/// Andrew Souza

/// Comp 200 -- Fall 2023

/// Streams Assignment

#include <iostream>

#include <fstream>

#include <cctype>

using namespace std;

struct Letter {

char upper;

char lower;

bool inKeyword = false;

};

// Encryption algorithm to define secret code

Letter\* EncryptKeyword(string keyword, Letter\* reference) {

int count = 0; // Counts number of unique characters

Letter\* output = new Letter[26];

for (int i = 0; keyword[i] != '\0'; i++) {

for (int j = 0; j < 26; j++) {

if (keyword[i] == reference[j].upper || keyword[i] == reference[j].lower) {

if (reference[j].inKeyword == false) { // Each letter of the alphabet

reference[j].inKeyword = true; // is tested against each letter

output[count] = reference[j]; // in the keyword. When a match

count++; // is found, the flag is changed

} // from false to true

}

}

}

int j = 25;

for (int i = count; i < 26; i++) {

if (reference[j].inKeyword == false) { // j counts down each letter of

output[i] = reference[j]; // the alphabet, filtering out

} else { // the letters that appear

i--; // in the keyword. i decrements

} // when a letter in keyword is

j--; // reached to prevent empty spaces.

}

return output;

}

// Algorithm to encrypt message

string EncryptMessage(string message, Letter\* index, Letter\* reference) {

for (int i = 0; message[i] != '\0'; i++) {

for (int j = 0; j < 26; j++) {

if (message[i] == reference[j].lower) { // Each character is compared to

message[i] = index[j].lower; // each letter of the alphabet.

break; // When a match is found, the

} else if (message[i] == reference[j].upper) { // character is replaced by it's

message[i] = index[j].upper; // respective character in the

break; // encryption code.

}

}

}

return message;

}

int main() {

string keyWord;

string fileName;

Letter\* alphabet = new Letter[26];

Letter\* encryptedAlphabet = nullptr;

ifstream in;

ofstream out;

string data;

// Initialize array of each letter of the alphabet

for (int i = 0; i < 26; i++) {

alphabet[i].upper = 'A' + i;

alphabet[i].lower = 'a' + i;

}

// Prompt user for keyword

cout << "Please enter a keyword" << endl;

cin >> keyWord;

// Plug keyword, Alphabet, and size into algorithm to output secret code

encryptedAlphabet = EncryptKeyword(keyWord, alphabet);

// Take input from file to input into encryption algorithm

cout << "Input name of file to read: ";

cin >> fileName;

string outputName;

for (int i = 0; fileName[i] != '.'; i++) {

outputName += fileName[i];

}

outputName += "\_output.txt";

in.open(fileName);

out.open(outputName);

// Output encrypted message to text file

if (!in.fail()) {

while (!in.eof()) {

getline(in, data);

out << EncryptMessage(data, encryptedAlphabet, alphabet) << endl;

}

}

out.close();

delete[] alphabet;

delete[] encryptedAlphabet;

return 0;

}

