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»

По теме: «Стеки и очереди»

Семестр 2

Выполнил работу

Студент группы РИС-22-1Б

Бреднев Максим Павлович

Проверил

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Г. Пермь-2022

**Постановка задачи**

Реализовать работу стеков и очередей в коде.

**Стеки**

**Анализ переменных**

Count – кол-во переменных.

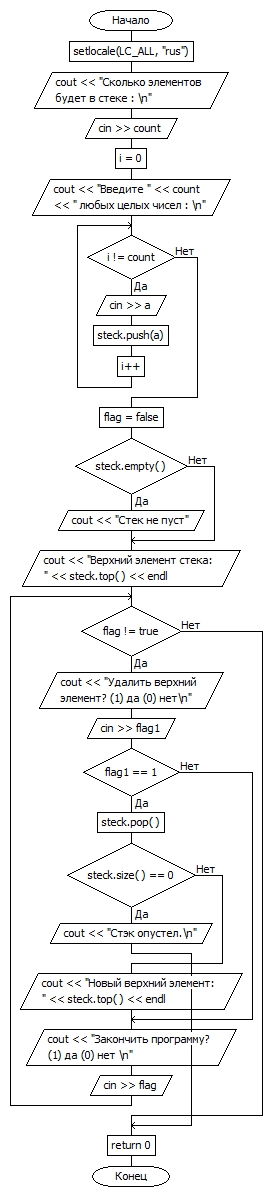
i – кол-во итераций ввода чисел в стек.

flag – переменная типа bool, переключатель работы цикла.

flag1 - переменная типа bool, переключатель работы цикла.

flag2 - переменная типа bool, переключатель работы цикла.

**Алгоритм программы**



**Код программы**

#include <iostream>

#include <stack>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

stack <int> steck;

int count;

cout << "Сколько элементов будет в стеке : \n";

cin >> count;

int i = 0;

cout << "Введите " << count << " любых целых чисел : \n";

while (i != count) {

int a;

cin >> a;

steck.push(a);

i++;

}

bool flag = false;

if (steck.empty())

cout << "Стек не пуст";

cout << "Верхний элемент стека: " << steck.top() << endl;

bool flag1;

bool flag2;

while (flag != true) {

cout << "\nУдалить верхний элемент? (1) да (0) нет\n";

cin >> flag1;

if (flag1 == 1) {

steck.pop();

if (steck.size() == 0) {

cout << "Стэк опустел.\n";

break;

}

cout << "Новый верхний элемент: " << steck.top() << endl << endl;

}

cout << "Закончить программу? (1) да (0) нет \n";

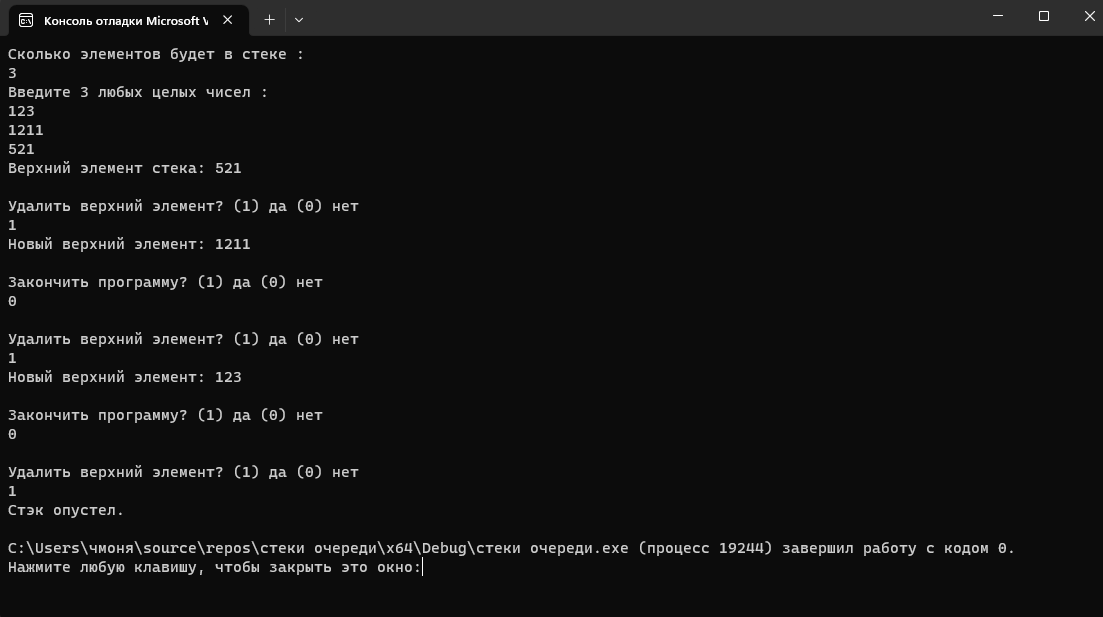
cin >> flag;

}

return 0;

}

**Результаты работы программы**

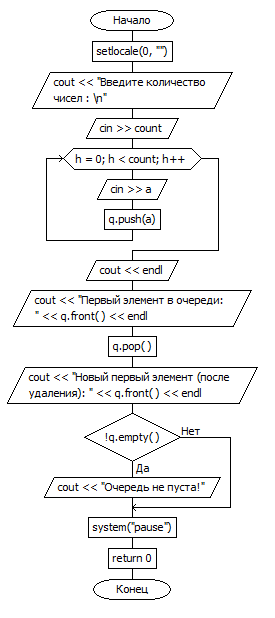


**Анализ результатов**

Как видно из результатов работы программы, код предлагает несколько выборов, удаляет верхний элемент из стека, проверяет пустой ли стек, показывает какой элемент сейчас на вершине.

**Очереди**

**Алгоритм**

****

**Код**

#include <iostream>

#include <queue>

using namespace std;

int main() {

setlocale(0, "");

queue <int> q;

int count;

cout << "Введите количество чисел : \n";

cin >> count;

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

int a;

cin >> a;

q.push(a);

}

cout << endl;

cout << "Первый элемент в очереди: " << q.front() << endl;

q.pop();

cout << "Новый первый элемент (после удаления): " << q.front() << endl;

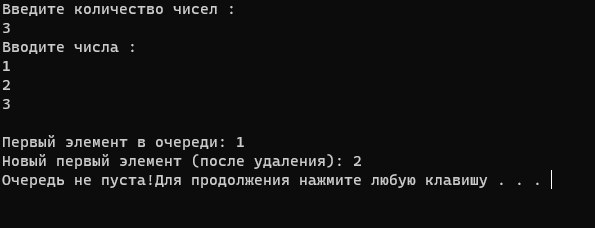
if (!q.empty()) cout << "Очередь не пуста!";

system("pause");

return 0;

}

**Результат программы**

****