

Московский Государственный Университет  
им. М. В. Ломоносова.  
Факультет Вычислительной Математики и Кибернетики

## Отчет по домашнему заданию №7

Мингалёв Олег  
oleg@mingalev.net  
101 группа

Москва, 2014

## Постановка задачи

Реализовать два метода сортировки: метод простого выбора и быстрая сортировка массива чисел типа `int` и провести их экспериментальное сравнение. Тип элементов массива, конкретные методы и вид сортировки определяются вариантом задания. Для каждого из реализуемых методов необходимо предусмотреть возможность работы с массивами длины от 1 до N.

При реализации каждого метода вычислить число сравнений элементов, число перемещений (обменов) элементов и провести замер времени работы метода в процессорных тактах.

Рассмотреть возможность выбора одного из двух видов сортировки:

- Числа упорядочиваются по невозрастанию.
- Числа упорядочиваются по неубыванию модулей, т.е. при сравнении элементов не учитывается знак.

Сравнение методов сортировки необходимо проводить на одних и тех же исходных массивах, при этом следует рассмотреть массивы разной длины. N рассмотреть, как минимум,  $n = 10, 20, 50, 100$ . Генерация исходных массивов для сортировки реализуется отдельной функцией, создающей в зависимости от заданного параметра и заданной длины конкретный массив, в котором:

- элементы уже упорядочены (1);
- элементы упорядочены в обратном порядке (2);
- расстановка элементов случайна (3, 4).

Результаты экспериментов оформить на основе нескольких запусков программы в виде сводной таблицы.

# Эксперимент

Рассмотрим сортировку Шелла и упорядочивание чисел по невозрастанию.

N	Параметр	Номер сгенерированного массива				Среднее значение
		1	2	3	4	
10	Сравения	27	22	30	28	26.75
	Обмены	13	0	13	10	9.00
	Такты	4128	1104	3592	3008	2958.00
20	Сравения	80	62	83	89	78.50
	Обмены	36	0	27	40	25.75
	Такты	10288	4632	9344	10024	8572.00
50	Сравения	263	203	370	373	302.25
	Обмены	105	0	195	195	123.75
	Такты	16888	7168	29040	28528	20406.00
100	Сравения	668	503	840	810	705.25
	Обмены	260	0	389	361	252.50
	Такты	35680	16728	79352	71064	50706.00
1000	Сравения	11716	8006	15810	15144	12669.00
	Обмены	4700	0	8345	7640	5171.25
	Такты	701136	248064	1331584	1430392	927794.00

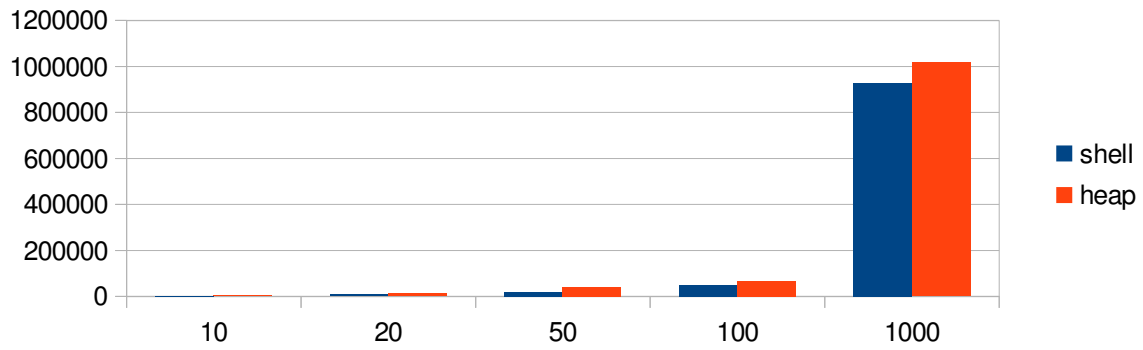
# Эксперимент

Теперь рассмотрим алгоритм пирамидальной сортировки с такой же функцией сравнения. .

