Министерство высшего образования и науки Российской Федерации

Санкт-Петербургский Политехнический Университет Петра Великого

—

Институт компьютерных наук и кибербезопасности

**ЛАБОРАТОРНАЯ РАБОТА № 2**

«**База данных самолётов аэропорта**»

по дисциплине «Структуры данных»

Выполнила Шевчук Н. Е.

студентка гр. 5131001/30003

<*подпись*>

Преподаватель

ст. преподаватель Семьянов П. В.

<*подпись*>

Санкт-Петербург

2024

[1. Цель работы 3](#_Toc166758541)

[2. Постановка задачи 3](#_Toc166758542)

[3. Теоретические исследования 3](#_Toc166758543)

[4. Описание решения 4](#_Toc166758544)

[4.1 Ключевые структуры данных 4](#_Toc166758545)

[4.2 Функции управления данными 4](#_Toc166758546)

[4.3 Логика программы 5](#_Toc166758547)

[5. Тестирование и результаты работы программы 9](#_Toc166758548)

[6. Выводы 9](#_Toc166758549)

[Приложение 10](#_Toc166758550)

1. Цель работы

Целью работы является разработка консольного приложения с функционалом безразмерной базы данных, написанную на языке программирования C, включающую в себя все типы данных.

2. Постановка задачи

* 1. Выбрать подходящую структуру данных.
  2. Реализовать следующие функции:
     + Сохранение данных в файле.
     + Считывание данных из файла.
     + Добавление самолёта в базу данных через терминал.
     + Вставка и удаление самолёта.
     + Редактирование данных отдельного самолёта.
     + Поиск самолёта по номеру модели.
  3. Обеспечить оптимальное использование памяти.

3. Теоретические исследования

Список — это структура данных, хранящая элементы в линейном порядке и позволяющая эффективно добавлять и удалять элементы в любом месте последовательности. Элементы в двусвязном списке имеют ссылки на предыдущий и следующий, что экономит память при хранении больших объёмов данных.

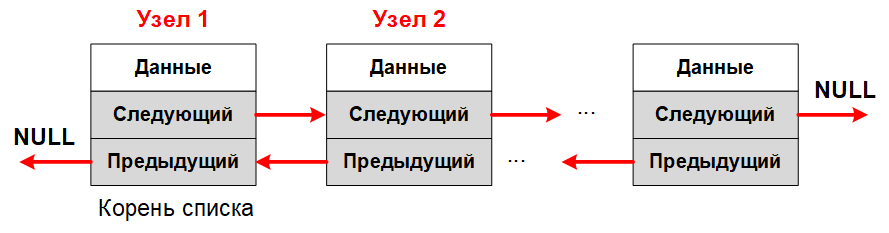


Рис. 1 Схема двусвязного списка

4. Описание решения

**4.1 Ключевые структуры данных**

* 1. struct plane - эта структура содержит информацию для каждого самолёта в программе, например: название производителя, дату сборки, максимальное время в полёте и другое.
  2. struct date - данная структура содержит в себе информацию о дате сборке самолёта.
  3. union special\_mark - объединение, хранящее особую отметку о самолёте, это может быть символ или цифра.
  4. enum plane\_class — перечисляемый тип данных для хранения информации о классах самолётов (пассажирский, грузовой, военный, специальный).

**4.2 Функции управления данными**

* 1. rec\_enter - функция предлагает пользователю ввести всю информацию для нового самолёта. Она обрабатывает чтение данных из консоли и присваивает их соответствующим полям в структуре клиента.
  2. new\_rec - функция выделяет память для новой структуры самолёта с помощью malloc, затем вызывает функцию rec\_enter для заполнения данных и возвращает указатель на вновь созданного самолёта.
  3. add - функция берет указатель на голову связанного списка самолётов и добавляет новый самолёт в конец. Она обрабатывает связывание указателей prev и next нового самолёта соответствующим образом, чтобы сохранить структуру списка.
  4. print - функция принимает указатель на самолёт и печатает всю его информацию в консоли. Сюда входит название производителя, номер модели, дата сборки, максимальное время в полёте, класс самолёта и особая отметка.
  5. rremove - функция берет указатель на самолёт и удаляет его из связанного списка. Она обновляет указатели prev и next соседних узлов, чтобы сохранить целостность списка.
  6. insert - функция принимает указатель на самолёт и вставляет новый самолёт после указанного в связный список. Она соответствующим образом корректирует указатели prev и next.
  7. find - функция принимает указатель на голову связанного списка самолётов и номер модели в качестве входных данных. Она выполняет поиск по списку и возвращает указатель на самолёт с совпадающим номером модели. Если такой самолёт не найден, возвращается NULL.

