Доколин Георгий ИУ7-22Б

Отчет по Проектно-технологической практике (тестированию, отладке и профилированию ПО)
Задание №4 в рамках вычислительного практикума.
Постановка замерного эксперимента

Оглавление

Цель

Цель: На основе задачи №4 ЛР№2 (сортировка) по курсу «Программирование на Си» провести сравнение производительности программы для разных способов работы с элементами одномерного массива:

• использование операции индексации a[i];

• формальная замена операции индексации на выражение *(a + i);

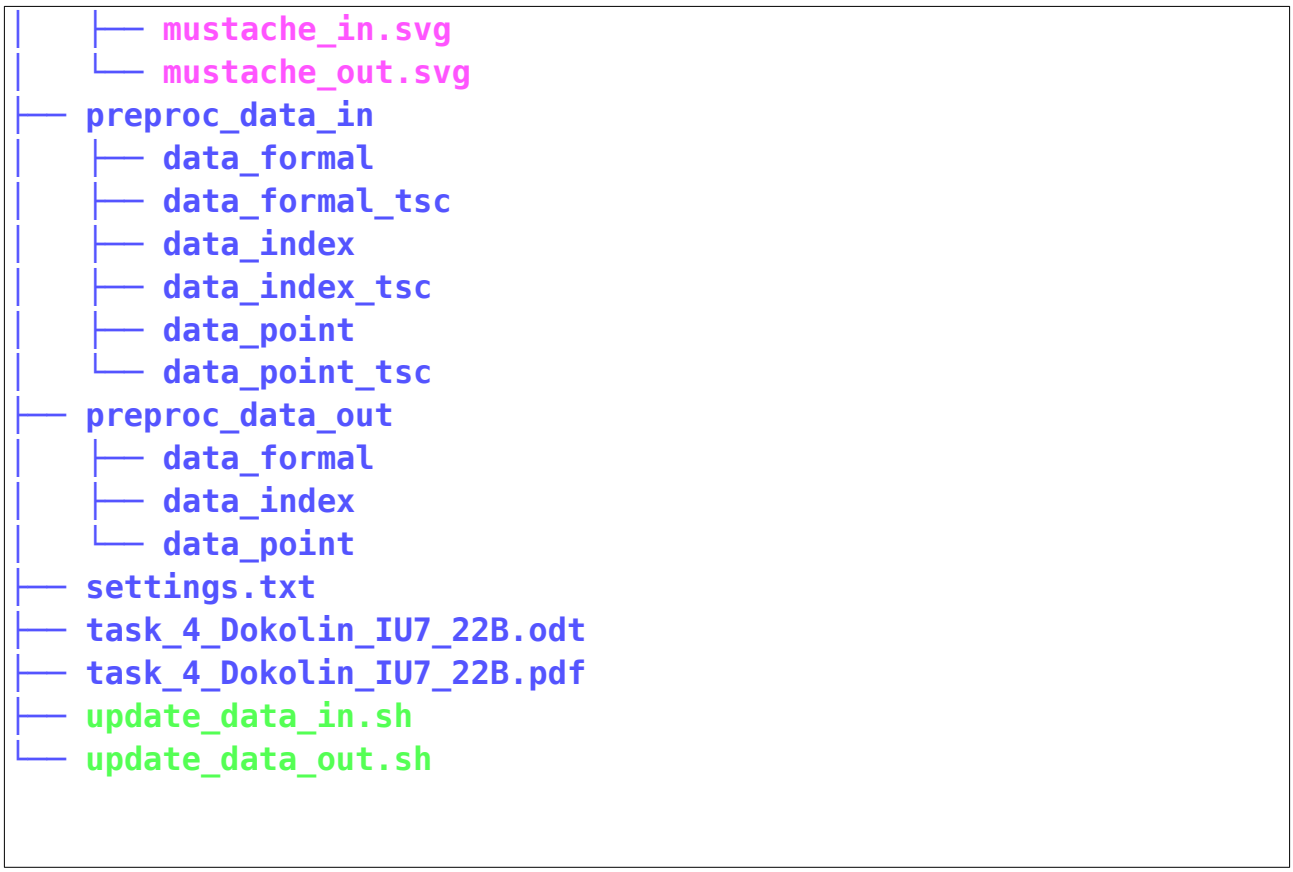
• использование указателей для работы с массивом.

Список необходимого ПО

1. **Matplotlib** для построение графиков.
2. **Numpy** для хранения данных, по которым строится график.

Файловая структура

```
.
├── app_in
├── apps_out
│   ├── app_formal
│   ├── app_index
│   └── app_point
├── build_apps_out.sh
├── c_files_in
│   ├── main_formal.c
│   ├── main_formal_tsc.c
│   ├── main_index.c
│   ├── main_index_tsc.c
│   ├── main_point.c
│   └── main_point_tsc.c
├── c_files_out
│   ├── main_formal.c
│   ├── main_index.c
│   └── main_point.c
├── clean.sh
├── cnt_rse.py
├── data_in
│   ├── data_formal
│   ├── data_formal_tsc
│   ├── data_index
│   ├── data_index_tsc
│   ├── data_point
│   └── data_point_tsc
├── data_out
│   ├── data_formal
│   ├── data_index
│   └── data_point
├── go.sh
├── make_postproc.py
├── make_preproc.py
├── postproc_data
│   ├── error_in.svg
│   ├── error_out.svg
│   ├── line_in.svg
│   ├── line_in_tsc.svg
│   ├── line_out.svg
```



