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**

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

Выполнил: студент учебной группы

ИСПк-202-52-00

Колобов Александр Алексеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

Цель работы

Изучить принципы работы с базовыми структурами данных, получить навыки организации case-меню

Формулировка задания

Вариант 12.

1. Написать программу для работы со структурой данных “Кольцевой 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.

Описание алгоритма

Данный алгоритм создаёт кольцевой односвязный список, реализованный на статической памяти. Работа со структурой осуществляется с помощью case-меню и имеет визуализацию.

Схема алгоритма с комментариями

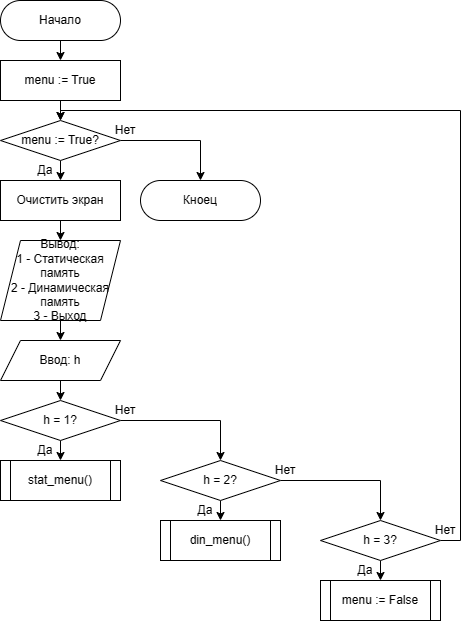


Рисунок 1 – Схема главного алгоритма

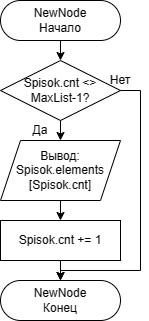


Рисунок 2 – Схема алгоритма процедуры NewNode

Рисунок 3 – Схема алгоритма процедуры stat\_view\_spisok

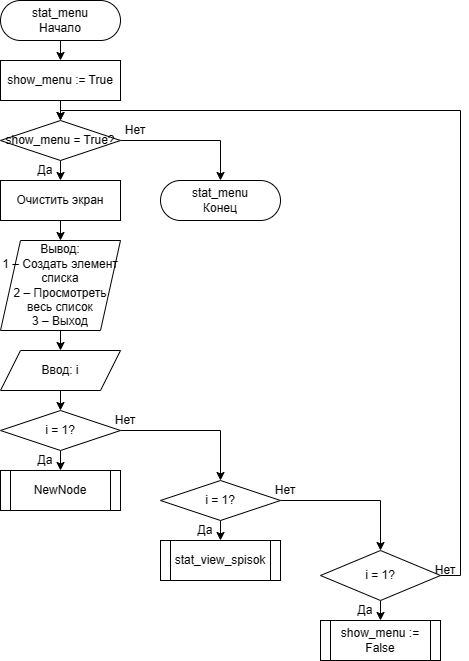
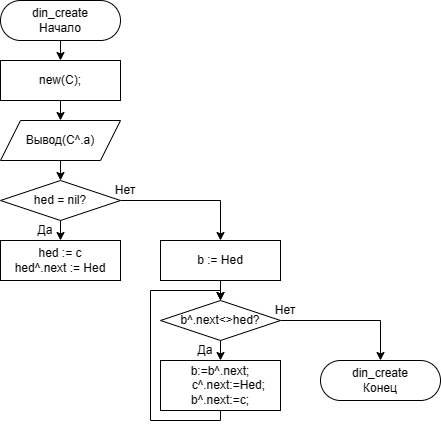


