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»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-203-52-00

Исакова Кристина Валерьевна

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

1. Цель работы: изучение принципов работы с базовыми структурами данных, получение навыков организации case-меню.
2. 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.

1. Скриншот задания

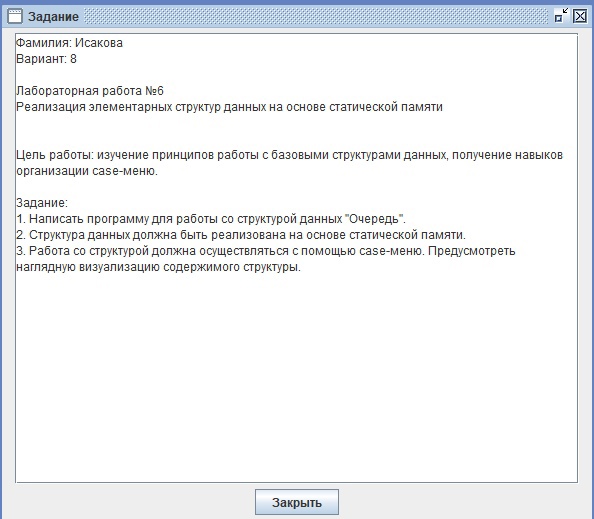


Рисунок 1 – Скриншот задания

1. Описание алгоритма

**Очередь** — динамическая структура данных, у которой в каждый момент времени доступны только два элемента: первый и последний. Добавление элементов возможно только с одного конца (конца очереди), а удаление элементов – только с другого конца (начала очереди).

Cуществует сокращение для очереди: FIFO = First In – First Out, с английского — «Кто первым вошел, тот первым вышел».

**Для очереди доступны следующие операции:**

* добавить элемент в конец очереди (PushTail);
* удалить элемент с начала очереди (Pop).

1. Код программы

**program** dkr66;

**const**

MAX\_SIZE = 10;

**type**

QueueArray = **array**[1..MAX\_SIZE] **of** integer;

**var**

queue: QueueArray;

front, rear: integer;

choice, value: integer;

**procedure** InitializeQueue();

**begin**

front := 0;

rear := 0;

**end**;

**function** IsEmpty(): boolean;

**begin**

**if** front = rear **then**

IsEmpty := true

**else**

IsEmpty := false;

**end**;

**function** IsFull(): boolean;

**begin**

**if** rear = MAX\_SIZE **then**

IsFull := true

**else**

IsFull := false;

**end**;

**procedure** Enqueue(value: integer);

**begin**

**if** IsFull() **then**

writeln('Очередь заполнена.')

**else**

**begin**

rear := rear + 1;

queue[rear] := value;

**end**;

**end**;

**procedure** Dequeue();

**begin**

**if** IsEmpty() **then**

writeln('Очередь пуста.')

**else**

**begin**

front := front + 1;

**end**;

**end**;

**procedure** DisplayQueue();

**var**

i: integer;

**begin**

**if** IsEmpty() **then**

writeln('Очередь пуста.')

**else**

**begin**

writeln('Содержимое очереди: ');

**for** i := front + 1 **to** rear **do**

**begin**

write(queue[i], ' ');

**end**;

writeln();

**end**;

**end**;

**begin**

InitializeQueue();

**repeat**

writeln('1 - Добавить элементы в очередь');

writeln('2 - Удаление элементов из очереди');

writeln('3 - Показать содержимое очереди');

writeln('4 - Выход');

writeln('Введите что вам необходимо:');

readln(choice);

**case** choice **of**

1:

**begin**

writeln('Введите значение очереди:');

readln(value);

Enqueue(value);

**end**;

2:

Dequeue();

3:

DisplayQueue();

4:

writeln('Bye! Bye!');