1. В папке app_in хранятся файлы, которые запускаются для внутреннего замера.
2. Папка apps_out разбита на три папки, которые хранят в себе файлы, которые необходимы для внешнего замера времени каждым из способов представления массива.
3. В папке c_files_in хранятся Си файлы, которые меряют время сортировки много раз внутри самой программы.
4. В папке c_files_out хранятся Си файлы, которые единоразово меряют время выполнения сортировки.
5. Папка data_in разделена на 6 папок, которые хранят в себе результаты внутренних замеров разными способами подсчета времени сортировки, для каждого вида представления массива.
6. Папка data_out разделена на 3 папки, которые хранят в себе результаты внешних замеров для каждого из способов представления массива.
7. В папке postproc_data хранятся только графики, постренные на основе результатов измерения.

8. Папка preproc_data_in разделена на 6 папок, которые хранят в себе результаты обработанных данных, собранных при помощи внутреннего замера.
9. Папка preproc_data_out разделена на 3 папки, которые хранят в себе результаты обработанных данных, собранных при помощи внешнего замера.

Описание скриптов и программ

build_apps_out.sh

Скрипт создает исполняемые файлы для всех видов представляения массива, а так же со всеми размерами, которые заданы в файле settings.txt в формате начального и конечного количества, с определенным шагом.

update_data_in.sh

Сначала скрипт создает исполняемый файл для замера времени сортировки, отдельно для каждого вида представляения массива, в милисекундах, а так же со всеми размерами, которые заданы в файле settings.txt в формате начального и конечного количества, с определенным шагом. Следующим действием он повторяет все те же операции, только создавая и обрабатывая файлы для замера времени сортировки в тиках. После каждого создания и запуска исполняемого файла, скрипт записывает результат в соответсвующий текстовый файл. Исполняемый файл продолжает свою работу по созданию и сортировке массива, пока RSE не станет менее 1 процента.

update_data_out.sh

Данный скрипт запускает каждый исполняемый файл из apps_out, и сохраняет результаты работы в папку data_out, которая разбита еще на несоклько папок так же как и папка apps_out. Отличие от прошлого скрипта заглючается в том, что скрипт запускает исполняемый файл заново, до того момента, пока RSE составляет более 1 процента. RSE считается при помощи программы cnt_rse.py.

cnt_rse.py

Программа работает для подсчета RSE при каждом запуске внешним способом измерения, если данная программа возвращает 0, значит измерения больше не требуются и можно переходить к следующему файлу, а если возвращает 1, значит требуются еще данные.

make_preproc.py

Программа создает новые текстовые файлы, в которые сохраняет данные для построения графиков, такие как: среднее арифметическое, медиана, максимум, минимум, квартили, а также я добавил сохранение RSE и количество замеров, поскольку эти параметры требуются для итоговой таблицы.

make_postproc.py

Программа для построения графиков, по данным, которые собрала программа make_preproc.py

Программы сортиртировки для внутреннего замера

c_files_in/main_formal.c – Программа, которая создает массив и замеряет его время сортировки в милисекнудах, в последствии считая RSE, который если больше единицы запускает создание и сортировку повторно. Работа с массивом осуществляется припомощи формальной замены операции индексация на выражение *(a + i).

c_files_in/main_index.c – Программа, которая создает массив и замеряет его время сортировки в милисекнудах, в последствии считая RSE, который если больше единицы запускает создание и сортировку повторно. Работа с массивом осуществляется при помощи индексации.

c_files_in/main_point.c – Программа, которая создает массив и замеряет его время сортировки в милисекнудах, в последствии считая RSE, который если больше единицы запускает создание и сортировку повторно. Работа с массивом осуществляется при помщи указателей.

c_files_in/main_formal_tsc.c – Программа, которая создает массив и замеряет его время сортировки в тиках, в последствии считая RSE, который если больше единицы запускает создание и сортировку повторно. Работа с массивом осуществляется припомощи формальной замены операции индексация на выражение *(a + i).

c_files_in/main_index_tsc.c – Программа, которая создает массив и замеряет его время сортировки в тиках, в последствии считая RSE, который если больше единицы запускает создание и сортировку повторно. Работа с массивом осуществляется при помощи индексации.

c_files_in/main_point_tsc.c – Программа, которая создает массив и замеряет его время сортировки в тиках, в последствии считая RSE, который если больше единицы запускает создание и сортировку повторно. Работа с массивом осуществляется при помщи указателей.

Программы сортиртировки для внешнего замера

c_files_out/main_formal.c – Программа, которая создает массив и замеряет его время сортировки, а проверка и повторный запуск осуществляются вне программы. Работа с массивом осуществляется припомощи формальной замены операции индексация на выражение *(a + i).

c_files_out/main_index.c – Программа, которая создает массив и замеряет его время сортировки, а проверка и повторный запуск осуществляются вне программы. Работа с массивом осуществляется при помощи индексации.

c_files_out/main_point.c – Программа, которая создает массив и замеряет его время сортировки, а проверка и повторный запуск осуществляются вне программы. Работа с массивом осуществляется при помщи указателей.

clean.sh

Скрипт предначзначен для удаления всех исполняемых и текстовых файлов, которые созданы при выполнении программ.

go.sh

Скрипт создан для запуска всех скриптов и программ последовательно.

