Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Новосибирский государственный технический университет»

**ОтчЁт ПО ЛАБОРАТОРНой работе №2**

**“СПИСКИ”**

|  |  |
| --- | --- |
| Выполнил:  Студент Ерлин Матвей Владимирович  Группа ДТ-160  «12» февраля 2022г. | Проверил:  Руководитель от НГТУ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(Ф. И. О.)*  Балл \_\_\_\_\_\_\_\_\_\_\_\_\_, ECTS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  *(«отлично», «хорошо»,*  *«удовлетворительно», «неуд.»)* |

**Задание**

* Определить количество элементов списка, значения которых совпадают со значениями в первом и последнем элементах списка.
* Создать список L3 из элементов, входящих и в список L1 и в список L2, вернуть длину нового списка.

**Тестовые данные (Для первой функции)**

List\* head = NULL; List\* head = NULL;

head = Insert\_first(3, head); head = Insert\_first(5, head);

head = Add\_last (2, head); head = Insert\_first(3, head);

head = Insert\_first(3, head); head = Insert\_first(2, head);

head = Add\_last (1, head); head = Insert\_first(4, head);

head = Insert\_first(1, head); head = Add\_last (5, head);

head = Insert\_first(5, head); head = Insert\_first(6, head);

head = Insert\_first(5, head);

head = Insert\_first(11, head);

head = Insert\_first(23, head);

head = Insert\_first(2, head);

head = Add\_last (3, head);

head = Insert\_first(1, head);

Вывод-6 Вывод-3

**Текст программы на языке (С++) (Для первой функции)**

// Поиск количества элементов списка, значения которых совпадают с первым и послединм элементом списка

int Search\_FirstAndLast(List\* head)

{

List\* p = head;

int first = p->value; //Взятие 1 значения

int last, sum\_elem =0;

while (p->next->next != NULL) //Поиск последнего значения

p = p->next;

last = p->next->value;

List\* z = head;

while (z != NULL) //Если список не пустой

{

if (z->value == first or z->value == last)

{

sum\_elem += 1;

z = z->next;

}

else

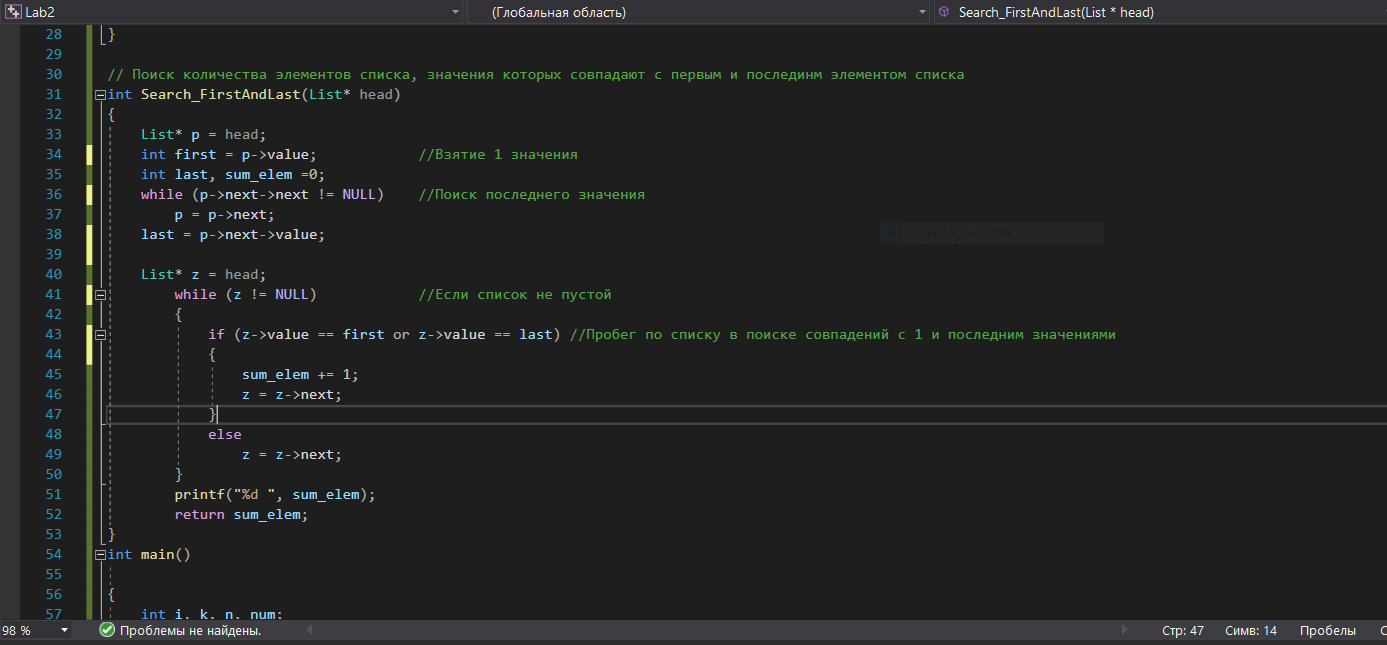
z = z->next;

