

Лабораторная работа № 5 «Работа с массивами структур. Исследование методов сортировки массивов»

Вариант №262

Введение

В ходе выполнения лабораторной работы должны быть разработаны:

1. Программа № 1, осуществляющая в диалоговом режиме ввод, сортировку и вывод последовательности данных, которая представляется в виде массива структур.
2. Программа № 2, осуществляющая таймирование сортировки массивов.

Программы № 1 и № 2 должны реализовывать поддержку работы с тремя алгоритмами сортировок: с двумя из состава индивидуального задания и с реализацией алгоритма быстрой сортировки из состава стандартной библиотеки — функцией `qsort()`.

Программа № 1 должна реализовывать следующую функциональность:

1. Ввод массива:
 - (a) из стандартного потока ввода потока («с клавиатуры»);
 - (b) из текстового файла (с возможностью указания имени файла);
 - (c) из бинарного файла (с возможностью указания имени файла).
2. Вывод массива:
 - (a) в стандартный поток вывода («на экран»);
 - (b) в текстовый файл (с возможностью указания имени файла);
 - (c) в бинарный файл (с возможностью указания имени файла).
3. Сортировка массива с возможностью выбора пользователем через диалоговое меню:
 - (a) алгоритма сортировки (одного из трёх);
 - (b) поля структуры, по которому осуществляется сортировка;
 - (c) направления сортировки (по убыванию или по возрастанию).

Программа № 2 должна реализовывать таймирование сортировки с возможностью выбора пользователем через диалоговое меню:

1. алгоритма сортировки (одного из трёх);
2. поля структуры, по которому осуществляется сортировка;
3. направления сортировки (по убыванию или по возрастанию);

4. количества элементов в генерируемых массивах;
5. количества генерируемых массивов.

Примечания:

1. Взаимодействие программ с пользователем должно быть выстроено с помощью иерархического диалогового меню.
2. Программа № 1 должна осуществлять проверку корректности данных, вводимых пользователем, и, в случае ошибок, выдавать соответствующие сообщения, после чего продолжать работу.
3. Программа № 1 должна осуществлять проверку корректности данных, считываемых из файлов. В случае ошибок формата файла — выдавать соответствующие сообщения в стандартный поток вывода ошибок и продолжать работу, считая что ввод не был выполнен успешно. В случае некорректных данных для конкретных записей — выдавать соответствующие сообщения в поток ошибок, после чего продолжать работу, игнорируя данные записи.
4. Для работы с данными, формат которых описан в индивидуальном задании, должен быть разработан собственный составной тип данных — структура.
5. Для работы с данными, структура которых описана в индивидуальном задании, должен быть разработан формат хранения в текстовом файле.
6. Для работы с данными, структура которых описана в индивидуальном задании, должен быть разработан формат хранения в бинарном файле.
7. Работа с текстовыми файлами должна осуществляться при помощи функций стандартной библиотеки `fopen()`, `fclose()`, `fprintf()`, `fscanf()`.
8. Работа с бинарными файлами должна осуществляться при помощи функций стандартной библиотеки `fopen()`, `fclose()`, `fread()`, `fwrite()`.
9. Логически законченные части алгоритма решения задачи должны быть оформлены в виде отдельных функций с параметрами. Использование глобальных переменных не допускается.
10. Исходные коды программы должны быть логичным образом разбиты на несколько файлов.
11. Программа должна корректным образом работать с памятью, для проверки необходимо использовать соответствующие программные средства, например: `valgrind` (при тестировании и отладке программы ее необходимо запускать командой вида `valgrind ./lab5`, а при анализе производительности — `./lab5`).

Отчётность по выполнению лабораторной работы должна включать:

1. Блок-схемы алгоритмов сортировки массива.
2. Исходные коды программ.
3. Тестовые наборы для иллюстрации работы программ.
4. Результаты таймирования, содержащие таблицы, графики зависимости времени выполнения сортировок от количества сортируемых элементов и аргументированные выводы об оценке сложности рассмотренных алгоритмов сортировки и её совпадении с теоретическими ожиданиями.

Индивидуальное задание

Структура данных

Автомобиль:

- марка (строка длиной до 16 символов, которая может включать в себя только буквы, дефис и пробелы);
- ФИО владельца (строка произвольной длины);
- пробег (дробное число, соответствующее величине пробега в тыс. км).

Алгоритмы сортировки

1. Пузырьковая сортировка (Bubble sort).
2. Двухсторонняя сортировка выбором (Double selection sort).