Таблица измерений времени сортировки массива в милисекундах, работа с которым происходит при помощи формальной индексации. (Внутренний замер)

Размер	t, мс	Кол-во повторов	RSE
2	0,00	25	0,00
502	0,09	100119	0.99
1002	0,45	12389	0.99
1502	1,06	513	0.99
2002	1,88	309	0.99
3002	4,19	97	0.99
4002	7,75	29	0.98
5002	12,16	25	0,61
10002	49,44	25	0,20
15002	111,32	25	0,18

Таблица измерений времени сортировки массива в милисекундах, работа с которым происходит при помощи формальной индексации. (Внешний замер)

Размер	t, мс	Кол-во повторов	RSE
2	0,00	25	0,00
502	0, 00	25	0,00
1002	0,45	12046	0,99
1502	1,08	618	0,99
2002	1,98	101	0,99
3002	4,38	157	0,99
4002	8,16	31	0,99
5002	12,63	32	0,99
10002	49,64	25	0,25
15002	111,32	25	0,14

Таблица измерений времени сортировки массива в милисекундах, работа с которым происходит при помощи индексации. (Внутренний замер)

Размер	t, мс	Кол-во повторов	RSE
2	0,00	25	0,00
502	0,09	100455	0,99
1002	0,44	12365	0,99
1502	1,05	443	0,99
2002	1,90	326	0,99
3002	4,18	83	0.99
4002	7,78	27	0,99
5002	12,12	25	0,55
10002	49,60	25	0,20
15002	111,56	25	0,10

Таблица измерений времени сортировки массива в милисекундах, работа с которым происходит при помщи индексации. (Внешний замер)

Размер	t, мс	Кол-во повторов	RSE
2	0,00	25	0,00
502	0,09	100592	0,99
1002	0,44	12518	0,99
1502	1,06	531	0,99
2002	1,89	311	0,99
3002	4,20	93	0,99
4002	7,88	25	0,84
5002	12,24	25	0,71
10002	49,44	25	0,20
15002	112,44	25	0,26

Таблица измерений времени сортировки массива в милисекундах, работа с которым происходит при помощи указателей. (Внутренний замер)

Размер	t, мс	Кол-во повторов	RSE
2	0,00	25	0,00
502	0,09	103780	0,99
1002	0,43	12989	0,99
1502	1,05	395	0,99
2002	1,83	406	0,99
3002	4,06	32	0,99
4002	7,60	40	0,99
5002	11,96	25	0,33
10002	48,08	25	0,12
15002	108,28	25	0,08

Таблица измерений времени сортировки массива в милисекундах, работа с которым происходит при помщи указателей. (Внешний замер)

Размер	t, мс	Кол-во повторов	RSE
2	0,00	25	0,00
502	0,00	25	0,00
1002	0,42	13313	0,99
1502	1,03	272	0,99
2002	1,85	364	0,99
3002	4,08	48	0,98
4002	7,62	42	0,99
5002	11,96	25	0,33
10002	48,20	25	0,17
15002	108,44	25	0,09

Таблица измерений времени сортировки массива в тиках, работа с которым происходит при помощи индексации. (Внутренний замер)

Размер	t, тики	Кол-во повторов	RSE
2	19,40	562	0,99
502	219747,04	25	0,25
1002	1085234,24	25	0,23
1502	2572530,24	25	0,50
2002	4609487,28	25	0,19
3002	10367302,40	25	0,38
4002	19265129,36	25	0,28
5002	30333269,04	25	0,15
10002	123109223,28	25	0,03
15002	280059196,64	25	0,09

Таблица измерений времени сортировки массива в тиках, работа с которым происходит при помощи формальной индексации. (Внутренний замер)

