Министерство науки и высшего образования РФ

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЁТ**

по лабораторной работе № 3

по курсу «Логика и основы алгоритмизации в инженерных задачах»

на тему «Динамические списки»

Выполнили ст. гр. 22ВВВ2:

Рябов С. И.

Китаев Я. Е.

Аргаткин А. А.

Приняли:

Митрохин М.А.

Акифьев И.В

ПЕНЗА 2023

**Цель работы**

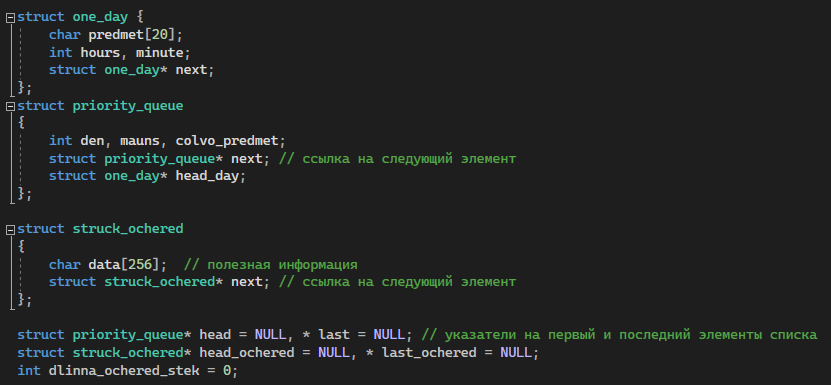
Изучить принципы работы динамических списков, реализовать разные структуры данных в программе. Добавить разные функции взаимодействия со списками.

**Лабораторное задание**

1. Реализовать приоритетную очередь, путём добавления элемента в список в соответствии с приоритетом объекта (т.е. объект с большим приоритетом становится перед объектом с меньшим приоритетом).
2. \* На основе приведенного кода реализуйте структуру данных *Очередь*.
3. \* На основе приведенного кода реализуйте структуру данных *Стек*
4. Также было разработана структура данных, содержащая расписание и сортирующая его.