Рисунок 4 – Схема алгоритма процедуры stat\_menu

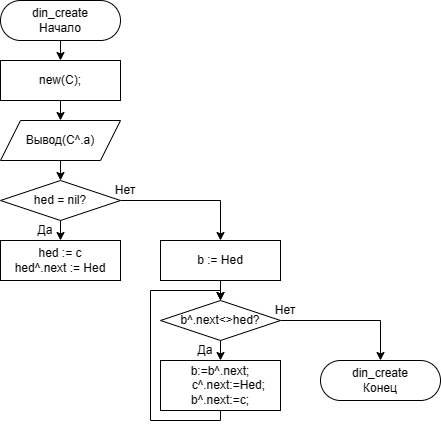


Рисунок 5 – Схема алгоритма процедуры din\_create

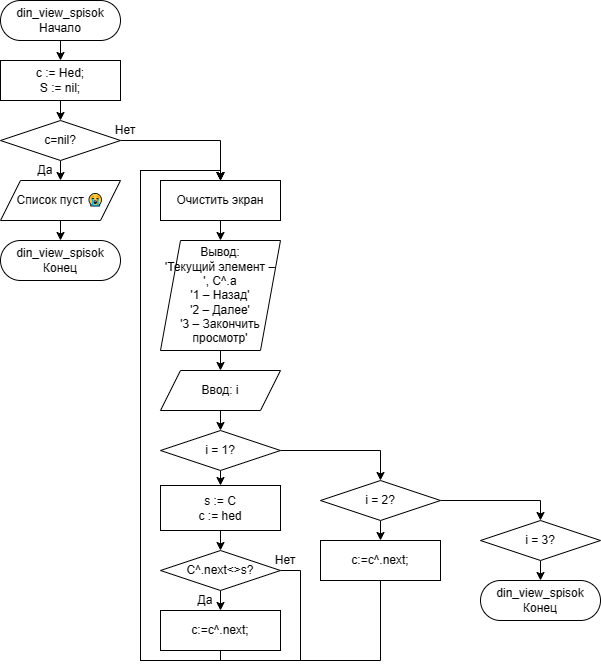


Рисунок 6 – Схема алгоритма процедуры din\_view\_spisok

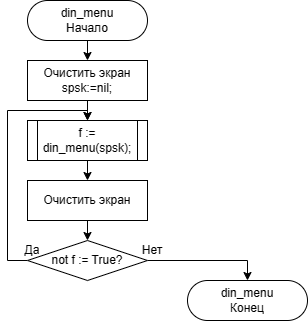


Рисунок 7 – Схема алгоритма процедуры din\_menu

Код программы

**Uses** crt;

**const** MaxList = 10;

**Type** StatNode=**record**

elements: **array**[0..MaxList-1] **of** integer;

cnt : integer;

**end**;

**var** show\_menu : boolean; Spisok : StatNode;

**procedure** NewNode();

**begin**

**if** Spisok.cnt <> MaxList-1 **then begin**

readln(Spisok.elements[Spisok.cnt]);

Spisok.cnt += 1;

**end**;

**end**;

**procedure** stat\_view\_spisok();

**var** i,c,selected,d : integer;

**begin**

i := 2;

**if** Spisok.cnt >= 3 **then while** True **do begin**

ClrScr;

writeln('Просмотр списка:');

write(' ');

**if** i = Spisok.cnt **then begin**

i := 0;

selected := Spisok.cnt-1;

write(Spisok.elements[Spisok.cnt-2], ' | ');

write(Spisok.elements[Spisok.cnt-1], ' | ');

write(Spisok.elements[i]);

**end else if** i = 1 **then begin**

selected := i - 1;

write(Spisok.elements[Spisok.cnt-1], ' | ');

write(Spisok.elements[i-1], ' | ');

write(Spisok.elements[i]);

**end else begin**

selected := i - 1;

write(Spisok.elements[i-2], ' | ');

write(Spisok.elements[i-1], ' | ');

write(Spisok.elements[i]);

**end**;

writeln(' → ');

writeln('2 – Далее по списку');

writeln('9 – Удалить элемент посередине');

writeln('0 – Выйти из просмотра');

readln(c);

**case** c **of**

2 : i += 1;

9 : **begin**

**for** d := selected **to** Spisok.cnt-1 **do begin**

Spisok.elements[d] := Spisok.elements[d+1]

**end**;

Spisok.elements[Spisok.cnt] := 0;

Spisok.cnt -= 1;

**end**;

0 : **break**;

**end**;

//readln();

**end else begin** ClrScr; writeln('Просмотр списка:'); **for** i := 0 **to** Spisok.cnt-1 **do** write(Spisok.elements[i],' '); writeln; writeln('Нажмите Enter для выхода'); readln; **end**;

**end**;

**procedure** stat\_menu();

**Var** i:char;

**begin**

show\_menu := True;

**while** show\_menu **do**

**begin**

ClrScr;

WriteLn('1 – Создать элемент списка ');

Writeln('2 – Просмотреть весь список ');

WriteLn('3 – Выход ');

//i:=readkey;

readLn(i);

