

Лабораторная работа № 4

Реализация алгоритма сортировки

Цель работы: формирование практических навыков выбора и разработки эффективного алгоритма сортировки с учетом особенностей конкретной задачи и структур обрабатываемых данных.

Задачи: реализовать алгоритм сортировки и исследовать его вычислительную сложность.

1. ЗАДАНИЕ

В лабораторной работе необходимо реализовать заданный алгоритм (см. таблицу ниже) и исследовать его вычислительную сложность. Алгоритм выбирается в соответствии с вариантом задания, полученным от преподавателя.

Вариант	Алгоритм
1.	Сортировка выбором. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
2.	Сортировка слиянием. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
3.	Сортировка расчёской. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
4.	Сортировка перемешиванием. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
5.	Сортировка Шелла. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
6.	Пирамидальная сортировка. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.

7.	Сортировка вставками. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
8.	Сортировка подсчётом. Проиллюстрировать работу сортировки на примере цветов. Сгенерировать массив цветов (RGB) и на цветовой палитре продемонстрировать процесс работы сортировки на каждой итерации (программно). На примере HSL модели.
9.	Сортировка выбором. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
10.	Сортировка слиянием. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
11.	Сортировка расчёской. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
12.	Сортировка перемешиванием. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
13.	Сортировка Шелла. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
14.	Пирамидальная сортировка. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
15.	Сортировка вставками. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.

16.	Карманная сортировка. Проиллюстрировать работу сортировки на примере цветов. Создать BMP файл (каждый пиксель задается случайным цветом). Продемонстрировать сортировку цветов создав новый BMP файл. На примере HSL модели.
17.	Быстрая сортировка разбиение по Хоару. Проиллюстрируйте работу сортировки с использованием горизонтальных линий. Продемонстрировать процесс работы разбиения на каждой итерации (программно)
18.	Быстрая сортировка разбиение по Ломуто. Проиллюстрируйте работу сортировки с использованием горизонтальных линий. Продемонстрировать процесс работы разбиения на каждой итерации (программно)

Порядок работы:

1. реализовать заданный алгоритм, убедиться в его корректности;
2. произвести эмпирический анализ производительность реализованного алгоритма сортировки, сделать выводы.

Указание: Сравнение выполнять аналогично исследованию в лабораторной работе № 1: для 10–20 разных значений размера входных данных провести несколько (4–7) измерений, отбросить максимальные результаты и усреднить.

2. ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Назовите критерий, на основе которого алгоритм сортировки может быть отнесен к классу алгоритмов сортировки сравнением.
2. Приведите примеры алгоритмов, относящихся к классу алгоритмов сортировки сравнением.
3. Сформулируйте теорему о нижней оценке алгоритмов сортировки сравнением.
4. Дайте определение устойчивости алгоритма сортировки.
5. Опишите идею алгоритма сортировки вставками, дайте оценку времени его работы и оценку требуемой памяти.
6. Опишите идеи перечисленных ниже алгоритмов сортировки, дайте оценку времени их работы и оценку требуемой памяти, назовите особенности алгоритмов (например, устойчивость, худший случай, характер данных и т. п.).
 - Сортировка вставками;
 - сортировка слиянием;
 - быстрая сортировка;
 - пирамидальная сортировка;
 - сортировка подсчетом;
 - карманная сортировка;
 - поразрядная сортировка.
7. Приведите примеры гибридных алгоритмов сортировки.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет по лабораторной работе должен содержать:

1. титульный лист установленного образца с указанными ФИО студента и номером варианта;
2. постановку задачи;
3. описание реализуемого алгоритма и его характеристик;
4. результаты сравнения (в табличном и графическом виде) и выводы;
5. в приложении – исходный код реализованного алгоритма.