Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №1**

**«Лабораторная работа №7»**

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

Выполнил: студент учебной группы

ИСПк-101-51-00

Клыков Денис Борисович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2025

В отчете должны отображаться:

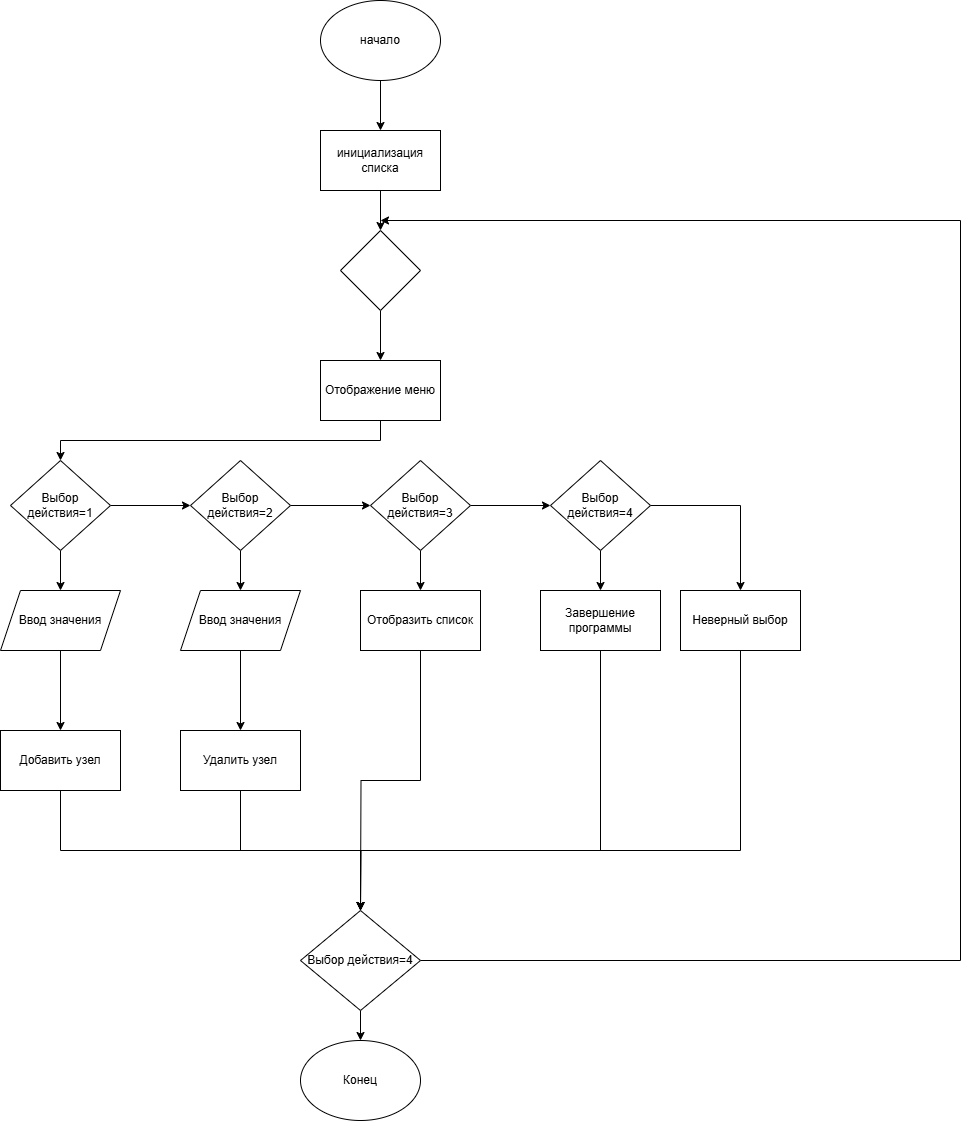
1. Цель работы
2. Формулировка задания (с вариантом)
3. Описание алгоритма и ответы на вопросы
4. Схема алгоритма с комментариями
5. Код программы
6. Результат выполнения программы
7. Вывод