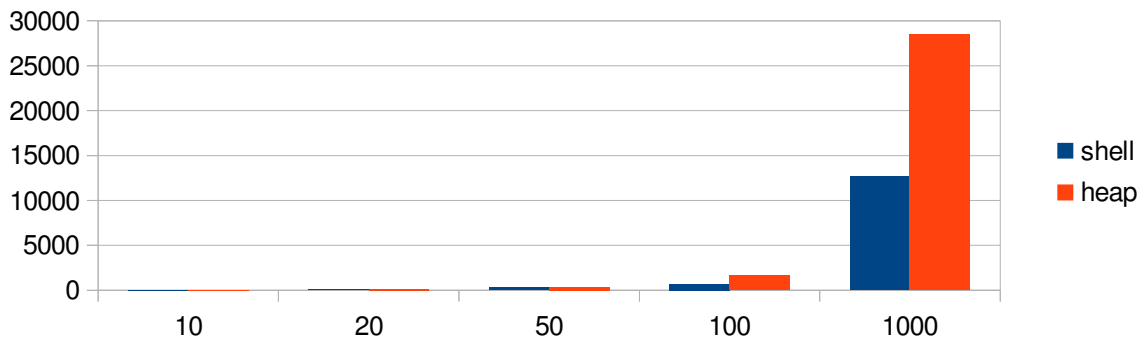
N	Параметр	Номер сгенерированного массива				Среднее значение
		1	2	3	4	
10	Сравнения	61	56	66	65	62.00
	Обмены	41	28	42	44	38.75
	Такты	4408	4088	3824	4120	4110.00
20	Сравнения	182	164	188	204	184.50
	Обмены	108	72	106	119	101.25
	Такты	13728	12976	12960	12800	13116.00
50	Сравнения	641	581	699	692	653.25
	Обмены	346	241	384	376	336.75
	Такты	40576	36936	43192	43616	41080.00
100	Сравнения	1618	1453	1760	1788	1654.75
	Обмены	837	577	940	965	829.75
	Такты	59024	58256	73768	75656	66676.00
1000	Сравнения	27671	23961	31620	30741	28498.25
	Обмены	13681	8981	17165	16272	14024.75
	Такты	1118728	723640	1103312	1122120	1016950.00

# Эксперимент

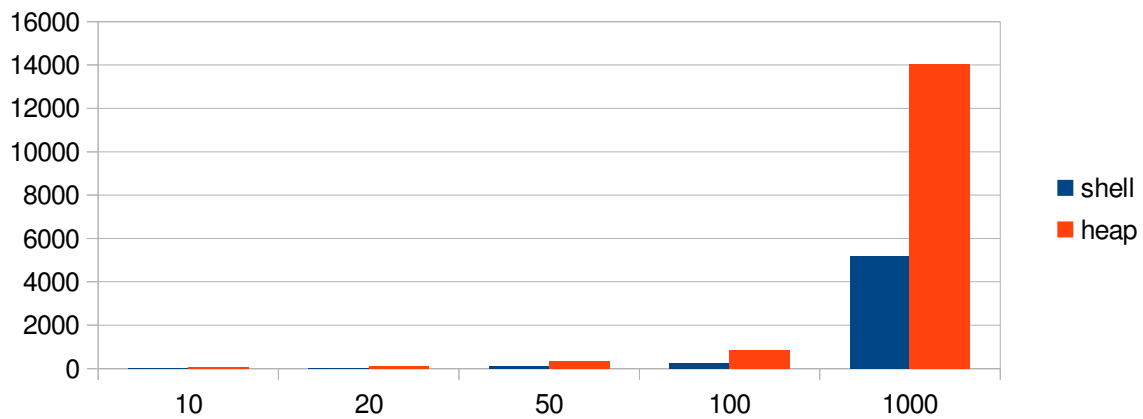
Сравнение времени работы



Сравнение количества сравнений



Сравнение количества обменов



# Описание процесса сборки программы

Для сборки программы в ОС Linux необходимо:

- утилита `make`
- утилита `make`
- утилита `make`

Используйте `make main_down` для выбора сортировки по невозрастанию и `make main_abs_up` для сортировки по неубыванию модуля.

Для удаления объектных файлов, созданных в ходе сборки программы следует вызвать цель `clean`: `make clean`.