**Лаба1.1**

Для хранения книг используется контейнер std::vector, что позволяет динамически управлять размером библиотеки. Подсчет количества книг каждого типа реализован двумя способами: с использованием оператора switch и конструкции if-else. Оба подхода продемонстрировали одинаково корректную работу, но switch может быть предпочтительнее при большом количестве типов.

Программа успешно компилируется, корректно выполняется и не содержит утечек памяти, так как используется только автоматическое управление памятью (без динамического выделения с new).

**Лаба1.2**

В данной работе реализована программа на языке C++, которая позволяет загрузить список автомобилей из текстового файла и вывести их на экран. Каждая машина представляется объектом класса Car, содержащего марку (make) и модель (model). Объекты хранятся в векторе в классе Garage, обеспечивающем управление их жизненным циклом.

Главной задачей было реализовать надёжную и читаемую систему хранения объектов и считывания данных с последующим выводом. Загрузка автомобилей производится из файла cars.txt, каждая строка которого содержит марку и модель. При чтении данных используется std::istringstream для разбиения строки на составляющие, что делает парсинг удобным и гибким.

Особое внимание уделено управлению памятью. Объекты Car создаются динамически с помощью оператора new, а затем добавляются в вектор указателей в объекте класса Garage. Чтобы избежать утечек памяти, в деструкторе Garage все объекты корректно удаляются при завершении программы. Такой подход требует осторожности, так как ошибки в управлении ресурсами могут привести к утечкам или повреждению памяти.

Тем не менее, можно отметить, что использование "сырых" указателей (raw pointers) не является лучшей практикой в современном C++. Было бы лучше использовать std::unique\_ptr или std::shared\_ptr, что обеспечило бы автоматическое управление памятью и исключило необходимость ручного удаления объектов.

Также важно, что программа корректно обрабатывает ситуацию, когда файл не удаётся открыть: выводится сообщение об ошибке, и выполнение продолжается безопасным образом.