**In-Class-Exercise - ICE 5/23/2019 - Due on or before midnight of Sunday May 26, 2019**

**Objective:** C++ string

|  |
| --- |
| **Important instructions:**   * *All programs must include comments at the top of your program: your name, course name-section number (CSIT 575), program name and the program description in brief.* * *Copy and paste your program code and outputs in Part B of each program.* * *Once it is done, save and submit this word file via Canvas.* |

**CountOccurenceChar.cpp**

Write a program that counts and displays occurrences of the characters of a string entered from users.

You are given function prototype (using function is optional)

void countLetter(const string, int counts[]);

**Sample output:**

Enter a string: Betty Botter bought a bit of butter

a: 1 times

b: 5 times

e: 3 times

f: 1 times

g: 1 times

h: 1 times

i: 1 times

o: 3 times

r: 2 times

t: 8 times

u: 2 times

y: 1 times

**Copy and paste your program (source) code and the outputs after this line**

**+++++++++++++++++++++++++++++++++++++++++++++++++**

/\* Erik Gonzalez

CO SCI 575

CountOccurenceChar.cpp

This program counts and displays occurrences of the characters of a string entered from users.

\*/

#include <iostream>

#include <string>

using namespace std;

string selectionSort();

void charCount();

string sentance;

int main()

{

int count = 0;

cout << "Enter a string: ";

getline(cin, sentance);

cout << endl;

int length = sentance.length();

for (int i = 0; i < length; i++)

{

sentance[i] = tolower(sentance[i]);

}

selectionSort();

charCount();

system("Pause");

return 0;

}

//Sorts in alphabetical order

string selectionSort()

{

int startScan, minIndex;

char minValue;

int length = sentance.length();

for (startScan = 0; startScan < length; startScan++)

{

minIndex = startScan;

minValue = sentance[startScan];

for (int index = startScan + 1; index < length; index++)

{

if (sentance[index] < minValue)

{

minValue = sentance[index];

minIndex = index;

}

}

sentance[minIndex] = sentance[startScan];

sentance[startScan] = minValue;

}

return sentance;

}

/\* This function automatically assigns every character in the string

a value of one and will only increase if a character matches another

in the string. If it matches the character value will be one more than

it is supposed to be as in the beginning it was assigned an extra value.

So once it matches it will subtract that extra value to correct itself. \*/

void charCount()

{

int count = 0;

int length = sentance.length();

for (int i = 0; i < length; i++)

{

if (sentance[i] != ' ')

{

count++;

if (sentance[i] == sentance[i + 1])

{

count++;

int check = count - 1;

count = check;

}

else

{

cout << sentance[i] << ": " << count << " times" << endl;

count = 0;

}

}

}

}

**OUTPUTS:**

**Enter a string: Betty Botter bought a bit of butter**

**a: 1 times**

**b: 5 times**

**e: 3 times**

**f: 1 times**

**g: 1 times**

**h: 1 times**

**i: 1 times**

**o: 3 times**

**r: 2 times**

**t: 8 times**

**u: 2 times**

**y: 1 times**

**Press any key to continue . . .**

**Enter a string: Erik eats everyday at the lake**

**a: 4 times**

**d: 1 times**

**e: 6 times**

**h: 1 times**

**i: 1 times**

**k: 2 times**

**l: 1 times**

**r: 2 times**

**s: 1 times**

**t: 3 times**

**v: 1 times**

**y: 2 times**

**Press any key to continue . . .**