**Лабораторная работа № 2**

**Стандартная библиотека шаблонов**

**Выполнил студент группы ИВТ-12М**

**Ковальчук Антон**

**Вариант 9**

**Задание 1.**

****

#include <fstream>

#include <iostream>

#include <cstdlib>

#include <iterator>

#include <list>

#include <string>

#include <cstring>

#include <cctype>

#include <vector>

#include <algorithm>

int main()

{

using namespace std;

int n = 5;

//Вместо листа, воспользуемся вектором

vector <char> numbers = {'T','R','E','O','F' };

// перебор с помощью итераторов

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

{

cout << numbers[i] << " " ;

}

replace(numbers.begin(), numbers.end(), 'e', 'g');

replace(numbers.begin(), numbers.end(), 'E', 'G');

cout << endl;

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

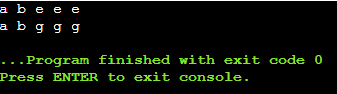
{

cout << numbers[i] << " ";

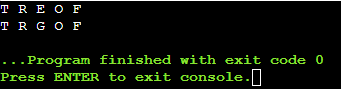
}

return 0;

}



Как мы видим, все элементы e в списке заменяются на все элементы g. Это работает и с заглавными буквами.



**Задание 2.**

****

#include <iostream>

#include <string>

#include <cctype>

int main()

{

std::string line = "Teoria, razrabotki, programmnogo, obespechenia";

int sub\_string = 1, tmp\_index, index;

int min = 9999, count\_ch = 0;

bool state1 = true, state2 = true;

for (int i = 0; i < line.size(); ++i)

{

if (line[i] == ',')

{

++sub\_string;

}

if (isalnum(line[i]))

{

state1 = false;

++count\_ch;

if (state2)

{

state2 = false;

tmp\_index = i;

}

}

else if (!isalnum(line[i]) && !state1)

{

state1 = true;

if (count\_ch < min)

{

state2 = true;

index = tmp\_index;

min = count\_ch;

}

count\_ch = 0;

}

}

std::cout << "Substring(s): " << sub\_string << '\n';

std::cout << "Minimal word: ";

while (isalnum(line[index]))

{

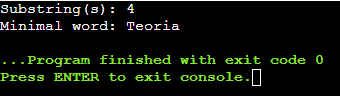
std::cout << line[index++];

}

return 0;

}

Результатом выполнения программы станет следующее



На картинке выше мы видим, что нам показывается количество подстрок, их действительно 4 (4 предложения, разделённых запятыми) и слово, в котором меньше всего букв, то есть в данном случае это слово теория. В том случае, если во всех подстроках будут содержаться слова с минимальным количеством букв, программа будет выдавать нам то слово, которое находится первым в первой подстроке

#include <iostream>

#include <string>

#include <cctype>

int main()

{

std::string line = "C, S, K, A";

int sub\_string = 1, tmp\_index, index;

int min = 9999, count\_ch = 0;

bool state1 = true, state2 = true;

for (int i = 0; i < line.size(); ++i)

{

if (line[i] == ',')

{

++sub\_string;

}

if (isalnum(line[i]))

{

state1 = false;

++count\_ch;

if (state2)

{

state2 = false;

tmp\_index = i;

}

}

else if (!isalnum(line[i]) && !state1)

{

state1 = true;

if (count\_ch < min)

{

state2 = true;

index = tmp\_index;

min = count\_ch;

}

count\_ch = 0;

}

}

std::cout << "Substring(s): " << sub\_string << '\n';

std::cout << "Minimal word: ";

while (isalnum(line[index]))

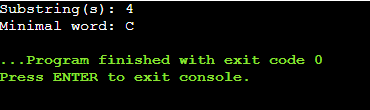
{

std::cout << line[index++];

}

return 0;

}



В данном примере во всех подстроках содержатся слова минимальной длины и, как мы видим, на изображении выше, программа выдаёт нам то слово, которое находится в первой подстроке.