**end**;

**until** choice = 4;

**end**.

1. Результат выполнения программы

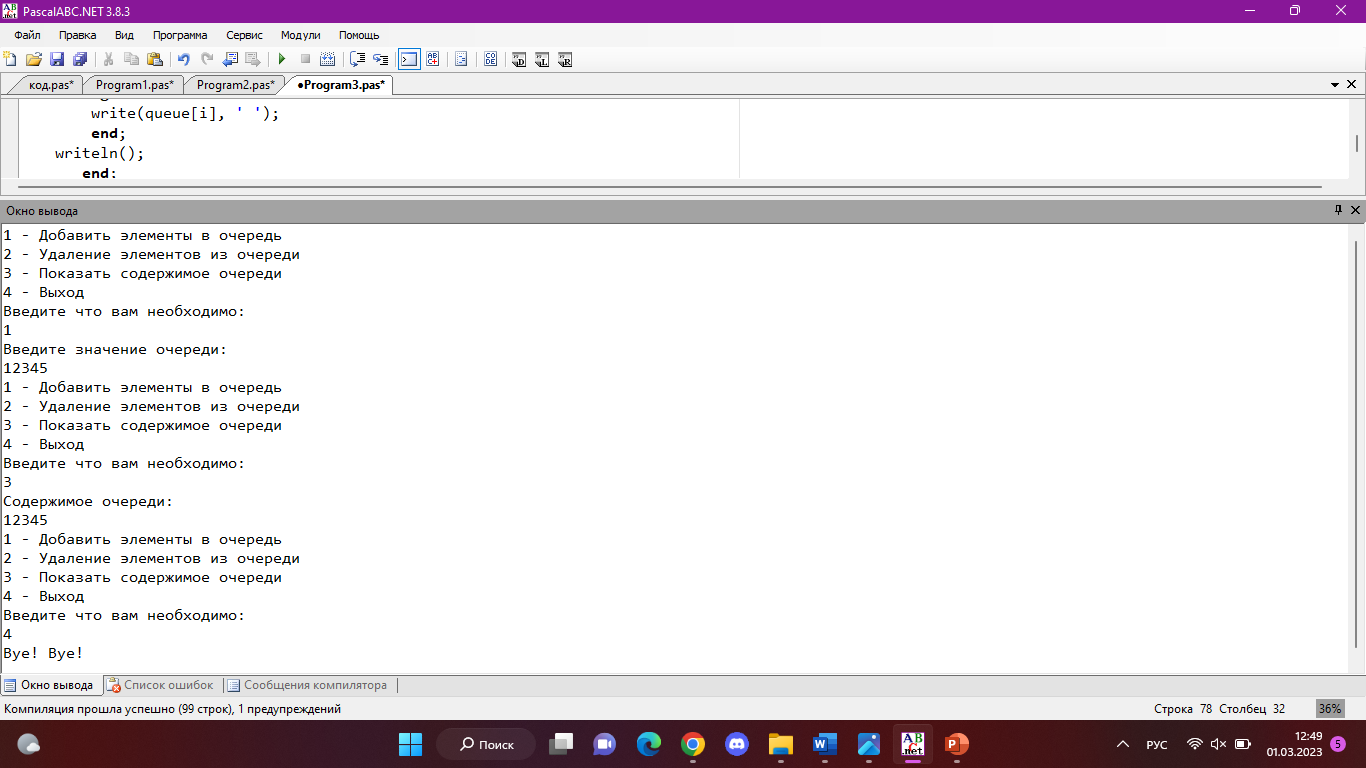
****

Рисунок №1 – Результат выполнения программы

1. Вывод

В ходе выполнения домашней контрольной работы № 6 был написан код программы, при выполнении которой была написана программа для работы со структурой данных "Очередь". Необходимо было вспомнить базовые конструкции языка программирования Pascal. В коде программы были использованы: процедуры, функции и case-menu. Процедуры представляет собой последовательность операторов, которая имеет имя, список параметров и может быть вызвана из различных частей программы. Также была использована работа с клавиатурой. Необходимо было изучить структурные данные на основе статической памяти, что прошло успешно. При выполнении данной работы были получены базовые навыки работы с процедурами и вызовом клавиатуры. Также возникли некоторые трудности, но они были разрешены.

Знания, полученные на лекциях и практических занятиях по предмету «Основы алгоритмизации и программирования», помогли в написании данной работы. В результате работы были достигнуты все поставленные цели.