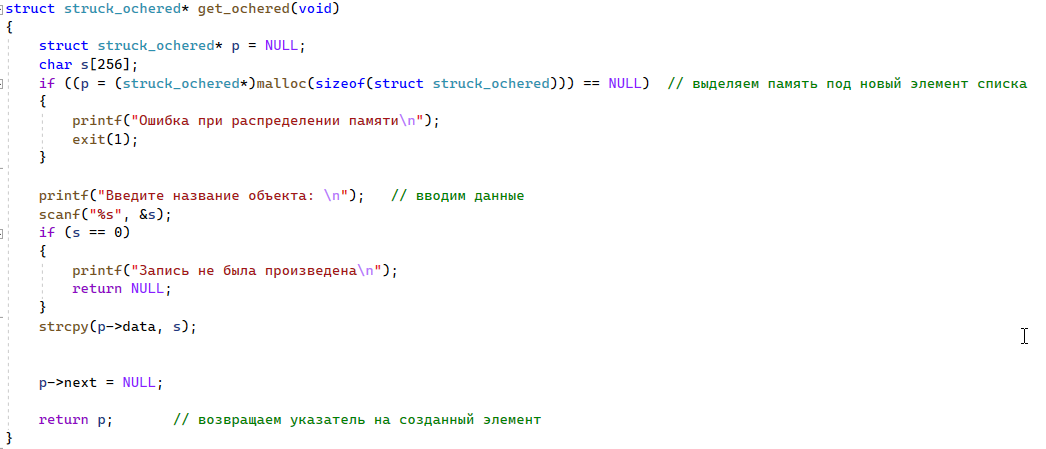
**Описание программы**

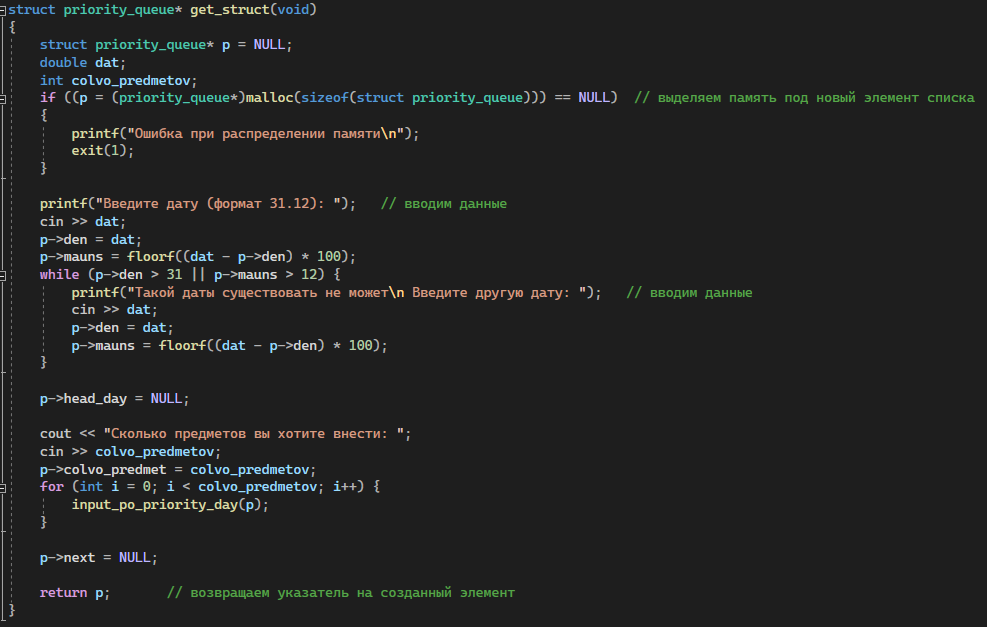
Программа предназначена для работы с разными списками, многие функции для них одинаковые, поэтому они имеют общие части.

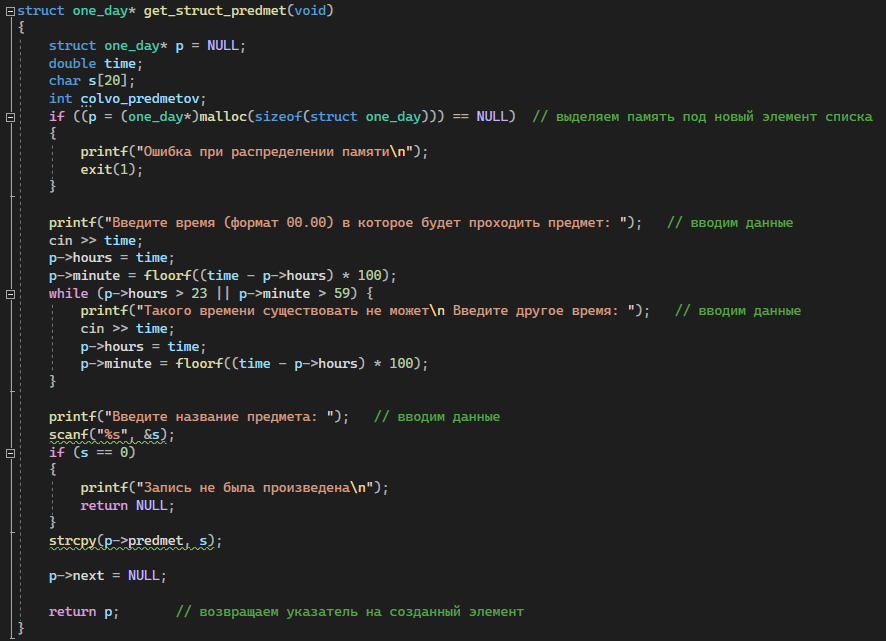
Сначала мы создаем структуры, которые будут использоваться в работе программы, а также добавляем необходимые указатели, созданные для содержания ссылок на ключевые элементы структур. 

**Описание функций:**

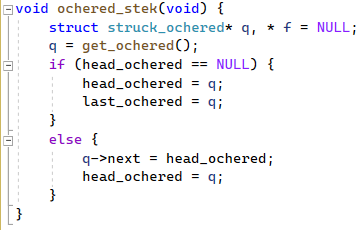
Функции **get\_ochered, get\_struct, get\_struct\_predmet** позволяют создавать новые элементы списка. Также они обрабатывает некоторые исключения.



