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 ЭКСТРЕМАЛЬНЫХ СТРУКТОР ДАННЫХ НА ОСНОВЕ СТАТИЧЕСКОЙ ПАМЯТИ»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-203-52-00

Тетерина Юлия Константиновна

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

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

**Задание и вариант:**

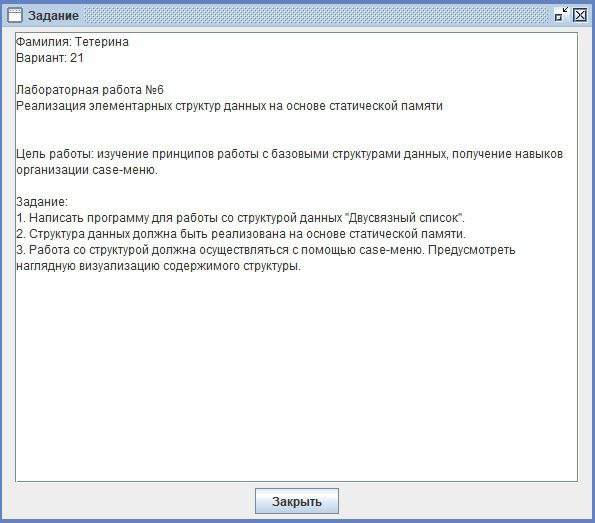


Рисунок 1. – Задание и вариант

**Описание алгоритма:**

Двусвязный (двунаправленный) список — это разновидность связного списка, при которой переход по элементам возможен в обоих направлениях (как вперед, так и назад), в отличие от односвязного (однонаправленного) списка.

**Код программы:**

**uses** Crt;

**const**

MAX\_NODES = 100;

**type**

TNode = **record**

data: integer;

next: integer;

prev: integer;

**end**;

**var**

head, tail: integer;

nodeArray: **array** [1..MAX\_NODES] **of** TNode;

nodeCounter: integer;

**procedure** InsertNode(**var** head, tail: integer; data: integer);

**var**

node: integer;

**begin**

Inc(nodeCounter);

node := nodeCounter;

nodeArray[node].data := data;

nodeArray[node].next := 0;

nodeArray[node].prev := 0;

**if** head = 0 **then**

**begin**

head := node;

tail := node;

**end**

**else**

**begin**

nodeArray[tail].next := node;

nodeArray[node].prev := tail;

tail := node;

**end**;

**end**;

**procedure** DeleteNode(**var** head, tail: integer; data: integer);

**var**

node, prevNode, nextNode: integer;

**begin**

node := head;

**while** (node <> 0) **and** (nodeArray[node].data <> data) **do**

**begin**

node := nodeArray[node].next;

**end**;

**if** node <> 0 **then**

**begin**

prevNode := nodeArray[node].prev;

nextNode := nodeArray[node].next;

**if** prevNode <> 0 **then**

**begin**

nodeArray[prevNode].next := nextNode;

**end**

**else**

**begin**

head := nextNode;

**end**;

**if** nextNode <> 0 **then**

**begin**

nodeArray[nextNode].prev := prevNode;

**end**

**else**

**begin**

tail := prevNode;

**end**;

Dec(nodeCounter);

**end**;

**end**;

**procedure** PrintList(head: integer);

**var**

node: integer;

**begin**

node := head;

**while** node <> 0 **do**

**begin**

Write(nodeArray[node].data, ' ');

node := nodeArray[node].next;

**end**;

Writeln;

**end**;

**var**

choice: integer;

data: integer;

**begin**

head := 0;

tail := 0;

nodeCounter := 0;

**repeat**

Writeln('Введите число: ');

Writeln('1. Ввод данных');

Writeln('2. Удаление данные');

Writeln('3. Вывести список данных');

Writeln('4. Выход из меню');

Readln(choice);

**case** choice **of**

1: **begin**

Write('Введите данные: ');

Readln(data);

InsertNode(head, tail, data);

**end**;

2: **begin**

Write('Введите данные: ');

Readln(data);

DeleteNode(head, tail, data);

**end**;

3: **begin**

Writeln('Список данных:');

PrintList(head);

**end**;

**end**;

**until** (choice = 4);

**end**.

**Результат выполнения:**

Изображение выглядит как текст, монитор, компьютер, внутренний

Автоматически созданное описание

Рис. 2 – Результат выполнения

Изображение выглядит как текст

Автоматически созданное описание

Рис. 3 – Результат выполнения

Изображение выглядит как текст

Автоматически созданное описание

Рис. 4 – Результат выполнения

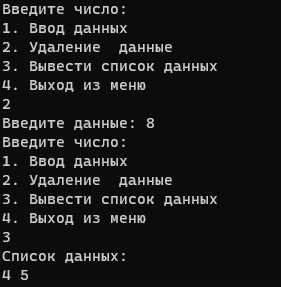


Рис. 5 – Результат выполнения

Изображение выглядит как текст

Автоматически созданное описание

Рис. 6 – Результат выполнения

**Вывод:** при выполнении данной домашней контрольной работы нам пришлось очень сильно постараться! Мы смотрели и изучали информацию, которая будет полезна для понимания как написать код заданной программы, напрягали все свои извилины чтобы полностью осмыслить алгоритм. Постепенно мы начали вникать в суть работы, все поняли и начали действовать. В итоге работы мы узнали, что такое списки, их виды, то как они работают, то как можно пополнить список, удалить конкретный элемент и как вывести весь список. Это было очень трудное, но такое интригующее задание.