

Лабораторная работа №19

Использование указателей для организации связанных списков

1 Цель работы

1.1 Изучить процесс применения указателей для организации списков в программах на языке C++.

2 Литература

2.1 Ашарина, И. В. Объектно-ориентированное программирование в C++: лекции и упражнения. Учебное пособие для вузов. / И. В. Ашарина. – 2-е изд., перераб. и доп. – Москва : Горячая Линия–Телеком, 2017. – 336 с. – URL: <https://ibooks.ru/reading.php?productid=359752>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный. – гл.15.

3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

4 Основное оборудование

4.1 Персональный компьютер.

5 Задание

5.1 Создать программу, в которой описать структуру работающую по принципу «стек» и реализовать заполнение стека N элементами (все данные вводятся пользователем).

Для реализации стека разработайте самоссылочную структуру «узел», которая должна содержать ссылку на следующий элемент (узел) стека и хранить значение текущего значения.

5.2 Реализовать функцию вывода N элементов стека на экран без извлечения из стека.

5.3 Реализовать вывод всех элементов стека на экран (количество элементов в стеке неизвестно) без извлечения из стека.

5.4 Реализовать функцию извлечения из стека X элементов (X указывается пользователем).

5.5 Реализовать функцию вычисления среднего арифметического всех элементов стека.

6 Порядок выполнения работы

6.1 Используя Microsoft Visual Studio, создать проект C++ и выполнить задания из п.5.

6.2 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Что такое динамические структуры?

8.2 Какие виды динамических структур существуют?

8.3 На что указывает указатель на структуру в самоссылочных структурах?

8.4 Почему динамические структуры называются самоссылочными?

8.5 По какому принципу организованы стек и очередь?

Как расшифровывается аббревиатура LIFO?