

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ М. В. ЛОМОНОСОВА
ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ

ОТЧЕТ ПО ЗАДАНИЮ №1

«Методы сортировки»

Вариант 2 / 1 / 1 / 3

Выполнил:
студент 105 группы
Яшин К. Е.

Преподаватель:
Смирнов А. В.

Москва
2018

Содержание

Постановка задачи	2
Результаты экспериментов	3
Структура программы и спецификация функций	4
Отладка программы, тестирование функций	5
Анализ допущенных ошибок	6
Список цитируемой литературы	7

Постановка задачи

Основная задача данного проекта есть реализация двух алгоритмов сортировки массива - сортировки Шелла и сортировки пузырьком. Второстепенными задачами являются

- оценка алгоритмов сортировки,
- составление статистической таблицы сравнения алгоритмов на различных данных,
- оценка скорости работы алгоритмов.

Требуется написать и проанализировать алгоритм сортировки 64-разрядных целых чисел по неубыванию. Также необходимо реализовать функцию, которая заполняет сортируемый массив произвольными числами, что позволит более точно оценить сложность получившегося алгоритма.

Результаты экспериментов

n	Параметр	Номер сгенерированного массива				Среднее значение
		1	2	3	4	
10	Сравнения	45	45	45	45	45
	Перемещения	0	45	25	27	24
100	Сравнения	4950	4950	4950	4950	4950
	Перемещения	0	4950	2577	2497	2506
1000	Сравнения	499500	499500	499500	499500	499500
	Перемещения	0	499500	245955	254276	194682
10000	Сравнения	49995000	49995000	49995000	49995000	49995000
	Перемещения	0	49995000	25134700	25146664	25069091

Таблица 1: Результаты работы сортировки пузырьком

В данной таблице представлены статистические данные работы алгоритма пузырьковой сортировки. По результатам тестирования программы можно произвести оценку работы алгоритма и выявить зависимость времени выполнения программы от объема сортируемых данных. Как известно из [1], алгоритм сортировки пузырьком имеет сложность $O(n^2)$, число сравнений и перестановок в худшем случае равно $N(N-1)/2$, в лучшем число перестановок равно 0, а сравнений - $N(N-1)/2$. Это подтверждают и наши табличные значения.

n	Параметр	Номер сгенерированного массива				Среднее значение
		1	2	3	4	
10	Сравнения	22	27	28	27	26
	Перемещения	0	13	10	7	7.5
100	Сравнения	503	668	904	792	717
	Перемещения	0	260	459	332	1051
1000	Сравнения	8006	11716	14760	14650	12283
	Перемещения	0	4600	7135	7129	4716
10000	Сравнения	120005	172578	260784	266657	205006
	Перемещения	0	62560	145795	151662	90004

Таблица 2: Результаты работы сортировки Шелла

В данной таблице представлены статистические данные работы алгоритма сортировки Шелла. По результатам тестирования программы можно произвести оценку работы алгоритма и выявить зависимость времени выполнения программы от объема сортируемых данных. Как известно из [1], среднее время работы алгоритма зависит от длин промежутков — d , на которых будут находиться сортируемые элементы исходного массива ёмкостью N на каждом шаге алгоритма. Существует несколько подходов к выбору этих значений. В данной программе был использован подход, предложенный самим Дональдом Шеллом. В худшем случае, сложность алгоритма составляет $O(n^2)$, что подтверждается. Также из таблицы видно преимущество сортировки Шелла над сортировкой пузырьком.

Структура программы и спецификация функций

Функции, которые используются в программе

- `void *good (int, long long int*)` - функция генерации уже отсортированного массива,
- `void *bad (int, long long int*)` - функция генерации наихудшего массива,
- `void *generate (int, long long int*)` - функция генерации случайного массива,
- `void *bubblesort (int, long long int*)` - функция, которая сортирует массив пузырьком,
- `void ShellsSort(int, long long int*)` - функция, которая сортирует массив сортировкой Шелла.

Опишем структуру программы. Функция `main(void)` генерирует массив из N элементов, где значение N вводится с клавиатуры. Далее, с помощью функций, описанных выше, программа последовательно заполняет массив тремя разными способами. После каждого из заполнений следует 2 сортировки и вывод на экран числа сравнений и обменов.

Опишем принцип работы сортировок.

Сортировка пузырьком - стандартный алгоритм сортировки, который работает по данному принципу: берется элемент массива и последовательно сравнится с другими соседними элементами массива, в случае нарушения порядка, элементы меняются местами и проход продолжается.

Сортировка Шелла - это модернизированный алгоритм сортировки вставками, который сортирует массив по подмножествам, которые определяются последовательностью расстояний между элементами, и на этом шаге алгоритм выполняет сортировку вставками, которая сортирует массив.

Отладка программы, тестирование функций

Прежде всего, сортировка Шелла была понята с помощью ручки и листа бумаги. Это помогло структурировать идеи и написать сам код. Далее каждая из функций отдельно тестировалась с помощью отладчика и стандартных функций языка. Была проверена генерация массива, сортировки на различных данных. С помощью отладчика и заранее известных асимптотических оценок была проверена правильность написания алгоритмов и корректность выходных данных.

Анализ допущенных ошибок

Во время работы над заданием было допущено несколько ошибок. При написании сортировки Шелла было несколько случаев выхода за границы массива, которые были устранены с помощью отладчика и вывода на экран данных. Кроме этого, с помощью бумаги и отладчика было проверено, правильно ли программа считает нужные нам данные для таблиц.

Список литературы

- [1] Кнут Д. Искусство программирования для ЭВМ. Том 3. — М.: Мир, 1978.