МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «Полоцкий государственный университет»

Факультет информационных технологий

Кафедра технологий программирования

**Отчёт по лабораторной работе № 2 по курсу «Программирование сетевых приложений»**

«Реализация линейной структуры данных «Стек» и основные

алгоритмы обработки.»

Вариант №11

ВЫПОЛНИЛ студент группы 21-ИТ-1

Шиковец Е.А.

ПРОВЕРИЛ преподаватель

Виноградова А.Д.

Полоцк, 2022 г.

**ЦЕЛЬ РАБОТЫ:** ознакомиться с основами линейной структуры данных «Стек», изучить основные алгоритмы обработки ЛСД «Стек», научиться применять полученные знания на практике.

**ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ:**

1. Определение понятия стек.

Стеком называется упорядоченный набор элементов, в котором

размещение новых и удаление существующих происходит с одного конца,

называемого вершиной.

2. Опишите принцип работы стека.

Принцип работы стека сравнивают со стопкой листов бумаги: чтобы взять

второй сверху, нужно снять верхний.

3. Виды стека.

Различают аппаратный и программный стек.

Аппаратный стек используется для хранения адресов возврата из функций

и их аргументов.

Программный стек – это пользовательская модель (структура) данных.

4. Операции для работы со стеком.

Над стеком реализованы следующие операции:

• инициализация стека init(s), где s — стек

• помещение элемента в стек push(s, i), где s — стек, i — помещаемый

элемент;

• удаление элемента из стека i=pop(s);

• получение верхнего элемента стека без его удаления i=stkTop(s),

где s — стек

• получение количества элементов стека

• определение, пуст ли стек isempty(s) возвращает 1 если стек пустой

и 0 в противном случае.

• вывод элементов стека stkPrint(s), где s — стек

5. Перечислите способы реализации стека.

Существует несколько способов реализации стека:

• с помощью одномерного массива

• с помощью связанного списка;

• с помощью класса объектно-ориентированного программирования.

6. Для чего используется аппаратный стек?

Аппаратный стек используется для хранения адресов возврата из функций

и их аргументов.

**ОПИСАНИЕ ПРОДЕЛАННОЙ РАБОТЫ:**

При выполнении работы я пользовался IDE Visual Studio Code. В результате выполнения работы был написан следующий код:

#include <iostream>

#include <stack> // подключаем библиотеку для использования стека

#include <vector>

using namespace std;

int main()

{

stack<double> stack; // создаем стек 1

int i = 0, n;

vector<double> arr;

cout << "N: ";

cin >> n; // проверка на дурака

while (n < 0)

{

cin >> n;

}

cout << "Input " << n << " numbers " << endl; // предлагаем пользователю ввести n чисел

double mid = 0; // переменная среднего значения

for (int i = 1; i <= n; i++)

{

double num;

cin >> num;

mid += num; // считаем сумму всех чисел

stack.push(num); // добавляем введенные числа

arr.push\_back(num);

if (i == n) // делим сумму всех чисел на количество всех чисел

mid /= i;

}

cout << "Middle: " << mid << endl;

cout << "First stack:\n"; // первый стек

while (!stack.empty()) // удаление стека 1

{

cout << stack.top() << " ";

stack.pop();

}

cout << "\nSecond stack:\n"; // второй стек

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

{

if (arr[i] > mid) // проверяем, больше ли среднего значения и пушим в стек

stack.push(arr[i]);

}

while (!stack.empty()) // удаление стека 2

{

cout << stack.top() << " ";

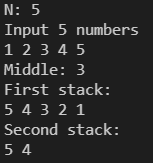
stack.pop();

}

return 0;

}

**РЕЗУЛЬТАТЫ ВЫПОЛЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ**:



**Вывод:** в результате выполнения лабораторной работы был реализован стек из n элементов, было выполнено индивидуальное задание. Также я разобрался в теории стеков, научился создавать, оперировать стеками на ЯП C++.