

Отчет по заданию №1
“Методы сортировки”
Вариант 2/1/3+1

Выполнил:
Анисимов Н. Д.

Москва
2021

Содержание

Постановка задач	2
Результаты экспериментов	3 - 4
Структура программы и спецификация функций	5
Отладка программы, тестирование функций	6
Анализ допущенных ошибок	7
Список цитируемой литературы	8

Постановка задачи:

Реализовать сортировку по неубыванию массива 64-разрядных целых чисел типа методами “пузырька” и Шелла, а также провести их экспериментальное сравнение. Для каждого из методов необходимо предусмотреть возможность работы с массивами длины от 1 до N ($N > 0$).

Также необходимо вычислить число сравнений элементов и число перемещений элементов.

Сравнение методов сортировки необходимо проводить на одних и тех же исходных массивах. Массив хранится в динамической памяти, генерируется для длин 10, 100, 1000, 10000. Необходимо создать отдельную функцию, которая заполняет массив:

- 1) Так, чтобы элементы были упорядочены.
- 2) Так, чтобы элементы были упорядочены в обратном порядке.
- 3) Так, чтобы элементы были расположены в случайном порядке.
- 4) Так, чтобы элементы были расположены в случайном порядке.

Метод Шелла						
n	Параметр	Номер сгенерированного массива				Среднее значение
		1	2	3	4	
10	Сравнения	22	27	27	28	26
	Перемещения	0	13	11	13	9.25
100	Сравнения	503	668	859	849	719.75
	Перемещения	0	260	412	388	265
1000	Сравнения	8006	11716	15117	15725	12641
	Перемещения	0	4700	7606	8241	5136.75
10000	Сравнения	120005	172578	261906	265310	204949.75
	Перемещения	0	62560	147019	150298	89969.25

т. 1

Метод “пузырька”						
n	Параметр	Номер сгенерированного массива				Среднее значение
		1	2	3	4	
10	Сравнения	45	45	45	45	45
	Перемещения	0	45	19	31	23.75
100	Сравнения	4950	4950	4950	4950	4950
	Перемещения	0	4950	2588	2172	2427.5
1000	Сравнения	499500	499500	499500	499500	499500
	Перемещения	0	499500	245086	249409	248498.75
10000	Сравнения	49995000	49995000	49995000	49995000	49995000
	Перемещения	0	49995000	25084989	24609518	24922376.75

т. 2

Теоретическая оценка:

Теоретическая оценка асимптотической сложности метода Шелла (зависит от выбора h) [1]:

Худший исход	Среднее	Лучший исход
$O(n^2)$	$O(n^{5/3})$	$O(n^{1.5})$

т. 3

Как мы видим, результаты эксперимента (т. 1) совпадают с теоретической оценкой (т. 3).

Теоретическая оценка асимптотической сложности и количества сравнений и перемещений в методе сортировки “пузырьком” [2]:

Критерий/исход	Худший исход	Среднее	Лучший исход
Сложность	$O(n^2)$	$O(n^2)$	$O(n^2)$
Сравнения	$(n - 1) * n / 2$	$(n - 1) * n / 2$	$(n - 1) * n / 2$
Перемещения	$(n - 1) * n / 2$	$(n - 1) * n / 4$	0

Как мы видим, результаты эксперимента (т. 2) совпадают с теоретической оценкой (т. 4).

Структура программы и спецификация функций:

Функции:

- 1) `bool Check()` – проверяет, отсортирован ли массив в порядке неубывания, возвращает `true` в случае корректности и `false` в случае ошибки в функции сортировки, принимает в качестве параметров размер массива и сам массив.
- 2) `void BubbleSort()` – функция, реализующая сортировку прямым обменом (“пузырьком”), принимает в качестве параметров размер массива и сам массив.
- 3) `void ShellSort()` – функция, реализующая сортировку методом Шелла (где $h = n / 2$, следующее значение $h = h / 2$), принимает в качестве параметров размер массива и сам массив.
- 4) `int* Generate()` – функция, создающая массив в зависимости от параметра v , принимает в качестве параметров размер массива и соответственно параметр v .

Отладка программы, тестирование функций:

Тестирование было осуществлено с помощью функции `Check()`, проводилось на массиве, элементы которого были:

- 1) Расположены в порядке возрастания
- 2) Расположены в порядке убывания
- 3) Расположены в случайном порядке

Функция `check()` отлавливает случаи неправильной сортировки элементов.

Анализ допущенных ошибок:

В результате отладки функций сортировки были “пойманы” ошибки, допущенные из-за невнимательности.

Список литературы:

- [1] Кнут Д. Искусство программирования для ЭВМ. Том 3. М.: Мир, 1978.
- [2] Вирт Н. Алгоритмы и структуры данных. М.: Мир, 1989.