Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«**Пермский национальный исследовательский политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЕТ**

Дисциплина: «Основы алгоритмизации и программирования»

Тема: Реализация стека через классы.

Семестр 2

Выполнил работу

Студент группы РИС-22-1Б

Токарев Павел Аркадьевич

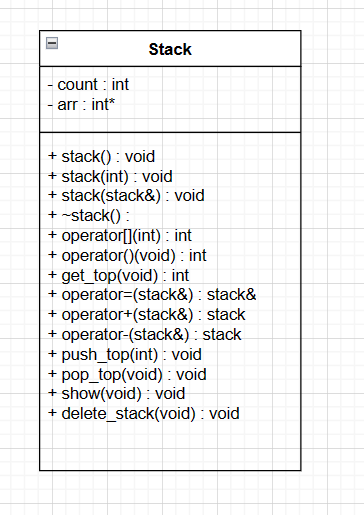
Проверил

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Г. Пермь-2023.

**Диаграмма UML**

****

**Код программы**

stack.cpp

#include "stack.h"

#include<iostream>

using namespace std;

stack::stack() {

count = 0;

arr = new int[10000];

}

stack::stack(const stack& a) {

count = a.count;

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

arr[i] = a.arr[i];

}

}

stack::stack(int n) {

int tmp;

count = n;

arr = new int[10000];

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

cout << i + 1 << " : ";

cin >> tmp;

push\_top(tmp);

}

}

stack::~stack() {

delete[]arr;

}

void stack::push\_top(int n) {

arr[count] = n;

count++;

}

void stack::pop\_top() {

count--;

}

void stack::show() {

cout << endl << "top" << endl;

int sz = count;

for (int i = sz -1;i >=0;i--) {

cout << arr[i] << endl;

}

cout << "bottom" << endl;

}

void stack::delete\_stack() {

count = 0;

}

int stack::operator()() {

return count;

}

int stack::operator[](int index) {

if (index < count) {

return arr[index];

}

}

stack& stack::operator=(const stack& t) {

count = t.count;

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

arr[i] = t.arr[i];

}

return\*this;

}

stack stack::operator+(const stack& t) {

stack a(t.count);

for (int i = 0;i < t.count;i++) {

a.arr[i] = arr[i] + t.arr[i];

}

return a;

}

stack stack::operator-(const stack& t) {

stack a(t.count);

for (int i = 0;i < t.count;i++) {

a.arr[i] = arr[i] - t.arr[i];

}

return a;

}

int stack::get\_top() {

return arr[count - 1];

}

stack.h

#pragma once

#include<iostream>

using namespace std;

class stack

{

private:

int count;

int \*arr;

public:

stack();

stack(int n);

stack(const stack&a);

~stack();

int operator[](int index);

int operator()();

int get\_top();

stack& operator=(const stack& t);

stack operator+(const stack& t);

stack operator-(const stack& t);

void push\_top(int a);

void pop\_top();

void show();

void delete\_stack();

friend ostream& operator<<(ostream& out, const stack& t) {

out << "top" << endl;

for (int i = 0;i < t.count;i++) {

out << t.arr[i] << endl;

}

out << "bottom" << endl;

return out;

}

friend istream& operator>>(istream& in, stack& t) {

for (int i = 0;i < t.count;i++) {

cout << i + 1 << " : ";

in >> t.arr[i];

}

return in;

}

};

main.cpp

#include <iostream>

#include"stack.h"

int main()

{

setlocale(LC\_ALL,"rus");

cout << "Введите размер стека : ";

int n,tmp;

cin >> n;

stack st;

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

cout << i + 1 << " : ";

cin >> tmp;

st.push\_top(tmp);

}

cout << "Введите количество элементов для добавления :";

cin >> n;

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

cout << i + 1 << " : ";

cin >> tmp;

st.push\_top(tmp);

}

st.show();

cout << "Введите количество элементов для удаления :";

cin >> n;

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

st.pop\_top();

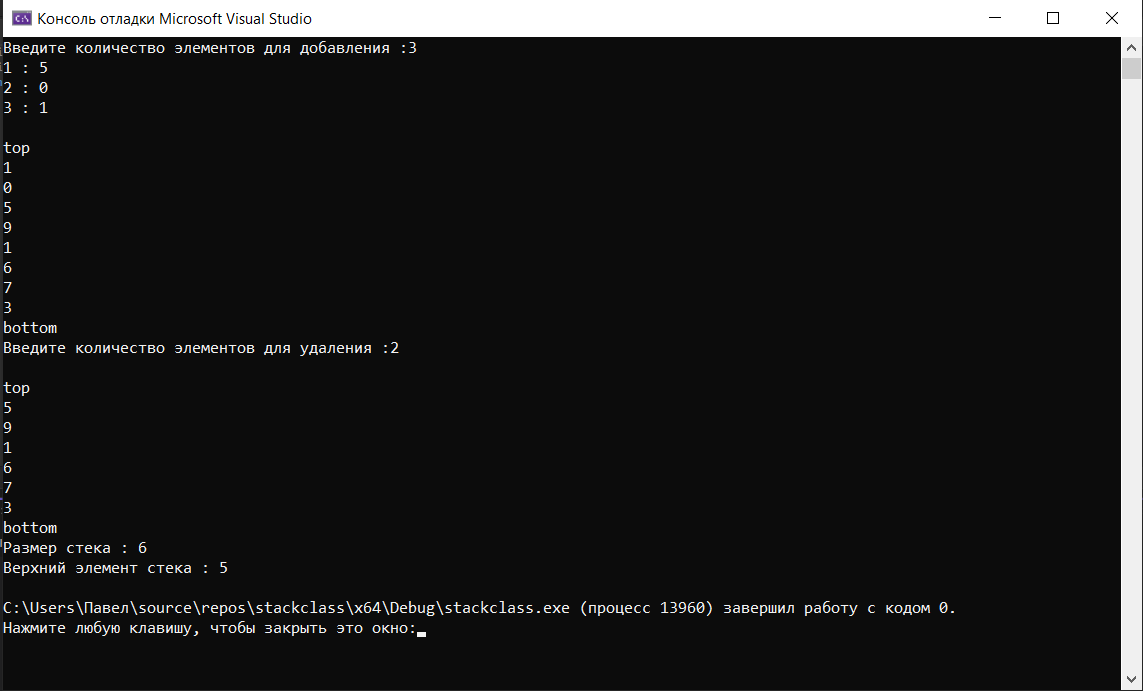
}

st.show();

cout << "Размер стека : " << st()<<endl;

cout << "Верхний элемент стека : " << st.get\_top() << endl;}

**Результат работы программы**

****

**Анализ результатов**

Удалось успешно реализовать стек через классы.