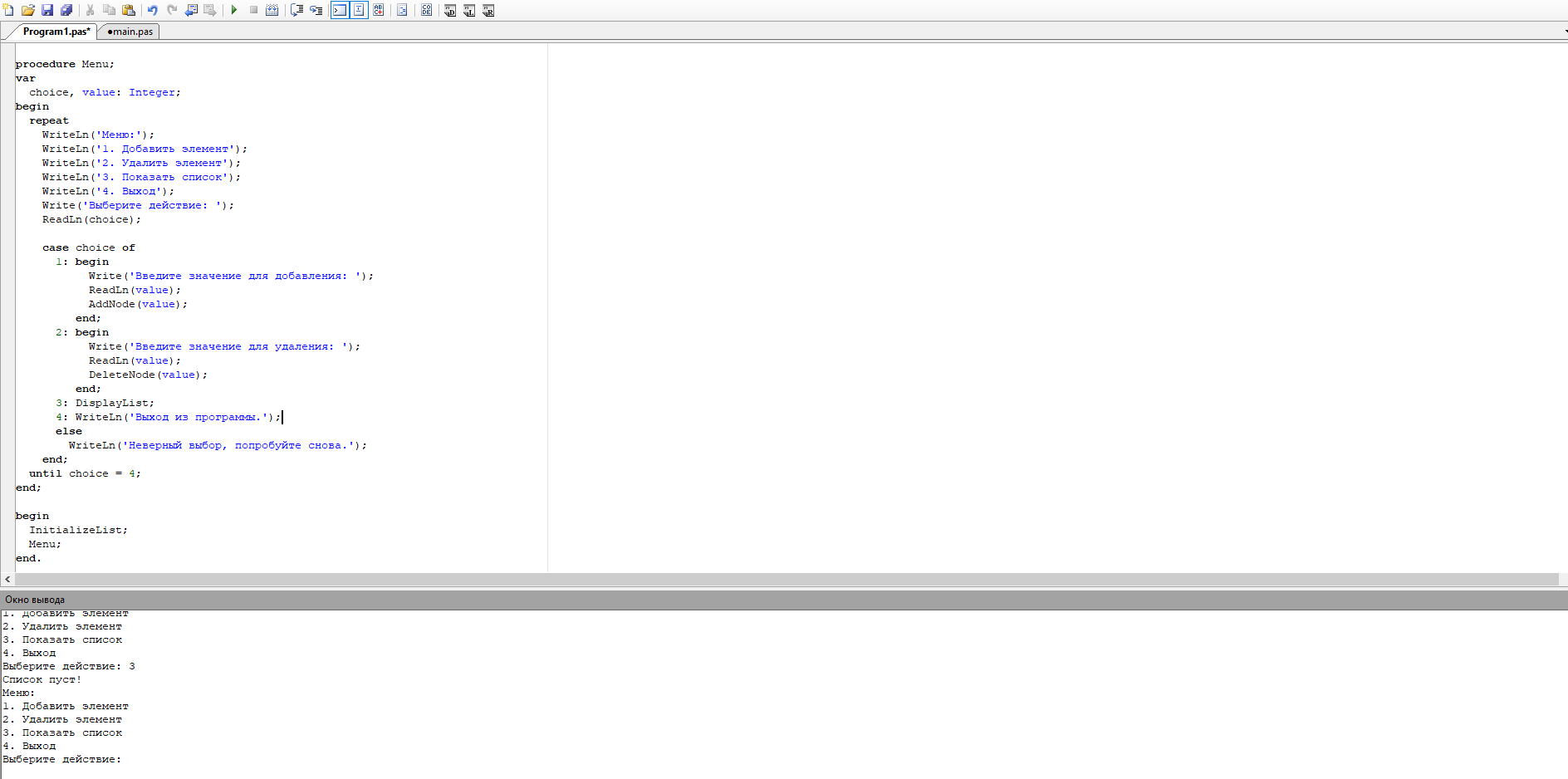
Вариант 6

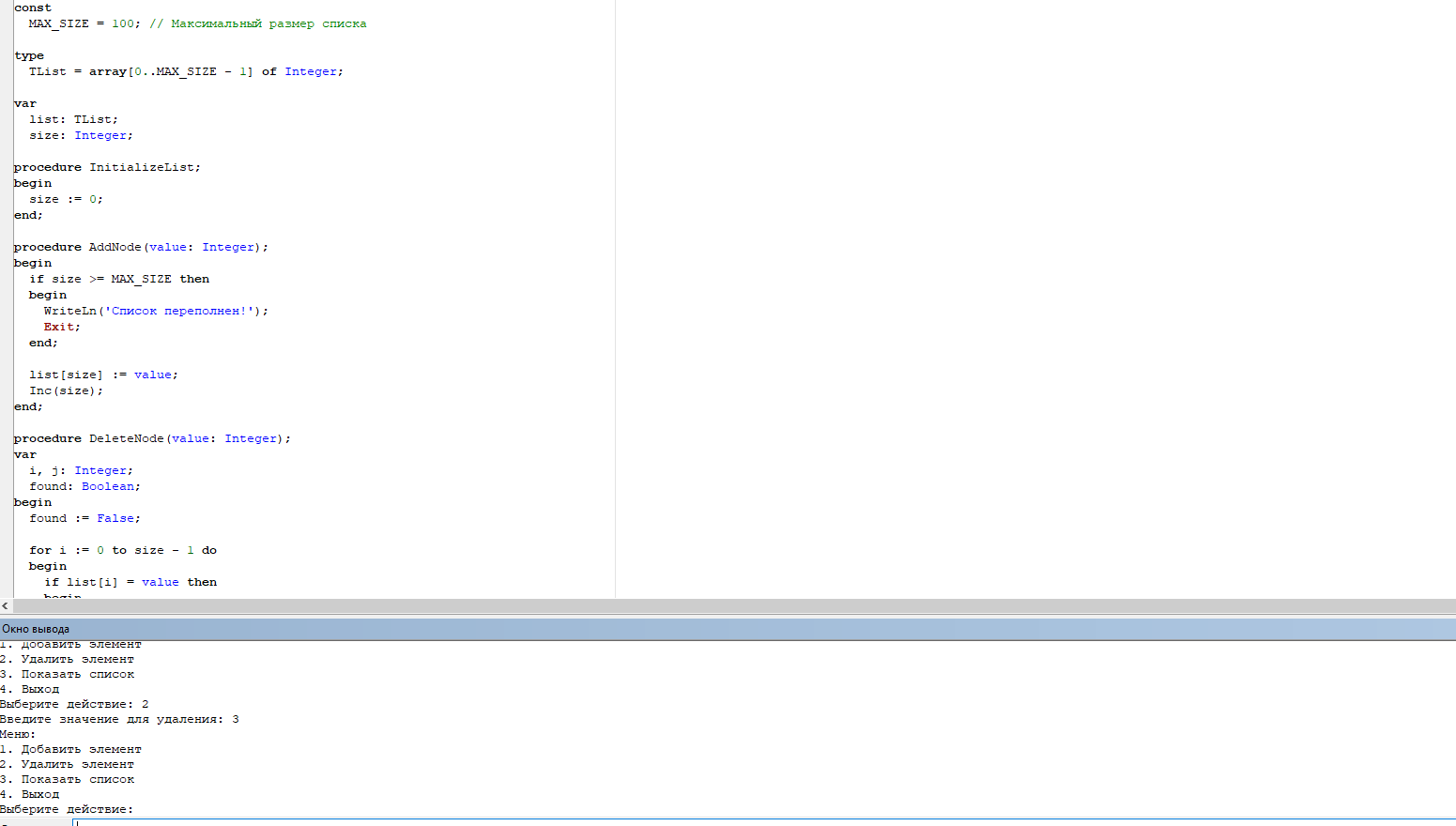
Цель работы: изучение принципов