**4.3 Логика программы**

При запуске программа выводит пользователю меню из 8 пунктов выбора.

Чтение и запись происходит в/из файл(-а) plane\_base.txt.

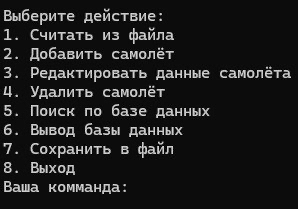


Рис. 2. Главное меню программы

При добавлении самолёта в базу данных программа просит ввести все необходимые данные.

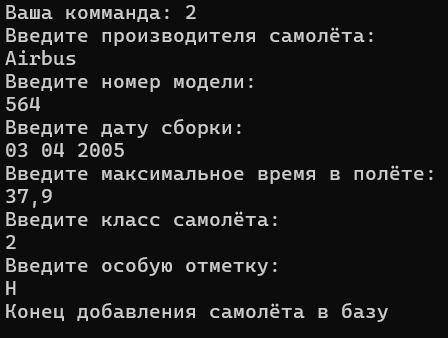


Рис.3. Ввод самолета

Удачно выполненный поиск самолёта в базе данных, позволяющий ознакомиться с данными, выполнить редактирование или удаление представлен на рис. 4.

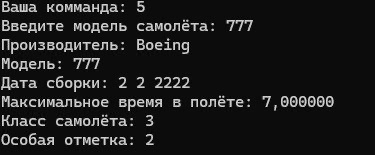


Рис. 4. Удачно выполненный поиск

При выборе пункта вывода базы данных раздельными блоками выводится информация о каждом клиенте.

Последний пункт выходит из программы. Для повторного входа необходимо заново скомпилировать программу.

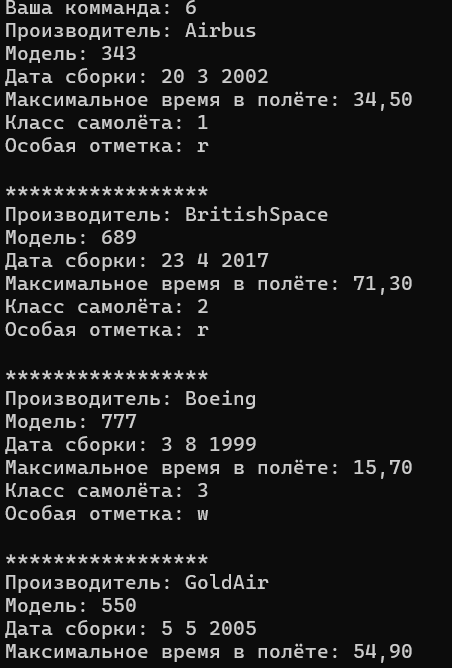


Рис.7. Вывод базы данных

5. Тестирование и результаты работы программы

Файл, из которого была считана база данных и результат работы программы представлен на рис. 8, после добавления нового самолета.

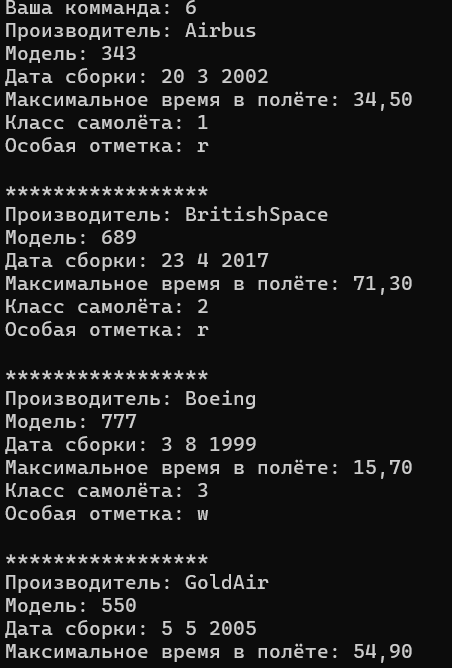


Рис.8. Результат работы программы

6. Выводы

В ходе работы была реализована база данных самолётов аэропорта, были освоены навыки работы с двусвязными списками, динамической памятью и различными структурами данных в языке программирования Си.