**case** i **of**

'1':NewNode();

'2':stat\_view\_spisok();

'3':show\_menu := False;

**end**;

**end**;

**end**;

**Type** Pint=^intr;

intr=**record**

a:integer;

next:Pint;

**end**;

**Procedure** din\_create(**Var** Hed:Pint);

**Var** C,B:Pint;

a:integer;

**begin**

**new**(C);

readln(C^.a);

**if** hed=nil **then**

**begin**

hed:=c;

hed^.next:=Hed;

**end else**

**begin**

b:=hed;

**while** b^.next<>hed **do**

b:=b^.next;

c^.next:=Hed;

b^.next:=c;

**end**;

**end**;

**Procedure** din\_view\_spisok(**Var** Hed:Pint);

**Var** C,S:Pint;

i:integer;

**begin**

c:=Hed;

S:=nil;

**if** c=nil **then**

**begin**

WriteLn('Список пуст 😭');

readLn;

**exit**;

**end**;

**repeat**

ClrScr;

WriteLn('Текущий элемент – ', C^.a);

WriteLn('1 – Назад');

WriteLn('2 – Далее');

WriteLn('3 – Закончить просмотр');

readLn(i);

**case** i **of**

1:**begin**

s:=C;

c:=hed;

**While** C^.next<>s **do**

c:=c^.next;

**end**;

2:c:=C^.next;

3:**break**;

**end**;

**until** False;

**end**;

**Function** din\_menu(**Var** Sp:pint) : boolean;

**Var** i :char;

**begin**

din\_menu := true;

WriteLn('1 – Создать элемент списка ');

Writeln('2 – Просмотреть весь список ');

WriteLn('3 – Выход ');

//i:=readkey;

readLn(i);

**case** i **of**

'1':din\_create(Sp);

'2':din\_view\_spisok(sp);

'3':din\_menu:=false;

**end**;

**end**;

**procedure** din\_menu();

**Var** spsk:Pint; F:boolean;

**Begin**

ClrScr;

spsk:=nil;

**repeat**

f := din\_menu(spsk);

clrscr;

**until not** F;

**end**;

**var** menu : boolean; h : char;

**begin**

menu := True;

**while** menu **do**

**begin**

ClrScr;

WriteLn('1 – Статическая память');

Writeln('2 – Динамическая память');

WriteLn('3 – Выход ');

readLn(h);

**case** h **of**

'1': stat\_menu();

'2': din\_menu();

'3': menu := False;

