

## Chapter 10 Homework – Strings

### Reading Assignment:

Read Chapter 10, Sections 1-2 in Introduction to Programming with C++, 3<sup>rd</sup> Edition by Liang

### Problem Assignment:

1. (25 pts) A company named XYZ Corporation develops Email addresses for their employees by using the first 4 digits of their last name (or all digits if less than 4), the first digit of their first name, and the last 3 digits of their EmployeeID, followed by “@XYZ.COM”. Write a program that will prompt the user to enter his/her last name, first name, and their EmployeeID. The program should then display their Email address. Turn in a printout of the program and the results for the 3 cases below. The 3<sup>rd</sup> case uses **your** name and any 7-digit EmployeeID you pick (make one up).

Inputs	Output
Last Name: Stephens First Name: Clarence EmployeeID: 9991234	Your Email address is StepC234@XYZ.COM
Last Name: Doe First Name: John EmployeeID: 5599876	Your Email address is DoeJ876@XYZ.COM
Last Name: (yours) First Name: (yours) EmployeeID: (pick one)	Your Email address is ...

2. (25 pts) Create a simple data file like the example shown below containing the 4 dates below plus 10 or more additional dates. The file should include 1 date per line and each date should have the form **MonthNumber-DayOfTheMonth-Last2DigitsOfTheYear** with no extra spaces. All dates should be in this century. No error checking for invalid dates is necessary.

```
2-19-15
7-4-14
11-4-17
9-30-16
.
.
```

Write a program that will read the dates in the input date file and create an output data file with the form **Month DayOfTheMonth, Year**. An example is shown below.

```
February 19, 2015
July 4, 2014
November 4, 2017
September 30, 2016
.
.
```

Turn in printouts of the program, the input data file, and the output data file.

(continued)

3. (28 pts) Download the file USDictionary.txt from the course Blackboard site that contains words in the US dictionary (about 118,000 words – all in lower case). Write a C++ program that will determine and display the following items:

- The total number of words in the dictionary.
- The total number of characters in the dictionary (not including white spaces)
- The total number of characters in the dictionary (including white spaces) – Hint: Use get( ).
- The total number of words ending in the letter e.
- The total number of 6 letter words.
- The total number of words beginning with a vowel.
- The total number of words containing the substring “ate”.
- The total number of occurrences of the letter e.
- The total number of words containing at least two occurrences of the letter e.

Turn in a printout of the program and the results.

4. (22 pts) Determine the output for the program below **by hand**. This is good test practice.

```
// Project: StringsHW
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string S1,S2,S3,S4,S5,S6,S7,S8,S9,S10,S11,S12,Vowels = "aeiou";
    int I1,I2,I3,I4,I5,I6,I7,I8;
    int Count1 = 0,Count2 = 0,Count3 = 0;
    S1 = "To be or not to be, that is the question.";
    S2 = S1.substr(0,9);
    S3 = S1.substr(9,9);
    S4 = S2 + S3;
    I1 = S1.find(S3);
    I2 = S1.find("be");
    I3 = S1.find("be",8);
    I4 = S1.find("be",17);
    I5 = S1.rfind("be",17);
    I6 = S3.length();
    I7 = S1.length();
    S5 = S1.substr(I7-9,8);
    S6 = S3;
    S6.erase(3,7);
    S7 = S4;
    S7.insert(9,S2);
    I8 = S1.find_first_of(Vowels);
    S8 = S2;
    S8 = S8.append(S2,0,5);
    S9 = S3;
    S9.replace(0,3,"NOT");
    S10 = S3;
    S10.resize(I6+4,'*');
    S11 = Vowels[4];
    S12 = S3;
```

```

for (int i = 3; i >=0; i--) S11 += Vowels[i];
for (int i = 0; i < I6; i++) S12[i] = toupper(S12[i]);
for (int i = 0; i < I7; i++)
{ if(S1[i] == 'o') Count1++;
  if(S1[i] == 116) Count2++;
  if(S1[i] >96 && S1[i] < 123) Count3++; }
cout << "S2 = " << S2 << endl;
cout << "S3 = " << S3 << endl;
cout << "S4 = " << S4 << endl;
cout << "I1 = " << I1 << endl;
cout << "I2 = " << I2 << endl;
cout << "I3 = " << I3 << endl;
cout << "I4 = " << I4 << endl;
cout << "I5 = " << I5 << endl;
cout << "I6 = " << I6 << endl;
cout << "I7 = " << I7 << endl;
cout << "S5 = " << S5 << endl;
cout << "S6 = " << S6 << endl;
cout << "S7 = " << S7 << endl;
cout << "I8 = " << I8 << endl;
cout << "S8 = " << S8 << endl;
cout << "S9 = " << S9 << endl;
cout << "S10 = " << S10 << endl;
cout << "S11 = " << S11 << endl;
cout << "S12 = " << S12 << endl;
cout << "Count1 = " << Count1 << endl;
cout << "Count2 = " << Count2 << endl;
cout << "Count3 = " << Count3 << endl;
return 0; }

```

S2 = \_\_\_\_\_

S3 = \_\_\_\_\_

S4 = \_\_\_\_\_

I1 = \_\_\_\_\_

I2 = \_\_\_\_\_

I3 = \_\_\_\_\_

I4 = \_\_\_\_\_

I5 = \_\_\_\_\_

I6 = \_\_\_\_\_

I7 = \_\_\_\_\_

S5 = \_\_\_\_\_

S6 = \_\_\_\_\_

S7 = \_\_\_\_\_

I8 = \_\_\_\_\_

S8 = \_\_\_\_\_

S9 = \_\_\_\_\_

S10 = \_\_\_\_\_

S11 = \_\_\_\_\_

S12 = \_\_\_\_\_

Count1 = \_\_\_\_\_

Count2 = \_\_\_\_\_

Count3 = \_\_\_\_\_