-mail address.

You will then prompt for and read in the name of the file containing the list of popular boy and girl names. If there is a problem opening the file, you will display a meaningful error and exit the program using the appropriate exit status. You may assume that the length of the file name does not exceed 32 characters (33 with the terminating null character).

Similarly, you will prompt for and read in the name of the file that will contain (i.e., you will write to this file) the formatted information for the list of top boy and girl names. If there is a problem opening the file, you will display a meaningful error and exit the program using the appropriate exit status. You may assume that the length of the file name does not exceed 32 characters (33 with the terminating null character).

You will write a function to process the input file that will properly format the information according to given specifications. In particular, each line will contain the rank of the names, followed by the boy name, and then the girl name. The formatting shall follow the following specifications:

- The rank shall have a width of 4 characters with the rank number right justified among these 4 characters. You may assume that the list contains at most 9999 rankings, so there is no issue with a ranking number having more than 4 characters in its numeric representation. There will be a space added between the number ranking and the boy name.
- The boy name shall contain at most 12 characters. The boy names shall be left justified among these 12 characters and any boy names not consisting of 12 characters shall have spaces added at the end of the name to fill the 12 characters. There will be a space added between the 12 character/spaces for the boy name and the girl name. Additionally, all boys' names should be formatted as proper names, with the first letter capitalized and remaining letters in the name in lowercase.
- Similarly, the girl name shall contain at most 12 characters. The girl names shall be left justified among these 12 characters and any girl names not consisting of 12 characters shall have spaces added at the end of the name to fill the 12 characters. Following the girl name, there will be a newline character added so that the next ranking names appear on a new line. Additionally, all girls' names should be formatted as proper names, with the first letter capitalized and remaining letters in the name in lowercase.

This programmer-defined function (to process the input file) should accept two parameters: the input file stream and the output file stream. Be sure to take care of all steps in working with the files.

You will be given a sample file containing a subset of the names in its raw (poorly formatted) form to test your program. See the sample program run for an example of what should be output to the file.

Design:

On a piece of paper (or word processor), write down the algorithm, or sequence of steps, you will use to solve the problem. You may think of this as a "recipe" for someone else to follow. Continue to refine your "recipe" until it is clear and deterministically solves the problem. Be sure to include the steps for prompting for input, performing calculations, and displaying output.

Type these steps into a document (Word, txt, PDF, etc.). Note that this should be done before you start coding as completing it afterwards does not help you in learning the design process.

Implementation:

Now that you have a working design, your next step is to translate these steps into C++ code. Use the algorithm development techniques discussed in class to implement your solution to the problem above. Add your C++ code a little at a time, and compile and test as you go.

Remember to add your comments to your code to explain your program. Do this before/during programming instead of waiting until the end. At a minimum, you should comment the header (e.g., name, class, date, brief description of the program, etc.), all variables (i.e., what they are used for), and specific "blocks" of code. For example, use comments to describe the inputs, the formulas used, and any other important steps, such as loops, in your code.

Your program will be graded based largely upon whether it works correctly on a CSE Department machine, so you should make sure your program compiles and runs on a CSE machine.

Your program will also be graded based upon your programming style. At the very least, your program should include:

- A consistent indentation style as recommended in the textbook and in class;
- Meaningful variable names;
- A block header comment section that includes: your name, e-mail address, and a brief description of the program.

CSCE 1030 – Homework 4

Due: 11:59 PM on Wednesday, April 8, 2015 CDT

Testing:

Test your program to check that it operates as desired with a variety of inputs to make sure that all "paths" through you code are correct. Sample input and output appears below (with input shown in **bold**):

```
mat0299@cse05:~/csce1030$ more names.txt
1    liam      emmA      2    noah    olivia    3   ethAN    AVA   4
mason      SOPHIa     5  logan    isabella     6  lucas mia       7   jacob    charlotte
       8 JACKSON        emily    9     aiden       maDISON      10 elijah
amelia
mat0299@cse05:~/csce1030$ ./a.out
+---------------------------------------------+
|       Computer Science and Engineering      |
|        CSCE 1030 - Computer Science I       |
|   Student Name     EUID    euid@my.unt.edu  |
+---------------------------------------------+
Enter name of input file containing list of popular names: names.txt
Enter name of output file for formatted list of popular names: list.txt
mat0299@cse05:~/csce1030$ more list.txt
   1 Liam         Emma
   2 Noah         Olivia
   3 Ethan        Ava
   4 Mason        Sophia
   5 Logan        Isabella
   6 Lucas        Mia
   7 Jacob        Charlotte
   8 Jackson      Emily
   9 Aiden        Madison
  10 Elijah       Amelia
```

Documentation:

When you have completed your C++ program, write a short report (2 – 3 paragraphs) describing what the objectives were, what you did to solve the problem, and the status of the program. Does it work properly for all test cases? Are there any known problems? Also include a reflection in what you learned by completing this program that you anticipate taking forward as your knowledge in C++ programming grows.

Save this report in a separate file to be submitted electronically. You should also include any specific instructions required to compile or execute your code.

CSCE 1030 – Homework 4

Due: 11:59 PM on Wednesday, April 8, 2015 CDT

Homework Submission:

In this class, we will be using electronic homework submission to make sure that all students hand their programming projects (and labs) on time. You will submit your program source file to the class website through the "**Homework 4**" drop box by the due date and time.

Note that this project must be done individually. The program will be checked using a code plagiarism tool against other solutions, so please ensure that all work submitted is your own.

Note that the dates on your electronic submission will be used to verify that you met the due date above. All homework up to 24 hours late will receive a 50% grade penalty. Later submissions will receive zero credit, so hand in your best effort on the due date.

Summary:

- You will design an algorithm (or steps used) to solve the problem.
- You will implement your program on the CSE machines using C++. You will make sure to use good style, good variable names, indentation, etc. You will compile, run, and test your code.
- You will write a brief report describing what your code does and how well it works.
- You will submit electronically your C++ code, your design, and your brief report.

General Guidelines (for ALL of your programming assignments):

- Your program's output should initially display the department and course number, your name, your EUID, and your e-mail address.
- Use meaningful variable names.
- Use appropriate indentation.
- Use comments, including a program header. Example program header:

```
/*
 ============================================================================
 Name        : homework2.cpp
 Author      : Mark A. Thompson
 Version     :
 Copyright   : 2015
 Description : The program performs simple arithmetic operations based on in-
               put from the user.
 ============================================================================
 */
```

- Add a header to each function. Example function header:

```
/*
================================================================================
 Function    : deposit
 Parameters  : a double representing account balance and a double represent-
               ing the deposit amount
 Return      : a double representing account balance after the deposit
 Description : This function computes the account balance after a deposit.
================================================================================
 */
```