**end**;

**end**;

**end**.

Результат выполнения программы

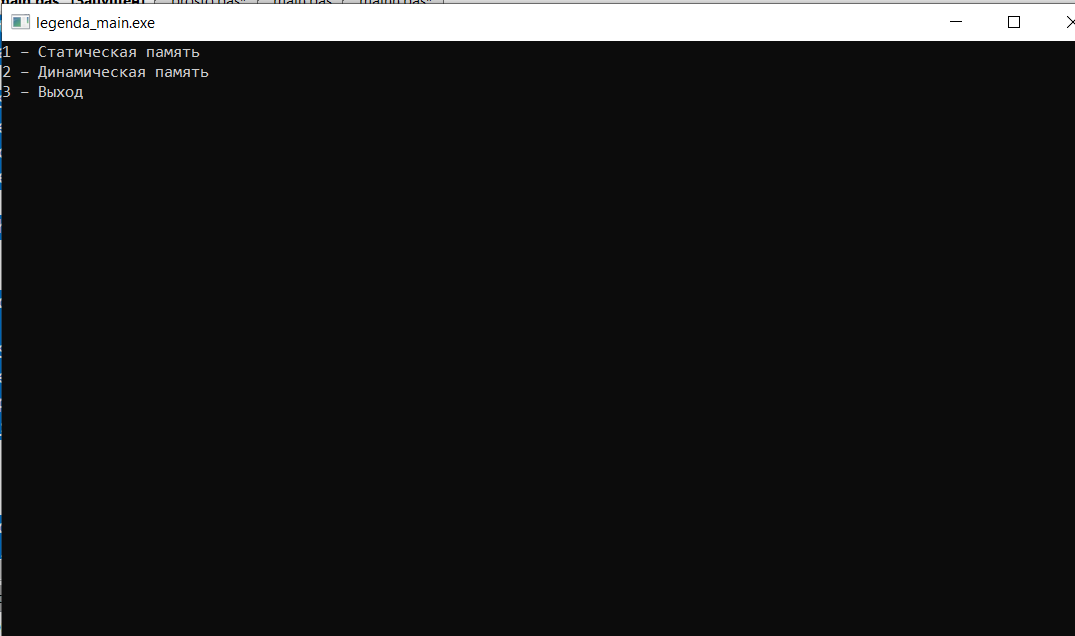


Рисунок 8 - Результат выполнения программы

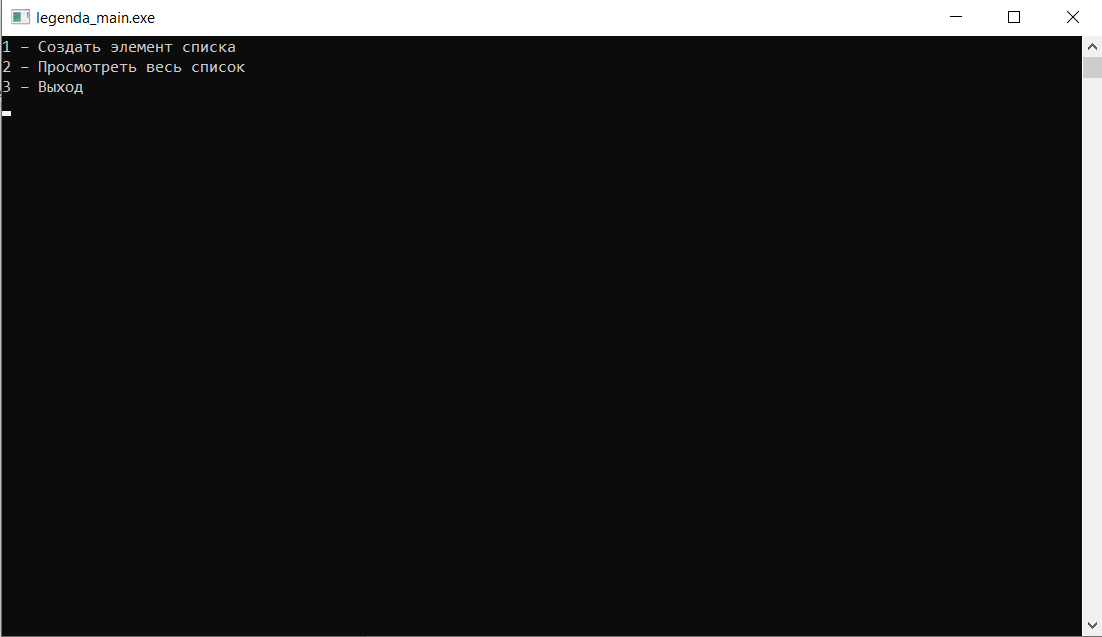


Рисунок 9 – Выбор памяти

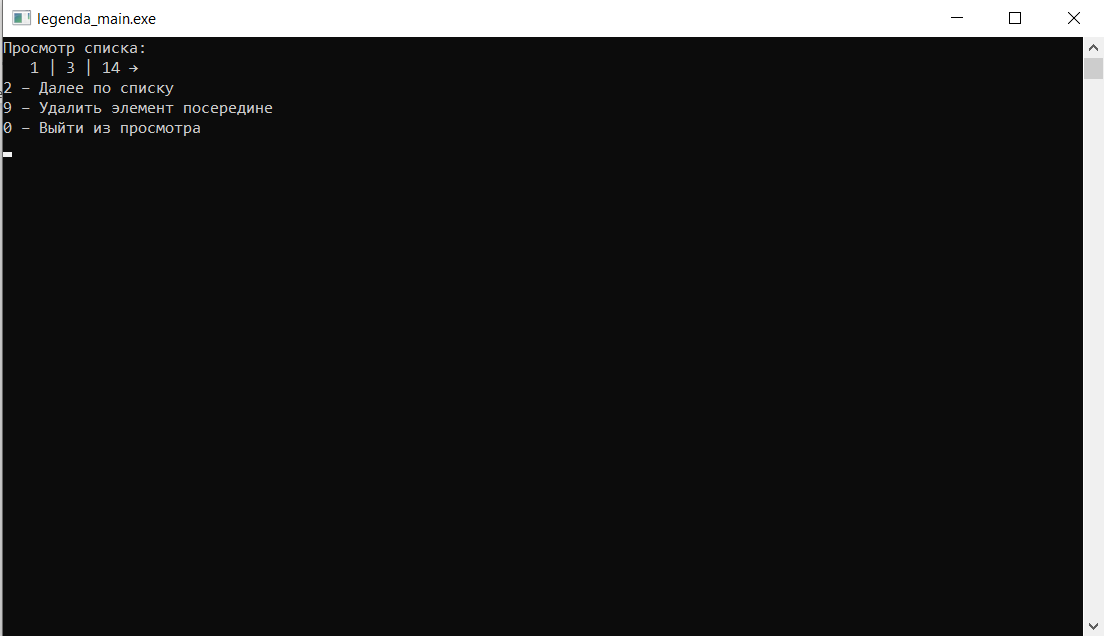


Рисунок 10 – Просмотр статической памяти

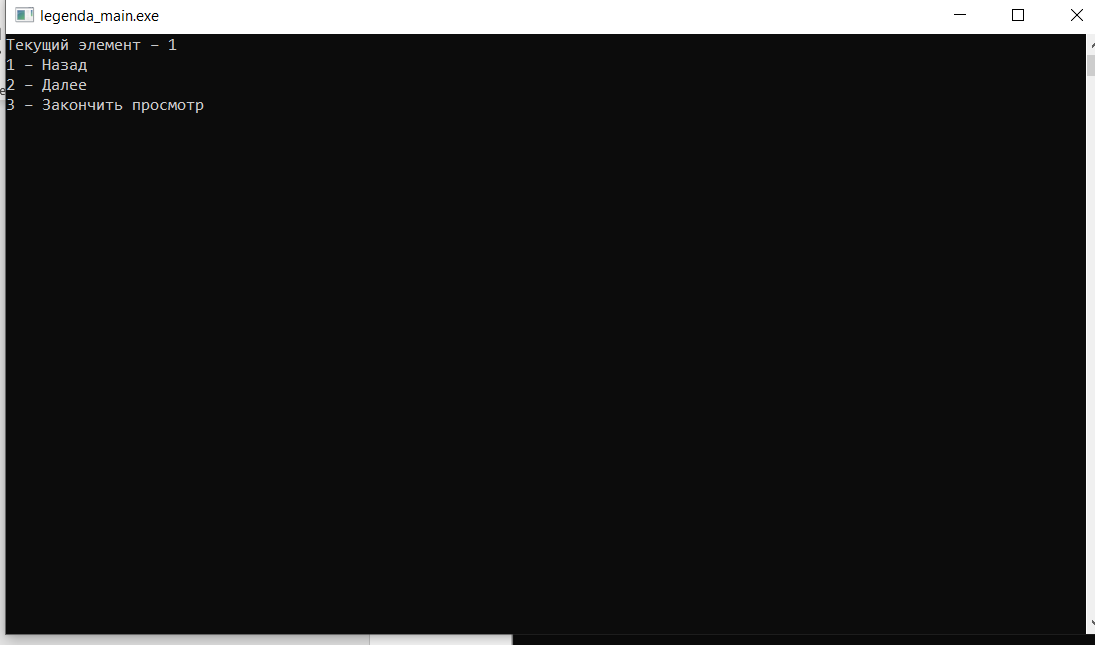
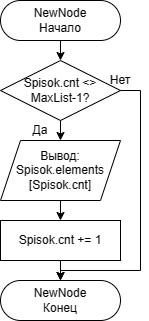


Рисунок 11 – Просмотр динамической памяти

Вывод

В ходе выполнения данной работы мы смогли применить изученный материал лабораторных занятий на практике, процедуры, условный оператор, цикл while, repeat, for, создание переменных и операторов ввода и вывода, а также мы построили схему алгоритма с помощью бесплатного онлайн-сервиса app.diagrams.net.

В результате выполнения данной работы получили базовые сведения о наиболее известных алгоритмах сортировки, изучили принципы работы с текстовыми файлами, а также выполнили данное нам практическое задание, использовав изученный материал на практике.

Таким образом, выполнение домашней контрольной работы №7 принесло нам ценный опыт и знания, которые будут полезны в нашем дальнейшем образовании. Мы научились применять определённые базовые конструкции языка программирования, а также разрабатывать схему алгоритма для решения задачи. Этот опыт будет полезен нам будущем для решений новых задач и работой над реальными проектами.