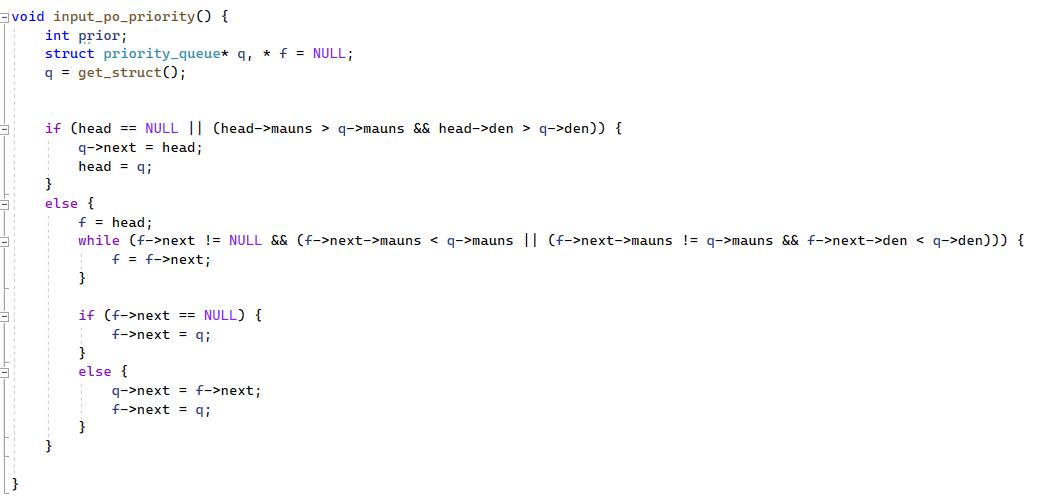
****

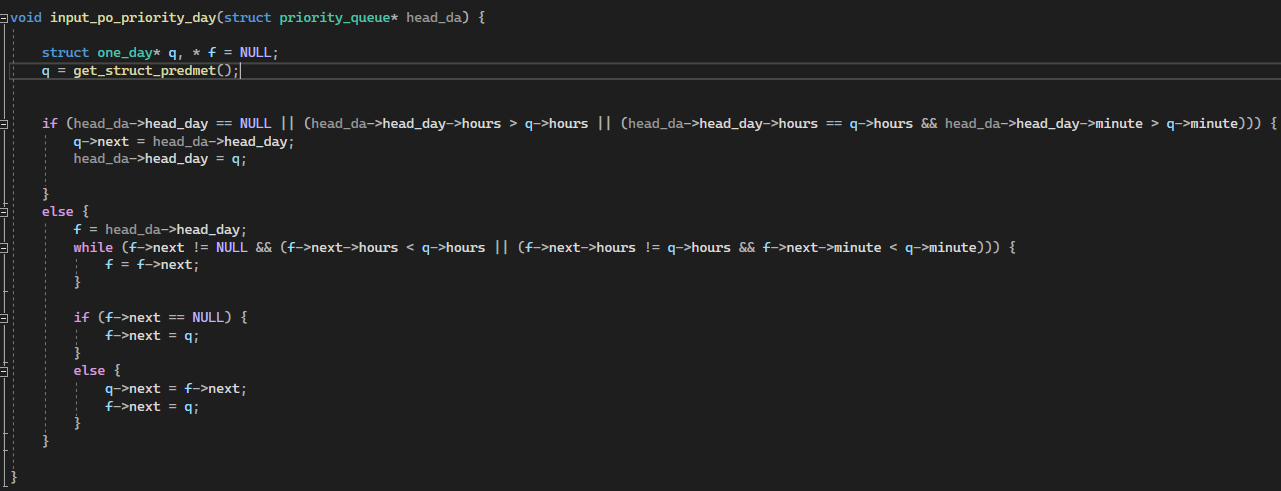
****

Функция **ohered\_stek** служит для добавления новых элементов в список и в стек.

