**Lab 9 (51 points)**

File Input and Output prewritten code is found in the #include <**fstream**> file Stream library.

The **fstream** library contains several *classes* that make file I/O functionality available: **ifstream** and **ofstream**.

The ***ifstream*** class is used for file input. The ***ofstream*** class is used for file output.

These classes contain the *functionality* that is needed to manipulate input and output files.

To access/use this built-in functionality, remember that both ifstream and ofstream are classes and that classes are TYPES. Since they are types, they can be used in a Declaration statement.

**Example:**

// **step 1** – make ifstream and ofstream classes available

#include <fstream>

int main() {

// **Step 2** – Declare instances of the classes.

// These two declaration statements causes the variables and functions of each class type

// to be placed in memory and made available for your use.

ifstream fin; // ‘fin’ now contains many functions for file input manipulation

ofstream fout; // ‘fout’ now contains many functions for file output manipulation

// **Step 3** – Use fin and fout functionality

// Use of fin and fout functions

// To access the functionality within fin and fout, you use the dot notation.

fin.open(“inputFile”); // opens-connects to input file

fin.close(); // closes-disconnects from input file.

fin.get(parm); // gets a char

fin >> aword // reads a word – note aword is a string type. Note >> is really a function.

fin.getline(parms) // reads a line. Note: a line terminate with an enter character, not wrap around.

and more functions …

… Excellent descriptions and “Working Code” examples for all functionality in found at:

<http://www.cplusplus.com/reference/iostream/fstream/>

**Exercise 1: (17 points)**

Word Count Program:

Write a program that reads a text file and prints out the number of occurrences of words found in the file ‘Think.txt’.

Create the file ‘Think.txt’. Place the following text in the file: “Thinking is Identifications. Thinking is necessary in order to know what you are looking at. When you know what you are looking at, then you can act in accord with the laws of nature. Acting in accord with the laws of nature, a things identity, allows you to build, create and make the items necessary to live. Not acting according the laws of nature by committing mental evasion of the need to think, of the need to identify, results in fraud, theft and physical force. So choosing to think or not to think is choosing life or death. If you choose to think, then you have right to defend you integrity/identity from those who choose not to think, to defend your identify from their evasion of identity by their fraud, theft and initiation of force.”

Create a second input file. Find a document online and save it to a second input file. Test your program with it also.

Write each unique word and its count to an output file called ‘wordcount.txt’.

**Exercise 2: (17 points)**

Secret Message: An Encode and Decode program

Write a program that accepts “Command line input” to select ‘Encode’ and ‘Decode’, and input and output file names.

Example: code 1 english.txt code.txt // encodes

Example: code 2 code.txt English.txt // decodes

Encode: This part of the program reads in the input message, converts each character read into a hex value and writes out the hex values to an ‘Encoded’ Output file.

Decode: This part of the program reads in a hex encoded file and converts it ‘Decodes’ back into regular characters.

Test with this Secret Message: ‘The winning lottery ticket number for Saturday in 12 42 32 11 9 and 6.’

Next, create a Message of your own. Encoded your message and give the encoded file to two of your classmates, have them decode it.

**Exercise 3: (17 points)**

Statistics Program:

Create a data file, called ‘inputNumbers.txt’, containing about 100 integers.

The file should have *about* 10 lines of 10 integers.

Write a function to read the integers into an array.

The program should include functions to determine and display the average of the numbers (a real value), the number of integers greater than the average, and the number of integers less than the average, the maximum number and the minimum number.

Write results out to a file called summarystats.txt

**Exercise 4: (extra credit 10 points)**

Merge files Program:

Create a data file that has 10 records (10 lines of data). Each line will have a person’s first name, last name, address, city state, zip, home phone, annual income and political party.

Create a file that has a canned letter seeking donations that will have the words FIRSTNAME, LASTNAME, ADDRESS, CITY, STATE, ZIP, HOMEPHONE, ANNUAL INCOME and PARTY in it. The words are called FIELDS. Compose an interesting letter seeking political donations.

Write a program that will reads in the canned letter and stores it.

Write a function that will read the 10 records into an array.

Write a function that will write out a letter that will seek political donations for the party, replacing each FIELD with actual data from a person record. Produce a letter for each person. That is, replace the FIELDS with actual values and print out a letter for each of the ten people in the input file.

For All Exercises: Print the input files, output files, along with the source code and necessary screen prints.