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**

**«Основы событийно-ориентированного программирования»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-101-51-00

Парфенов Данил Васильевич

Преподаватель:

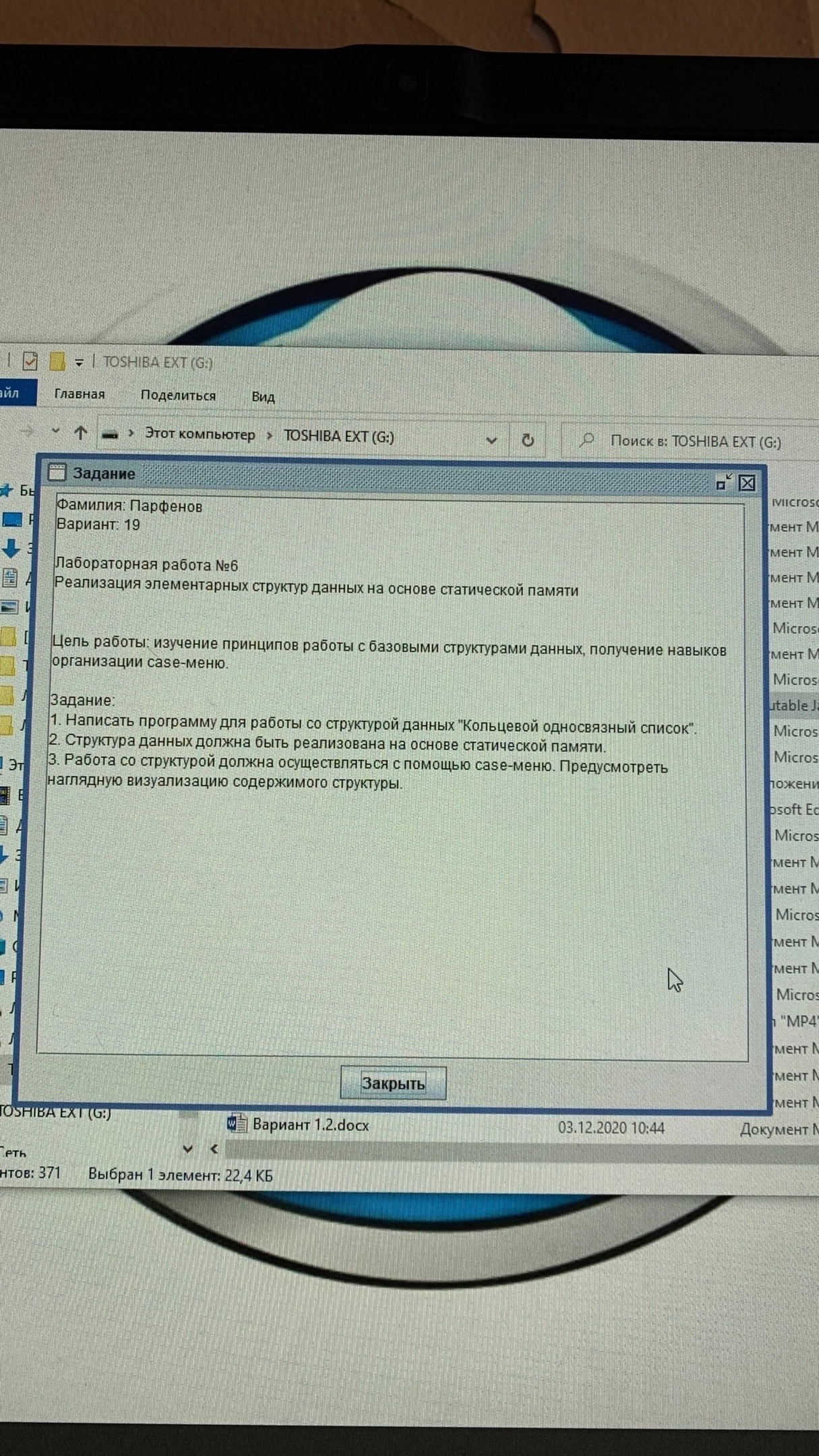
Сергеева Елизавета Григорьевна

Киров

2023

В отчете должны отображаться:

1. Цель работы: изучение принципов работы с базовыми структурами данных, получение навыков организации саѕе-меню
2. Скриншот задания (с вариантом)



1. Код программы

**program** CircularLinkedList;

**const**

Max = 100;

**type**

NodePtr = ^Node;

Node = **record**

Data: integer;

Next: NodePtr;

**end**;

**var**

List: **array** [1..Max] **of** Node;

Head, Tail: NodePtr;

Size: integer;

**procedure** Initialize;

**begin**

Head := nil;

Tail := nil;

Size := 0;

**end**;

**function** IsEmpty: boolean;

**begin**

IsEmpty := Size = 0;

**end**;

**function** IsFull: boolean;

**begin**

IsFull := Size = Max;

**end**;

**procedure** InsertAtHead(Data: integer);

**var**

NewNode: NodePtr;

**begin**

**if** IsFull **then**

**begin**

writeln('Ошибка список полон');

**exit**;

**end**;

**New**(NewNode);

NewNode^.Data := Data;

**if** IsEmpty **then**

**begin**

Head := NewNode;