}

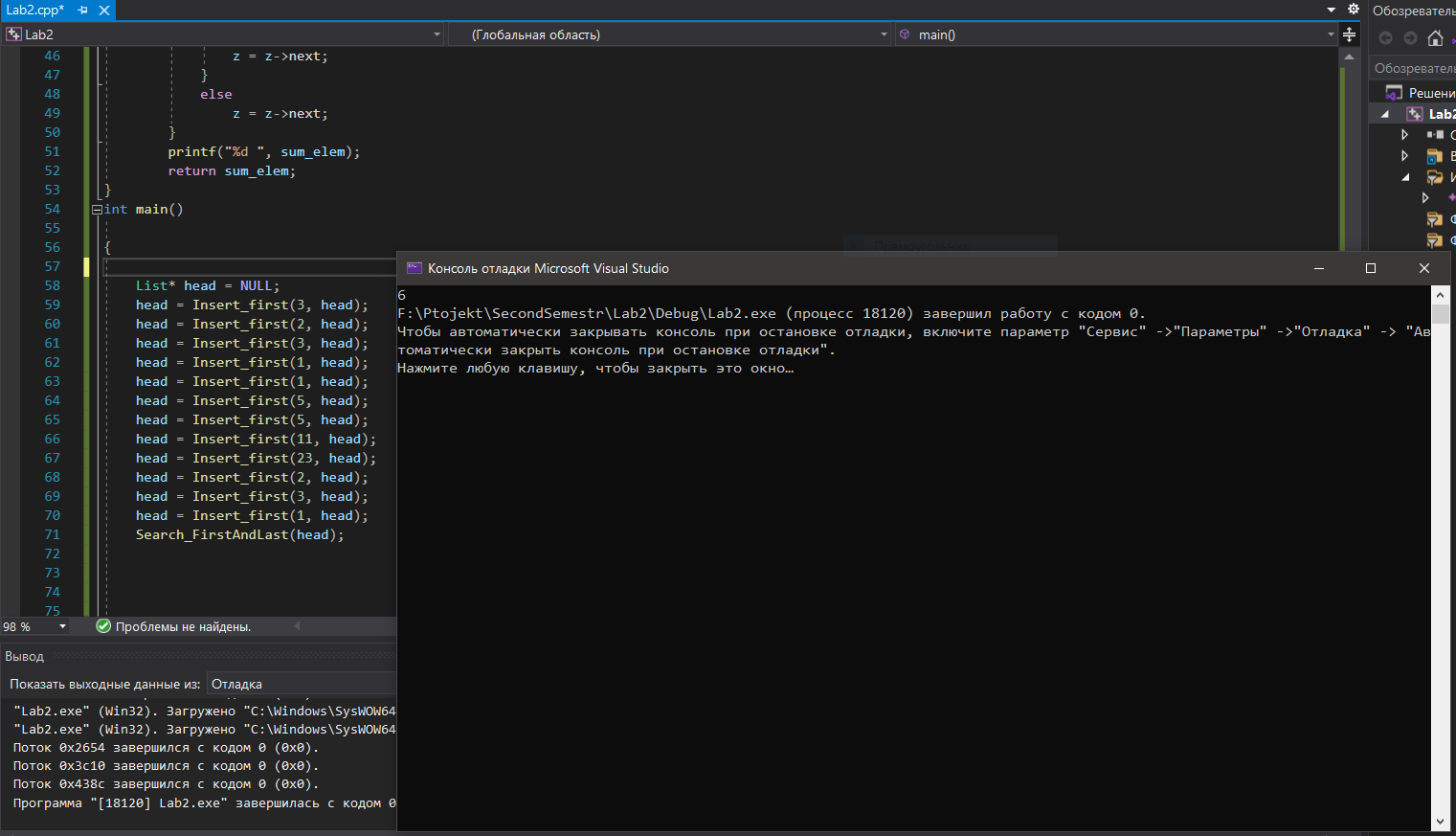
printf("%d ", sum\_elem);

