

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

В ходе выполнения лабораторной работы должны быть разработаны:

1. Программа, осуществляющая чтение массива из текстового файла, его сортировку и запись результата в другой текстовый файл.
2. Программа, осуществляющая таймирование сортировки массива.

Программы 1 и 2 должны реализовывать поддержку работы с тремя алгоритмами сортировок: двумя из индивидуального задания и с реализацией алгоритма быстрой сортировки из состава стандартной библиотеки — функцией `qsort()`.

Программа 1 должна через аргументы командной строки получать следующие параметры:

1. Алгоритм сортировки (через опцию).
2. Поле, по которому осуществляется сортировка (через опцию).
3. Направление сортировки (через опцию).
4. Имя входного файла (через обязательный позиционный аргумент).
5. Имя выходного файла (через обязательный позиционный аргумент).

Программа 2 должна через аргументы командной строки получать следующие параметры:

1. Алгоритм сортировки (через опцию).
2. Поле, по которому осуществляется сортировка (через опцию).
3. Направление сортировки (через опцию).
4. Количество элементов в генерируемых массивах (через обязательный позиционный аргумент).
5. Количество генерируемых массивов (через обязательный позиционный аргумент).

Примечания:

1. Программы должны осуществлять обработку аргументов командной строки с помощью функции `getopt()` из состава стандартной библиотеки.
2. Программа должна осуществлять проверку корректности данных, получаемых от пользователя через аргументы командной строки. В случае ошибок — выдавать соответствующие сообщения в поток вывода ошибок и завершаться с соответствующим кодом ошибки.

3. Программа должна осуществлять проверку корректности данных, считываемых из текстового файла. В случае ошибок формата файла — выдавать соответствующие сообщения в поток вывода ошибок и завершаться с соответствующим кодом ошибки. 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, формат которых описан в индивидуальном задании, должен быть разработан простой формат хранения данных в текстовом файле.
6. Работа с текстовыми файлами должна осуществляться при помощи функций стандартной библиотеки `fopen()`, `fclose()`, `fprintf()`, `fscanf()`.
7. Логически законченные части алгоритма решения задачи должны быть оформлены в виде отдельных функций с параметрами. Использование глобальных переменных не допускается.
8. Исходные коды программы должны быть логичным образом разбиты на несколько файлов.
9. 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 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.