Tail := NewNode;

NewNode^.Next := Head;

**end**

**else**

**begin**

NewNode^.Next := Head;

Head := NewNode;

Tail^.Next := Head;

**end**;

Inc(Size);

**end**;

**procedure** InsertAtTail(Data: integer);

**var**

NewNode: NodePtr;

**begin**

**if** IsFull **then**

**begin**

writeln('Ошибка: лист полный');

**exit**;

**end**;

**New**(NewNode);

NewNode^.Data := Data;

**if** IsEmpty **then**

**begin**

Head := NewNode;

Tail := NewNode;

NewNode^.Next := Head;

**end**

**else**

**begin**

Tail^.Next := NewNode;

Tail := NewNode;

Tail^.Next := Head;

**end**;

Inc(Size);

**end**;

**procedure** RemoveAtHead;

**var**

Temp: NodePtr;

**begin**

**if** IsEmpty **then**

**begin**

writeln('Ошибка: лист пуст');

**exit**;

**end**;

**if** Size = 1 **then**

**begin**

dispose(Head);

Head := nil;

Tail := nil;

**end**

**else**

**begin**

Temp := Head;

Head := Head^.Next;

Tail^.Next := Head;

dispose(Temp);

**end**;

Dec(Size);

**end**;

**procedure** RemoveAtTail;

**var**

Temp, Prev: NodePtr;

**begin**

**if** IsEmpty **then**

**begin**

writeln('Ошибка: пустой лист');

**exit**;

**end**;

**if** Size = 1 **then**

**begin**

dispose(Head);

Head := nil;

Tail := nil;

**end**

**else**

**begin**

Prev := Head;

**while** Prev^.Next <> Tail **do**

Prev := Prev^.Next;

Temp := Tail;

Tail := Prev;

Tail^.Next := Head;

dispose(Temp);

**end**;

Dec(Size);

**end**;

**procedure** DisplayList;

**var**

Temp: NodePtr;

**begin**

**if** IsEmpty **then**

**begin**

writeln('Список пуст');

**exit**;

**end**;

writeln('Список:');

Temp := Head;

**repeat**

writeln(Temp^.Data);

Temp := Temp^.Next;

**until** Temp = Head;

**end**;

**var**

Choice, Data: integer;

**begin**

Initialize;

**repeat**

writeln('1. Добавить в начало списка число');

writeln('2. Добавить в конец списка число');

writeln('3. Убрать число из начала списка');

writeln('4. Убрать число из конца списка');

writeln('5. Отобразить список');

write('Введите свой выбор: ');

readln(Choice);

**case** Choice **of**

1: **begin**

write('Введите данные: ');

readln(Data);

InsertAtHead(Data);

**end**;

2: **begin**

write('Введите данные: ');

readln(Data);

InsertAtTail(Data);

**end**;

3: RemoveAtHead;

4: RemoveAtTail;

5: DisplayList;

**else** writeln('Недействительный выбор');

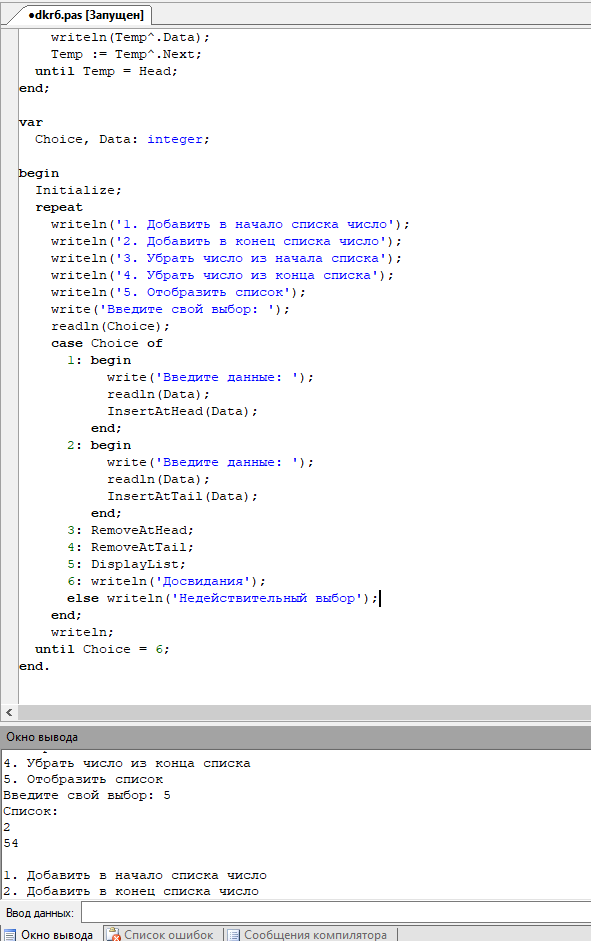
**end**;

writeln;

**until** Choice = 6;

**end**.

1. Результат выполнения программы



1. Вывод

В ходе выполнения данной домашней контрольной работы, были получены навыки работы со структурой данных “Кольцевой односвязный список”, были получены навыки работы с case-меню. В результате в программе реализация структуры данных происходит на основе динамической памяти. работа была выполнена в полном объеме. В процессе по программе был составлен отчёт.