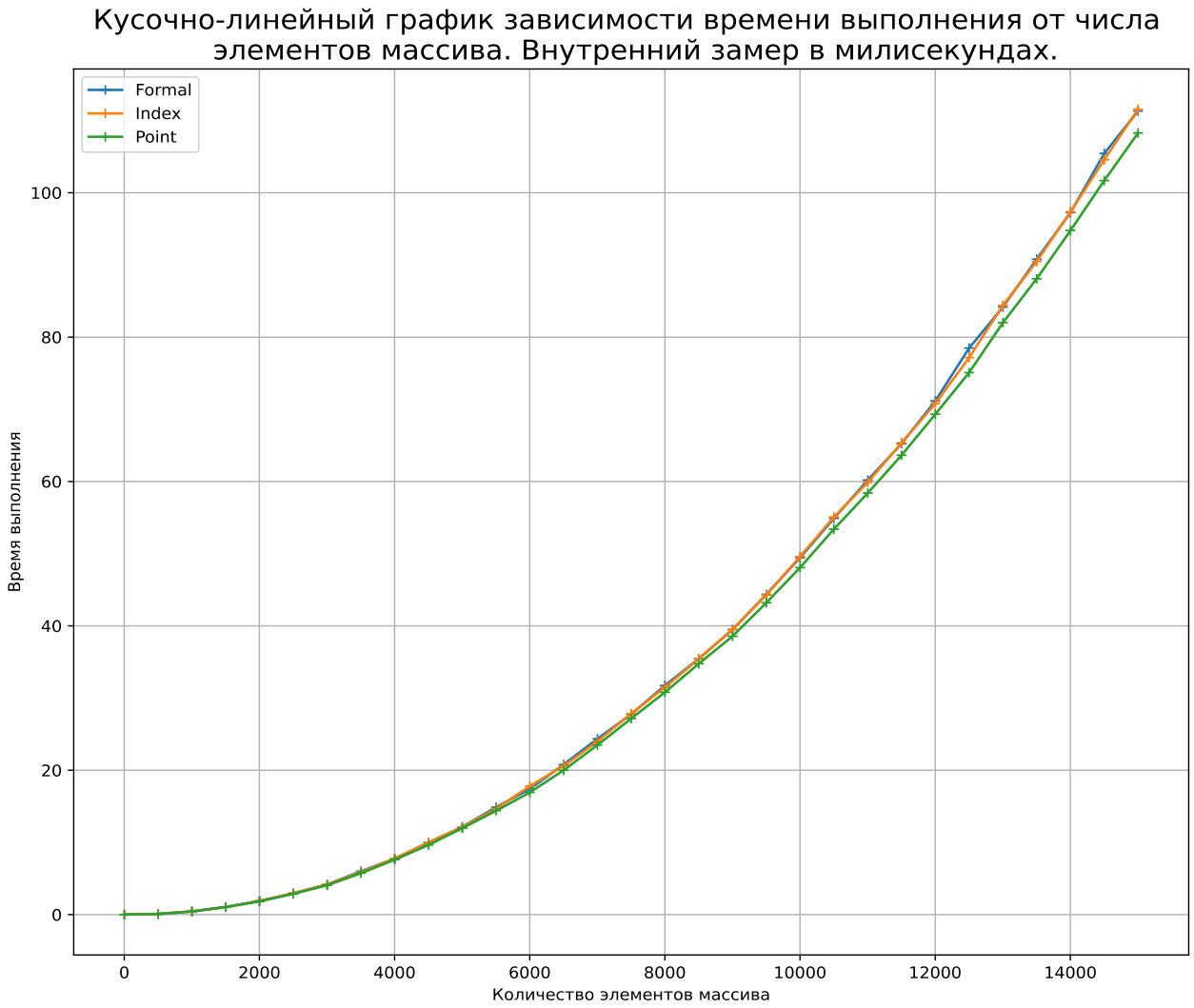
Размер	t, тики	Кол-во повторов	RSE
2	19,22	379	0,99
502	217576,08	25	0,04
1002	1097453,04	25	0,55
1502	2555982,00	25	0,21
2002	4642437,52	25	0,84
3002	10338917,12	25	0,30
4002	19252932,40	25	0,20
5002	30198891,36	25	0,07
10002	123503186,80	25	0.13
15002	279207858,16	25	0,03

Таблица измерений времени сортировки массива в тиках, работа с которым происходит при помощи указателей. (Внутренний замер)

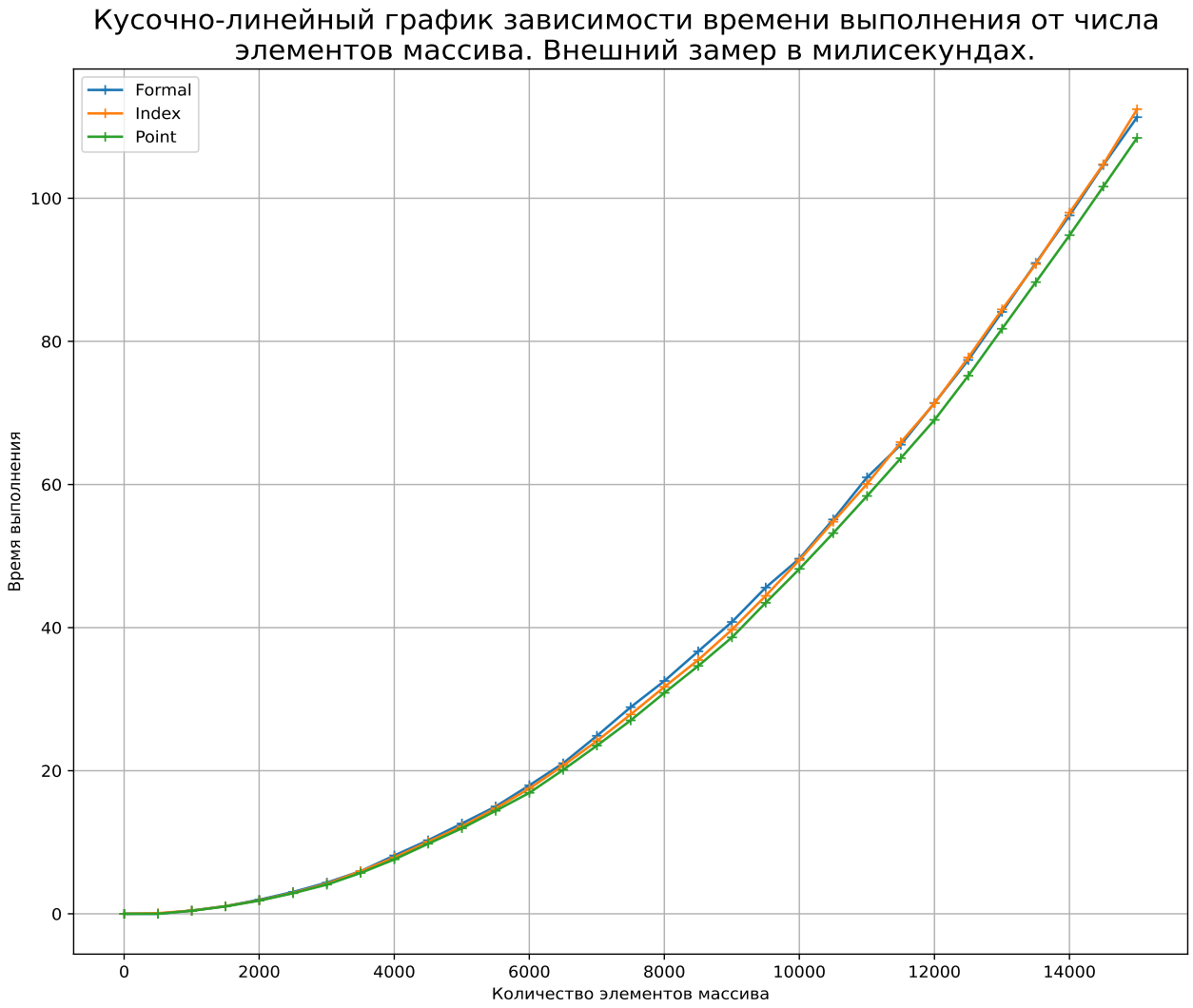
Размер	t, тики	Кол-во повторов	RSE
2	20,40	690	0,99
502	211269,60	25	0,41
1002	1046260,32	25	0,08
1502	2468088,40	25	0,21
2002	4423467,52	25	0,06
3002	9936894,40	25	0,07
4002	18095799,12	25	0,15
5002	28654588,64	25	0,10
10002	115743638,00	25	0,01
15002	261036356,48	25	0,01

