

CSci 1113: Introduction to C/C++  
Programming for Scientists and Engineers  
Homework 6  
Fall 2018

**Due Date: Wednesday, November 7, 2018 before 11:00pm.**

**Instructions:** This is an individual homework assignment. There are two problems worth 20 points each. Solve the problem below by yourself (unlike the labs, where you work collaboratively), and submit the solution as a C++ source code file. Here are a few more important details:

1. Unlike the computer lab exercises, this is not a collaborative assignment.
2. Because all homework assignments are submitted and tested electronically, the following are important:
  - You follow any naming conventions mentioned in the homework instructions.
  - You submit the correct file(s) through Moodle by the due deadline.
  - You follow the example input and output formats exactly given in each problem description.
  - **Regardless of how or where you develop your solutions, your programs compile and execute on cselabs computers running the Linux operating system.**
3. You should test your program on other test cases (that you make up) as well. Making up good test cases is a valuable programming skill, and is part of ensuring your code solution is correct.

**Problem A: Word count** (20 points)

Write a ***recursive*** C++ program that lets you enter a both a file and a word. This recursive function should count how many lines of the file contain the said word. You can get all your variables/objects setup in main before you use your recursive function. (Hint: Like you could put the file into an array or string.) You can assume there will not be more than 500 lines in the file.

Note: By “line contains”, I mean does the word appear anywhere on the line, even as part of another word. So in Example 1, when looking for the word “one”, words like “someone” should count as those three letters appear exactly as shown, despite them being part of a larger word.

Example 1 (user input is underlined):

```
What file?  
words.txt  
What word?  
one  
Number of lines with the word: one  
5
```

Example 1 (user input is underlined):

```
What file?  
words.txt  
What word?  
are  
Number of lines with the word: are  
8
```

When you are done, name the source code file `<username>_6A.cpp`. Here you replace `<username>` with your U of M email address; for example, if your email address is `smithx1234@umn.edu`, your file should be named `smithx1234_6A.cpp`. Then submit your program using the HW 6 Problem A submission link in Moodle.

**Problem B: Alpha-Beta numbers (20 points)**

Assume “Alpha numbers” are described by the following formula:

$$A(x) = A(x - 1)^2 + B(x) + 1$$

... where  $B(x)$  are “Beta numbers” defined as:

$$B(x) = B(\lfloor x/2 \rfloor) - A(\lfloor x/2 \rfloor) + x$$

Assume that  $A(0) = 0$  and  $B(0) = 0$ . Write a ***recursive*** C++ program that lets you enter a number, then find both the Alpha and Beta numbers for that value.

Example 1 (user input is underlined):

Which number?

1

Alpha number:

2

Beta number:

1

Example 1 (user input is underlined):

Which number?

5

Alpha number:

2313442

Beta number:

0

When you are done, name the source code file `<username>_6B.cpp`. Here you replace `<username>` with your U of M email address; for example, if your email address is `smithx1234@umn.edu`, your file should be named `smithx1234_6B.cpp`. Then submit your program using the HW 6 Problem B submission link in Moodle.

**Problem C: Pairs of people (5 points *extra credit*)**

Ask the user to enter a file name. This file has the format: “`<name1> : ) <name2>`”. This indicates that `<name1>` and `<name2>` want to be in a group together. Print out all possible pairings that make everyone happy. If there are no such pairings, display this as well. (Note: there should be no possible pairings if the number of people are odd, as one person will be “left out”.)

Example 1 (user input is underlined):

What file?

pref.txt

(Lysandros, Yale) (Sanjana, Lorea)

**Example 2 (user input is underlined):**

What file?

pref2.txt

(Lysandros, Sanjana) (Yale, Lorea) (Moyra, Arya) (Ninlil, Katja)  
(Lysandros, Sanjana) (Yale, Lorea) (Moyra, Ninlil) (Arya, Katja)  
(Lysandros, Sanjana) (Yale, Lorea) (Moyra, Ninlil) (Arya, Katja)  
(Lysandros, Yale) (Sanjana, Lorea) (Moyra, Arya) (Ninlil, Katja)  
(Lysandros, Yale) (Sanjana, Lorea) (Moyra, Ninlil) (Arya, Katja)  
(Lysandros, Yale) (Sanjana, Lorea) (Moyra, Ninlil) (Arya, Katja)  
(Lysandros, Ninlil) (Sanjana, Lorea) (Yale, Moyra) (Arya, Katja)  
(Lysandros, Ninlil) (Sanjana, Lorea) (Yale, Moyra) (Arya, Katja)  
(Lysandros, Ninlil) (Sanjana, Katja) (Yale, Lorea) (Moyra, Arya)  
(Lysandros, Katja) (Sanjana, Lorea) (Yale, Ninlil) (Moyra, Arya)

**Example 3 (user input is underlined):**

What file?

pref3.txt

No pairs found!

When you are done, name the source code file <username>\_6C.cpp. Here you replace <username> with your U of M email address; for example, if your email address is smithx1234@umn.edu, your file should be named smithx1234\_6C.cpp. Then submit your program using the HW 6 Problem C submission link in Moodle.