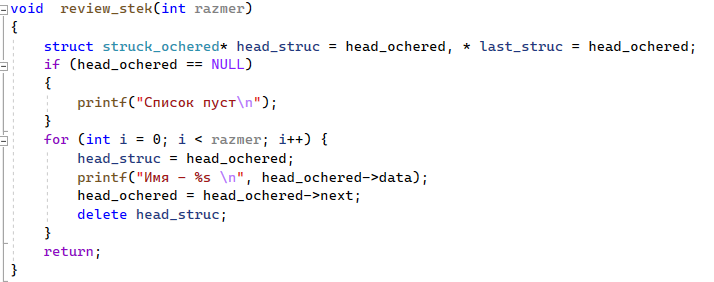


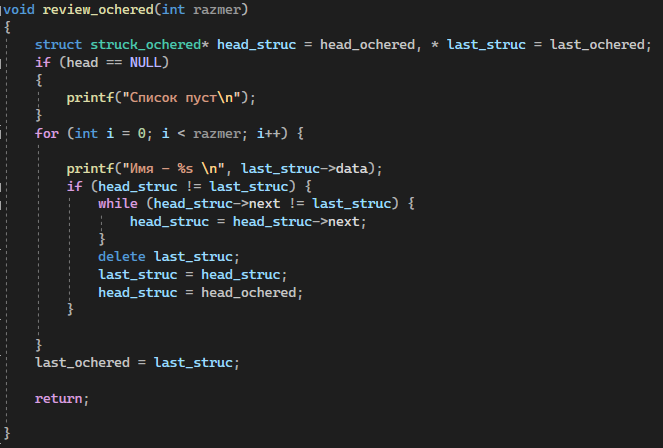
Функции **input\_po\_prioritety и input\_po\_priority\_day** служат для добавления новых элементов в очередь с приоритетом. Они выполняют сравнение, чтобы прогнать список и найти место для элемента. Элемент с самым наименьшим значением приоритета заносится в начало списка, при извлечении достается элемент с наибольшим значением.



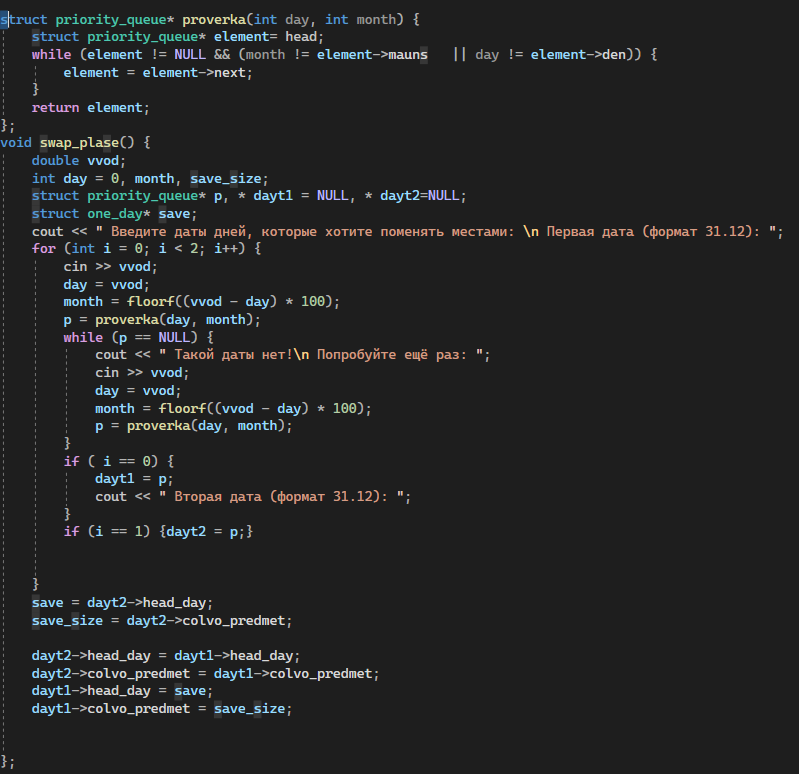
****

Для каждой структуры была создана своя функция “review”, позволяющая выбирать количество элементов, которые мы хотим вывести из списка. Из стека выводится с конца, из списка выводится с начала из очереди с приоритетом – элемент с самым высоким приоритетом.



****

Функции **proverka** и **swap\_plase** предназначены для того, чтобы ментяь расписане в 2 днях местами. Пользователь вводит даты дней, расписание которых хочет поменять, после чего функция **proverka** находит эти дни ( если они существуют, если нет, то ползователю предлагается ввести другую дат). В конце указатели на главные элементы струкруты **get\_struct\_predmet** содержащиеся в этих днях меняются местами.

****

**Основная функция main**

Был реализован пользовательский интерфейс, позволяющий выбирать тип данных, а также выбирать желаемую операцию. Из списков в программе: стек, список и приоритетная очередь. В каждом варианте есть выбор: добавить определённое количество элементов, достать определённое количество элементов или завершить работу со структурами данных.

**Вывод:** Научились работать со сложными структурами данных и реализовывать разные виды списков. В ходе работы была разработана программа, позволяющая пользователю выбирать тип желаемого списка, и проводить с ним базовые операции: вводить и выводить из него любое количество данных.