Графики

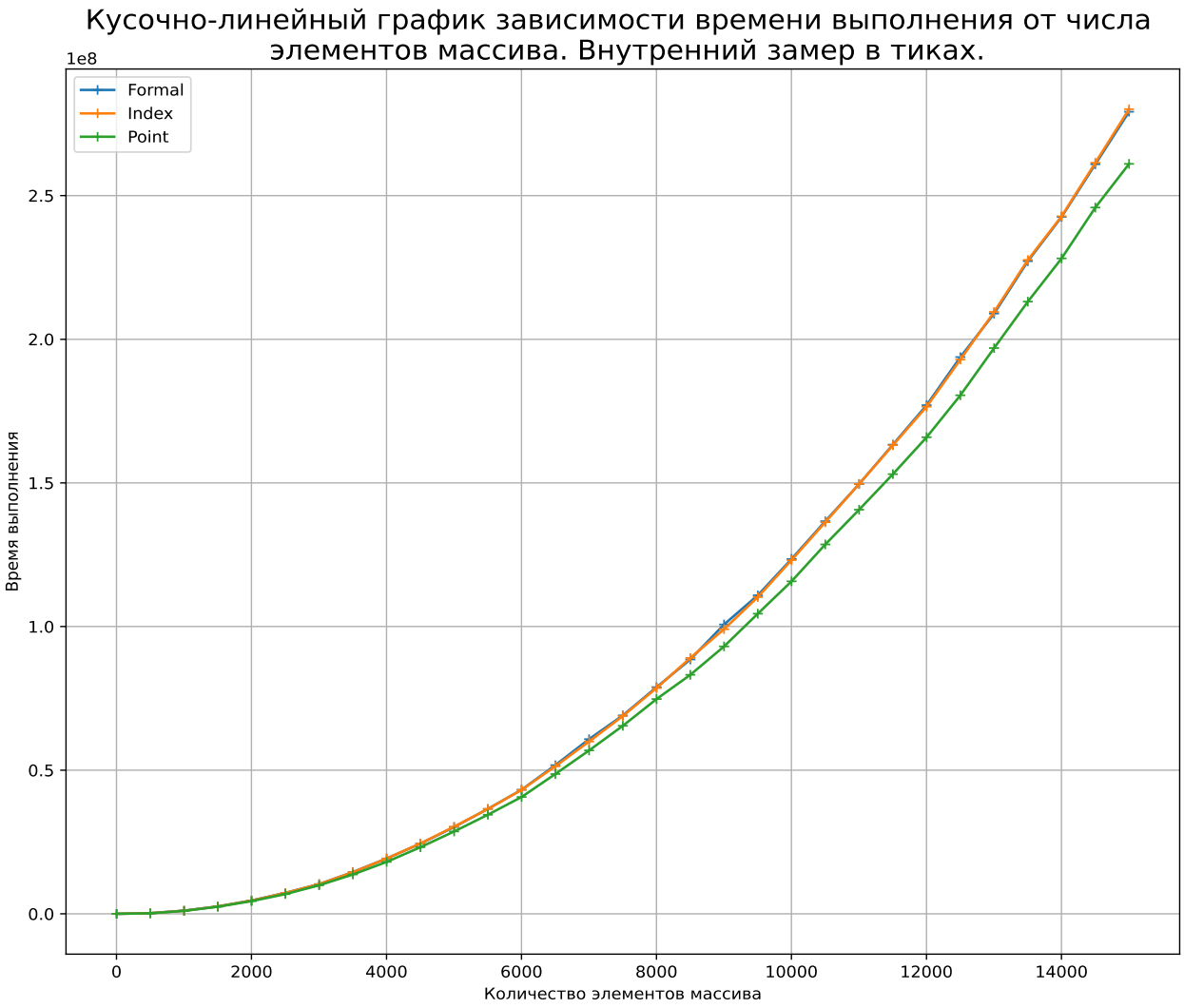
Кусочно-линейный график зависимости времени выполнения от числа элементов массива. Внутренний замер в милисекундах.