работы с базовыми структурами данных, получение навыков case-меню.

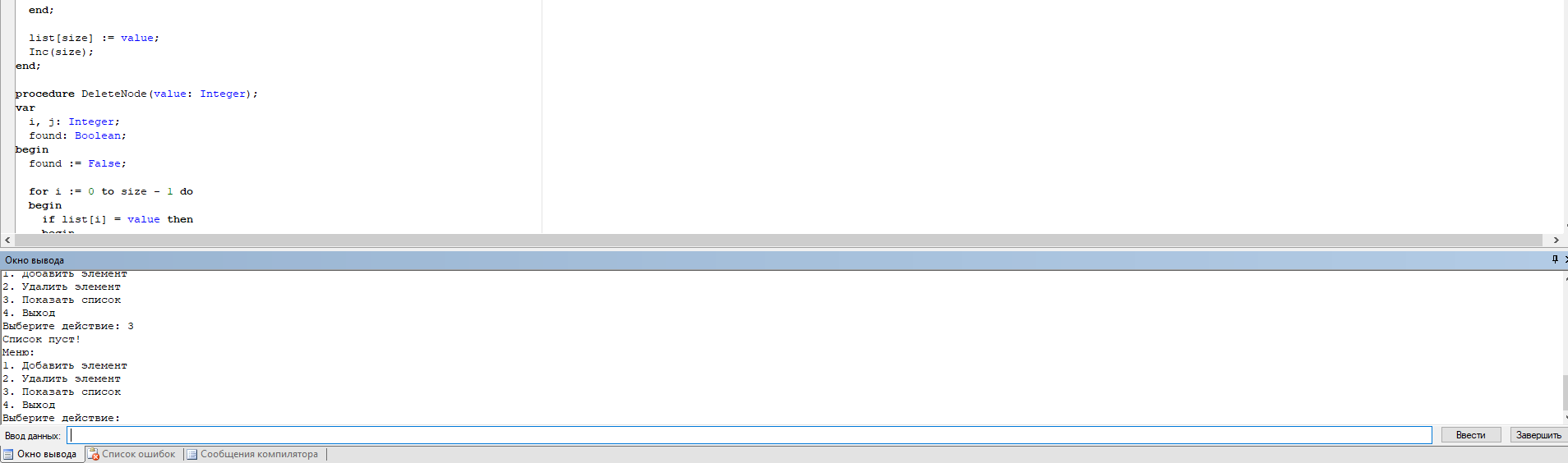
Задание:

1. Написать программу для работы со структурой данных "Односвязный список".
2. Структура данных должна быть реализована на основе статической памяти.
3. Работа со структурой должна осуществляться с помощью case-меню. Предусмотреть наглядную визуализацию содержимого структуры.









Код программы

﻿const

MAX\_SIZE = 100; // Максимальный размер списка

type

TList = array[0..MAX\_SIZE - 1] of Integer;

var

list: TList;

size: Integer;

procedure InitializeList;

begin

size := 0;

end;

procedure AddNode(value: Integer);

begin

if size >= MAX\_SIZE then

begin

WriteLn('Список переполнен!');

Exit;

end;

list[size] := value;

Inc(size);

end;

procedure DeleteNode(value: Integer);

var

i, j: Integer;

found: Boolean;

begin

found := False;

for i := 0 to size - 1 do

begin

if list[i] = value then

begin

found := True;

// Сдвигаем элементы влево

for j := i to size - 2 do

begin

list[j] := list[j + 1];

end;

Dec(size);

Break;

end;

end;

if not found then

WriteLn('Элемент не найден!');

end;

procedure DisplayList;

var

i: Integer;

begin

if size = 0 then

begin

WriteLn('Список пуст!');

Exit;

end;

Write('Содержимое списка: ');

for i := 0 to size - 1 do

begin

Write(list[i], ' ');

end;

WriteLn;

end;

procedure Menu;

var

choice, value: Integer;

begin

repeat

WriteLn('Меню:');

WriteLn('1. Добавить элемент');

WriteLn('2. Удалить элемент');

WriteLn('3. Показать список');

WriteLn('4. Выход');

Write('Выберите действие: ');

ReadLn(choice);

case choice of

1: begin

Write('Введите значение для добавления: ');

ReadLn(value);

AddNode(value);

end;

2: begin

Write('Введите значение для удаления: ');

ReadLn(value);

DeleteNode(value);

end;

3: DisplayList;

4: WriteLn('Выход из программы.');

else

WriteLn('Неверный выбор, попробуйте снова.');

end;

until choice = 4;

end;

begin

InitializeList;

Menu;

end.

Вывод:

Изучение принципов работы с базовыми структурами данных, такими как односвязные списки, и освоение навыков создания интерфейсов с использованием case-меню, предоставляет множество преимуществ и возможностей для разработчиков.

1. **Понимание структур данных**: Знание о том, как работают односвязные списки и другие структуры данных, помогает разработчикам эффективно организовывать и управлять данными. Это знание критически важно для оптимизации алгоритмов и повышения производительности программ.
2. **Динамическое управление памятью**: Работа с односвязными списками учит основам динамического выделения и освобождения памяти, что является важным аспектом программирования, особенно в языках, где управление памятью не автоматизировано.
3. **Разработка пользовательского интерфейса**: Создание case-меню позволяет разработчикам создавать интуитивно понятные интерфейсы для взаимодействия с пользователем.
4. **Логическое мышление и алгоритмическое проектирование**: Работа с структурами данных и реализация различных операций добавление, удаление, поиск.
5. **Гибкость и расширяемость**: Понимание основ работы с базовыми структурами данных позволяет разработчикам создавать более сложные и эффективные системы, комбинируя различные структуры и алгоритмы для решения конкретных задач.

В целом, изучение базовых структур данных и навыков работы с меню является важным шагом способного решать разнообразные задачи и создавать эффективные программные решения