return sum\_elem;

}



**Пример работы программы (Для первой функции)**



**Тестовые данные (Для второй функции)**

List\* head = NULL; List\* head = NULL;

head = Insert\_first(3, head); head = Insert\_first(5, head);

head = Add\_last (2, head); head = Insert\_first(3, head);

head = Insert\_first(3, head); head = Insert\_first(2, head);

head = Add\_last (1, head); head = Insert\_first(4, head);

head = Insert\_first(1, head); head = Add\_last (5, head);

head = Insert\_first(5, head); head = Insert\_first(6, head);

head = Insert\_first(5, head);

head = Insert\_first(11, head);

head = Insert\_first(23, head);

head = Insert\_first(2, head);

head = Add\_last (3, head);

head = Insert\_first(1, head);

Вывод-18

**Текст программы на языке (С++) (Для второй функции)**

// Cовмещение 2 списков

int AmongUs(List\* head, List\* headen)

{

int lng=0;

List\* TWO = NULL; // инициализация 3 списка

List\* p = headen; // взятие всех значений из 1 списка

while (p != NULL)

{

TWO = Insert\_first(p->value, TWO);

lng += 1;

p = p->next;

}

List\* s = head; // взятие всех значений из 2 списка

while (s != NULL)

{

TWO = Insert\_first(s->value, TWO);

lng += 1;

s = s->next;

}

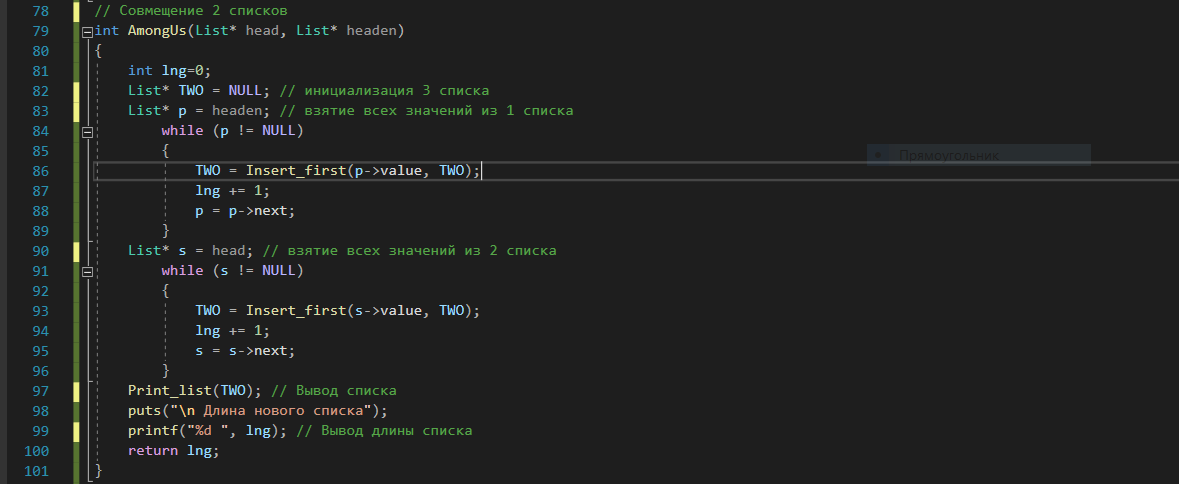
Print\_list(TWO); // Вывод списка

puts("\n Длина нового списка");

printf("%d ", lng); // Вывод длины списка

return lng;

}



**Пример работы программы (Для второй функции)**