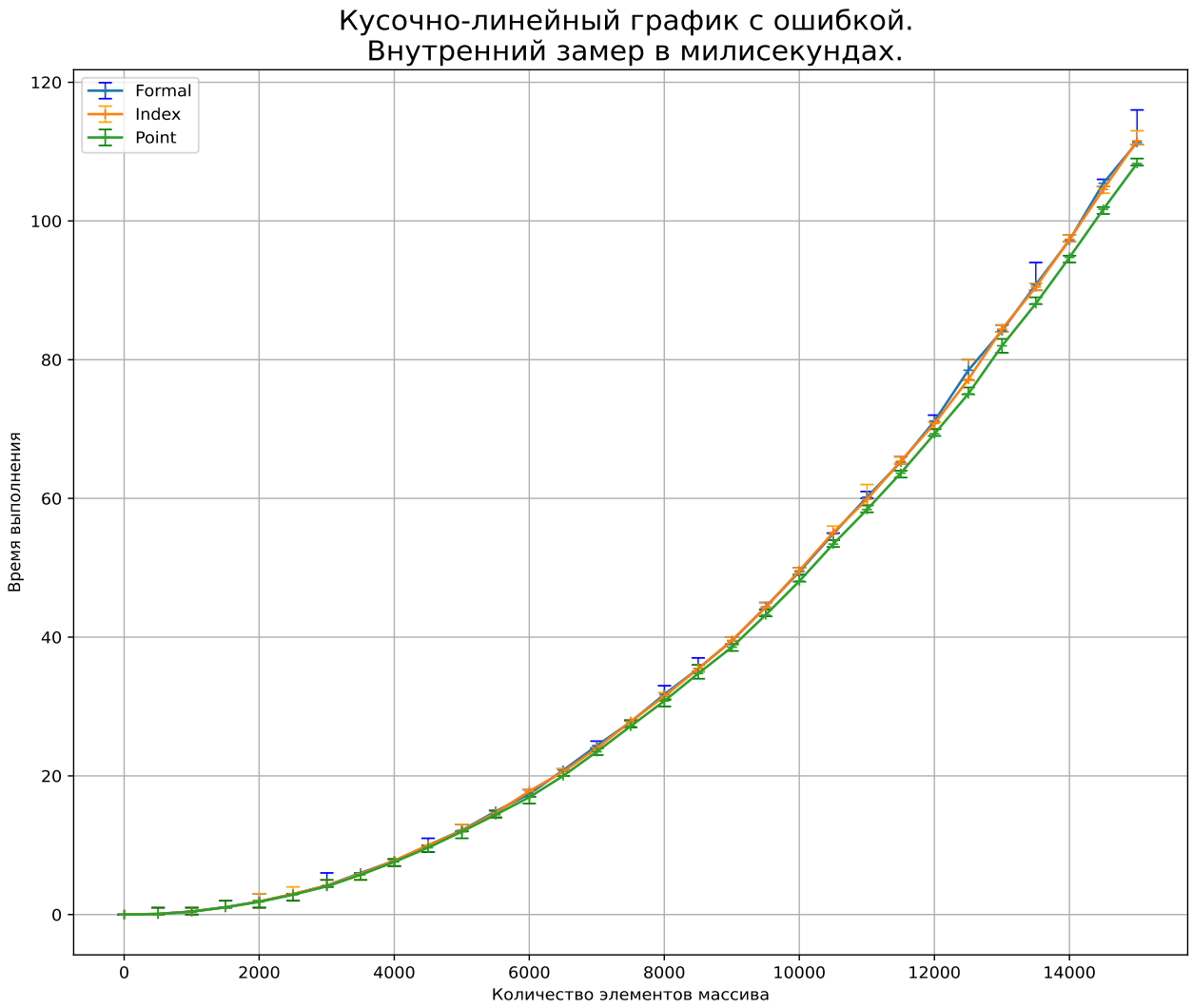
Кусочно-линейный график зависимости времени выполнения от числа элементов массива. Внешний замер в милисекундах.



Кусочно-линейный график зависимости времени выполнения от числа элементов массива. Внутренний замер в тиках.



Кусочно-линейный график с ошибкой. Внутренний замер в милисекундах.



Кусочно-линейный график с ошибкой. Внешний замер в милисекундах.

