МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт компьютерных технологий и информационной безопасности

Кафедра информационно-аналитических систем безопасности

имени Л.С. Берштейна

**ОТЧЕТ**

**По Лабораторной работе № 11**

***«Работа со структурами»***

**Вариант - 20**

Выполнил:

студент гр. КТбо1-10

Мельман Д.Э

Проверил:

Профессор кафедры ИАСБ

Барковский С.А.

«\_\_\_» 2020 г.

Таганрог – 2021

Код Программы:

#include <iostream>

#include <Windows.h>

#include <string>

using namespace std;

int stateMenu;

int spos;

int rpos;

int amount;

int b = 0;

struct state {

string Country;

string Capital;

string Language;

string Population;

string Money;

string Area;

}x[100];

state\* a;

// Добавить в очередь

void qstoreLine(state\* x) {

cout << "Введите название Страны: ";

getline(cin, x[spos].Country);

cout << "Введите Столицу: ";

getline(cin, x[spos].Capital);

cout << "Введите родной Язык: ";

cin >> x[spos].Language;

cout << "Введите количество населения:";

cin >> x[spos].Population;

cout << "Введите денежную единицу: ";

cin >> x[spos].Money;

cout << "Введите площать страны: ";

cin >> x[spos].Area;

spos++;

system("cls");

}

// Удалить из очереди

state\* qretrieveLine() { // Линейная очередь

if (rpos == spos) {

cout << "Стран больше нет" << endl;

}

rpos++;

a = new state[100];

int c = 0;

for (int i = rpos; i <= amount; i++) {

a[c] = x[i];

c++;

}

return a;

}

// Вывод данных

void showData(state\* Obj, int amount) {

int i = 1;

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

if (Obj[j].Country != "\0") {

cout << i << endl

<< "Страна: " << Obj[j].Country << endl

<< "Сталица: " << Obj[j].Capital << endl

<< "Язык: " << Obj[j].Language << endl

<< "Количество начеления: " << Obj[j].Population << endl

<< "Денежная единица: " << Obj[j].Money << endl

<< "Площадь: " << Obj[j].Area << endl

<< "--------------------------------------" << endl;

i++;

}

}

if (amount == 0) cout << "Стран не обнаружено" << endl;

system("pause");

system("cls");

}

// Реализация циклической очереди

state\* qretrieve() {

if (rpos == 100) rpos = 0; /\* установить на начало \*/

if (rpos == spos) {

cout << "Стран больше нет" << endl;

}

rpos++;

a = new state[100];

int c = 0;

for (int i = rpos; i <= amount; i++) {

a[c] = x[i];

c++;

}

return a;

}

// Изменить элемент очереди

void correctData(state\* x) {

int numb, ne;

string var, nov;

cout << "Введите номер элемента: ";

cin >> numb;

numb--;

cin.ignore(32767, '\n');

cout << "Что бы вы хотели изменить?" << endl;

getline(cin, var);

cout << endl;

if (var == "Страна") {

cout << "Введите новое значение: ";

getline(cin, nov);

x[numb].Country = nov;

}

else if (var == "Столица") {

cout << "Введите новое значение: ";

getline(cin, nov);

x[numb].Capital = nov;

}

else if (var == "Язык") {

cout << "Введите новое значение: ";

getline(cin, nov);

x[numb].Language = nov;

}

else if (var == "Население") {

cout << "Введите новое значение: ";

cin >> ne;

x[numb].Population = ne;

}

else if (var == "Финансы") {

cout << "Введите новое значение: ";

cin >> ne;

x[numb].Money = ne;

}

else if (var == "Прощадь") {

cout << "Введите новое значение: ";

cin >> ne;

x[numb].Area = ne;

}

else {

cout << "Проверьте правильность ввода" << endl;

}

}

// Меню

void MainMenu() {

cout << "\t\tМеню:" << endl

<< "1 - Добавить в очередь" << endl

<< "2 - Удалить из очереди" << endl

<< "3 - Показать очередь" << endl

<< "4 - Изменить элемент очереди" << endl

<< "0 - Выход" << endl;

cin >> stateMenu;

cin.ignore(32767, '\n');

};

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

MainMenu();

int c = 0;

int number = 0;

spos = rpos = amount = 0;

cout << endl;

while (stateMenu != 0) {

switch (stateMenu) {

case 1: // Добавить в очередь

system("cls");

do

{

qstoreLine(x);

amount++;

cout << "Добавить ещё? " << endl

<< "1 - да" << endl

<< "2 - нет" << endl;

cin >> number;

cin.ignore(32767, '\n');

system("cls");

} while (number != 2);

a = new state[100];

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

a[c] = x[i];

c++;

}

MainMenu();

break;

case 2: // Удалить из очереди

system("cls");

qretrieveLine();

MainMenu();

break;

case 3: // Показать очередь

system("cls");

showData(a, amount);

MainMenu();

break;

case 4: // Изменить элемент очереди

system("cls");

correctData(a);

MainMenu();

break;

case 0: // Выход

return 0;

default:

system("cls"); system("color 4");

cout << "ERROR: Попробуйте заново и введите нормальное значение!!!" << endl << endl;

system("pause"); system("color 7"); system("cls");

MainMenu();

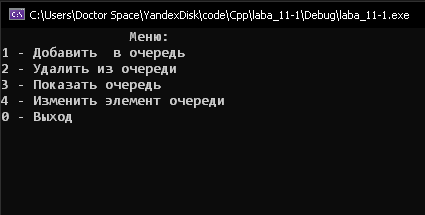
}

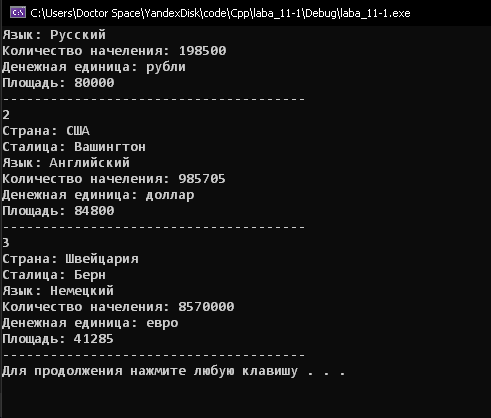
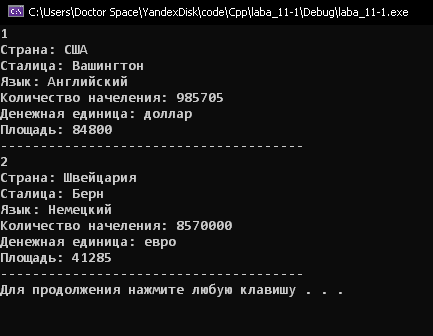
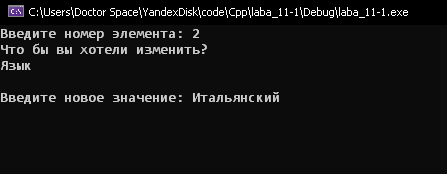
}

}

Пример:

1. Вывод меню



1. Вводим данные через добавление в очередь(1) и выводим через показать очередь (3) 
2. Теперь удалим первый элемент в очереди, нажмём на удаление из очереди (2)
3. Теперь изменим Язык у Швейцарии на другой, для этого мы заходим в пункт изменить элемент в очереди, выбираем 2-ой номер элемента и пишим что хотим изменить, в нашем случае Язык.
4. Теперь посмотрим на результат