Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №6**

**«Реализация элементарных структур данных на основе статической памяти»**

**«МДК 05.02 Разработка кода информационных систем»**

Выполнил: студент учебной группы

ИСПк-202-52-00

Рыков Ефим Витальевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

Цель работы: изучение принципов работы с базовыми структурами данных, получение навыков организации case-меню.

Задание:

* Написать программу для работы со структурой данных «Очередь».
* Структура данных должна быть реализована на основе статической памяти.
* Работа со структурой должна осуществляться с помощью case-меню.
* Предусмотреть наглядную визуализацию содержимого структуры.

Описание алгоритма:

Настоящий алгоритм был разработан для использования абстрактного типа данных «Очередь» реализованного с помощью списка. Для визуализации списка и управлением очередью используется case-меню.

Для работы с очередью реализованы функции: добавления элемента в очередь, взятия элемента из очереди, выведение содержимого очереди.

Каждый элемент очереди хранит в себе: значение, номер элемента в списке и ссылку на следующий элемент списка.

При добавлении нового элемента в очередь пользователь вводит значение элемента, а номер элемента и ссылка определяются автоматически.

При взятии элемента из очереди голова очереди ссылается на следующий элемент от первого, то есть на второй. Номера элементов в списке пересчитываются при каждом взятии элемента.

Визуализация элементов списка происходит с помощью вывода значений списка в консоль.

Схема алгоритма с комментариями:

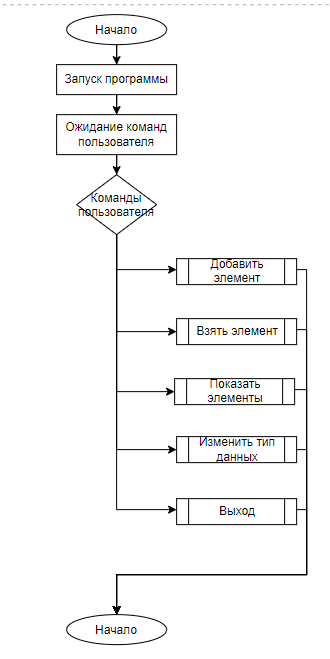


Рисунок 1 - Общие описание алгоритма

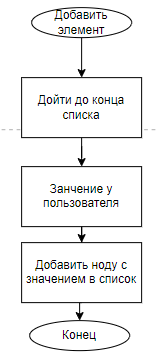


Рисунок 2 - алгоритм добавления элемента в очередь

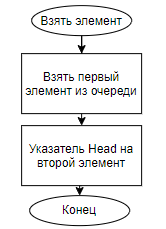


Рисунок 3- алгоритм взятия элемента из очереди

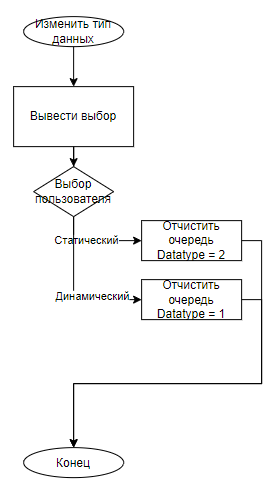


Рисунок 4 - алгоритм смены типа данных

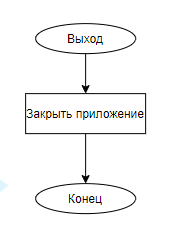


Рисунок 5 - алгоритм выхода