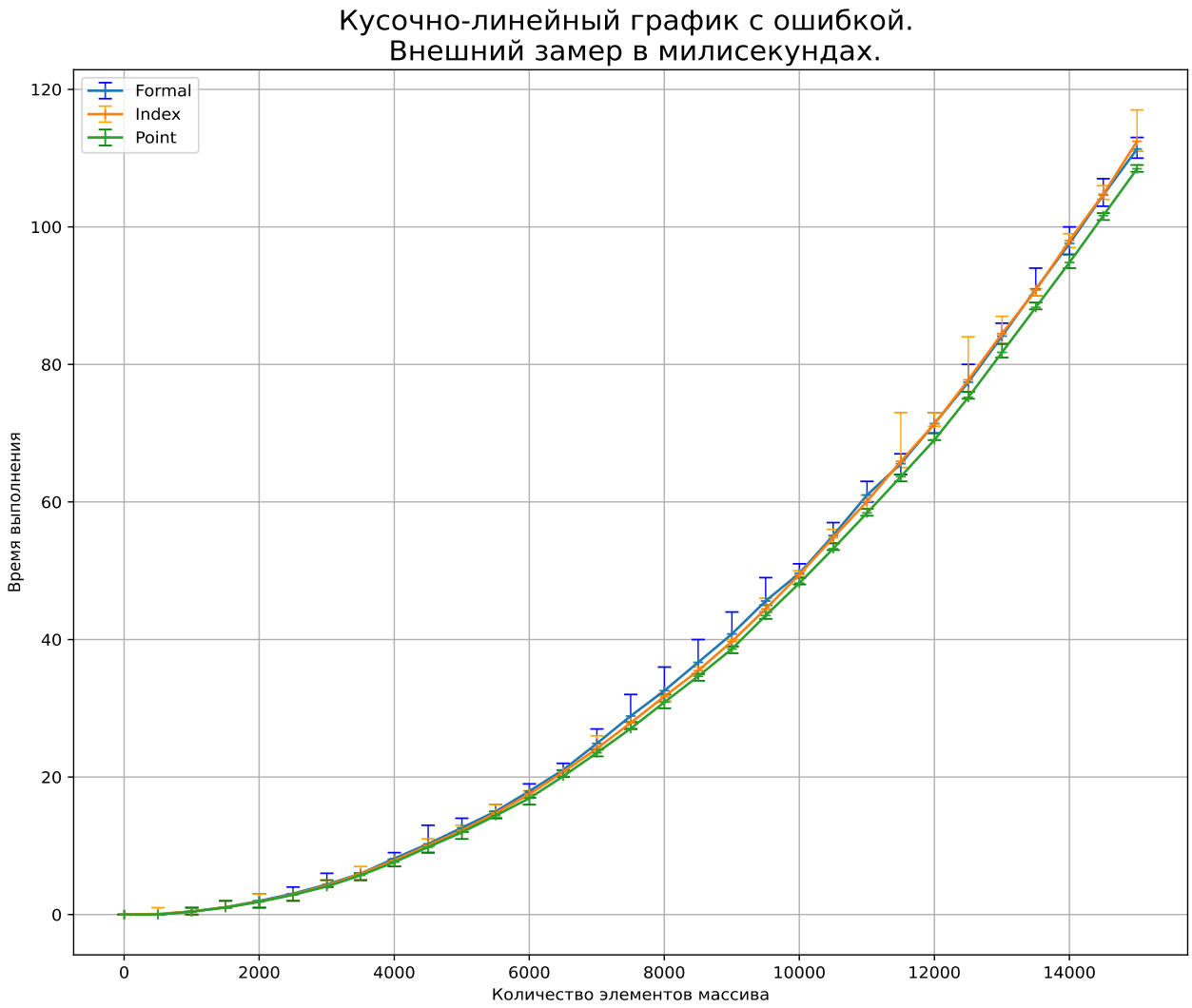


График с усами. Внутренний замер в милисекундах.

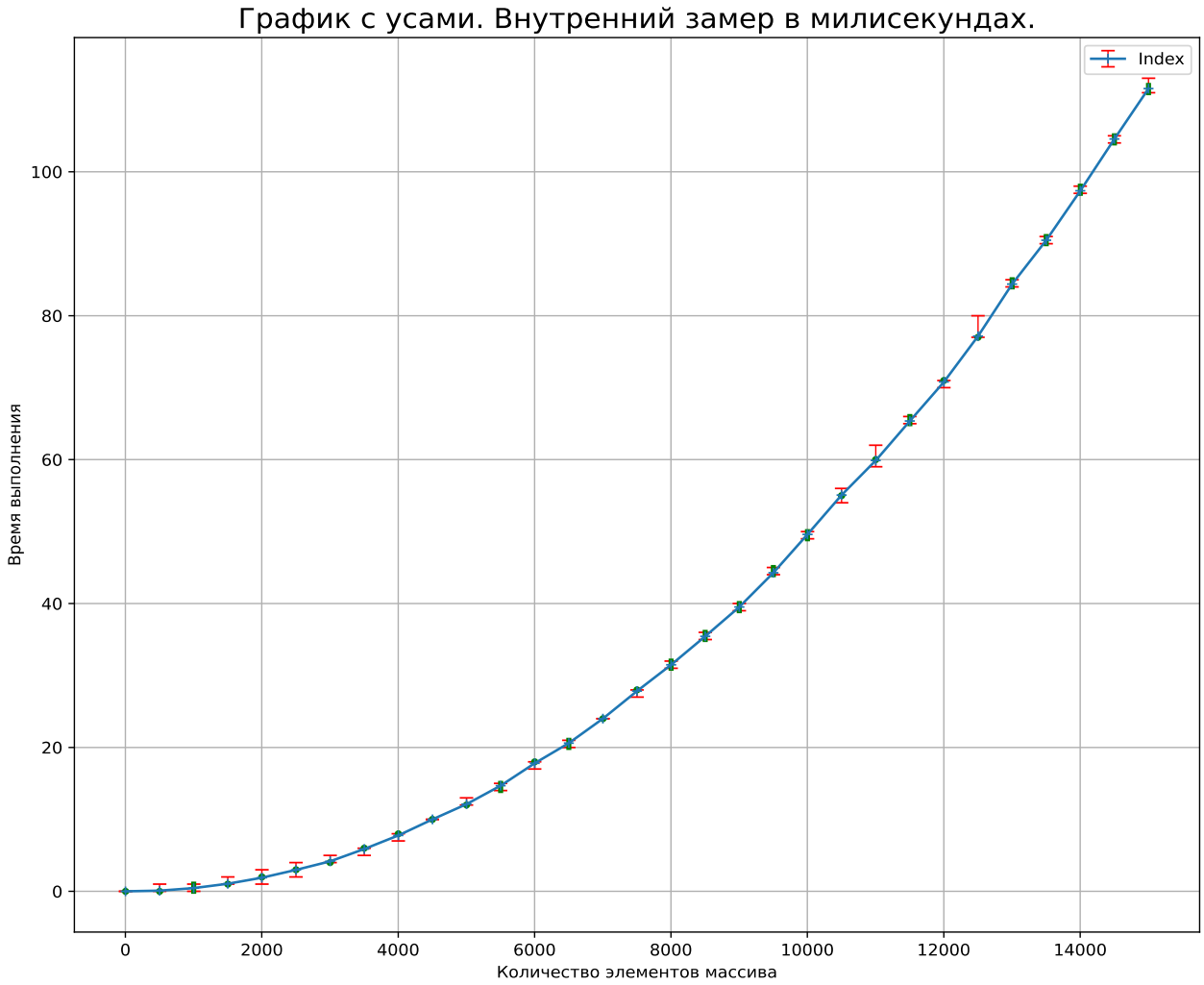
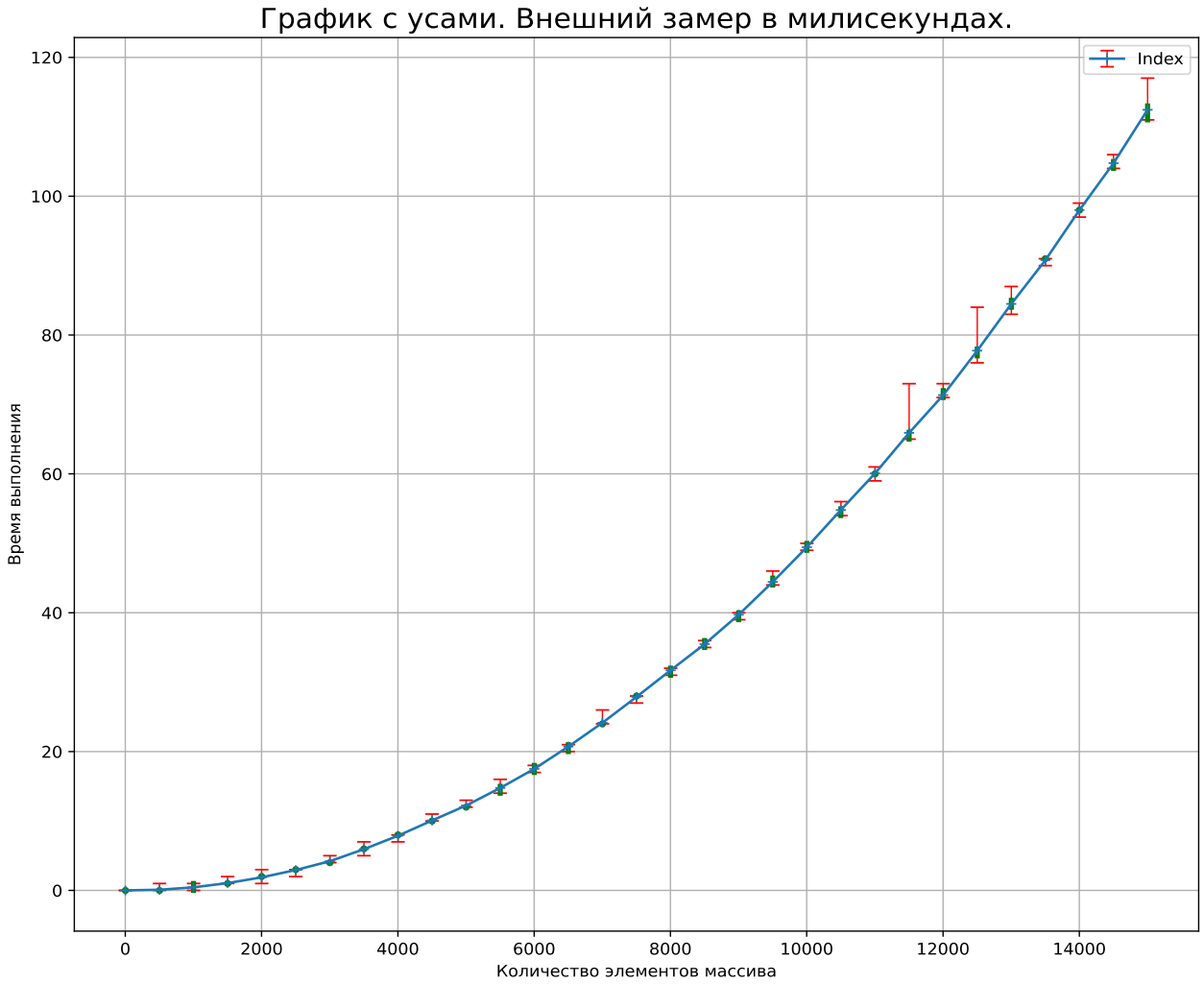


График с усами. Внешний замер в милисекундах.



Вывод

Просмотрев графики и оценив таблицы с данными, можно понять, что программа работает быстрее если работать с массивом при помощи указателей, это показали внутренние и внешние замеры времени сортировок массивов разных длин с использованием различных методов счета времени. Благодаря данной работе, я научился правильно определять время работы программы разными способами, используя для исследования несколько языков праграммирования. Цели работы достигнуты.