Санкт-Петербургский
государственный
университет
www.spbu.ru

Реализация быстрых операций длинной арифметики для intel x64
Отчёт по учебной практике

Казанцев Антон Алексеевич, группа 20.Б11-мм

Научный руководитель: старший преподаватель Баклановский М.В.

Санкт-Петербург
2022

Цель работы

Написание программы, реализующей быстрое выполнение операций длинной арифметики для x64.

Задачи

- изучить существующие реализации длинной арифметики
- выбрать набор операций длинной арифметики для реализации
- изучить особенности устройства процессора intel x64 и возможные способы ускорить вычисления
- составить качественную тестовую базу
- научиться сравнивать скорость вычислений двух и более различных реализаций длинной арифметики
- написать свою реализацию, используя инкрементный метод разработки для качественного анализа роста производительности по отношению к применяемым технологиям процессора

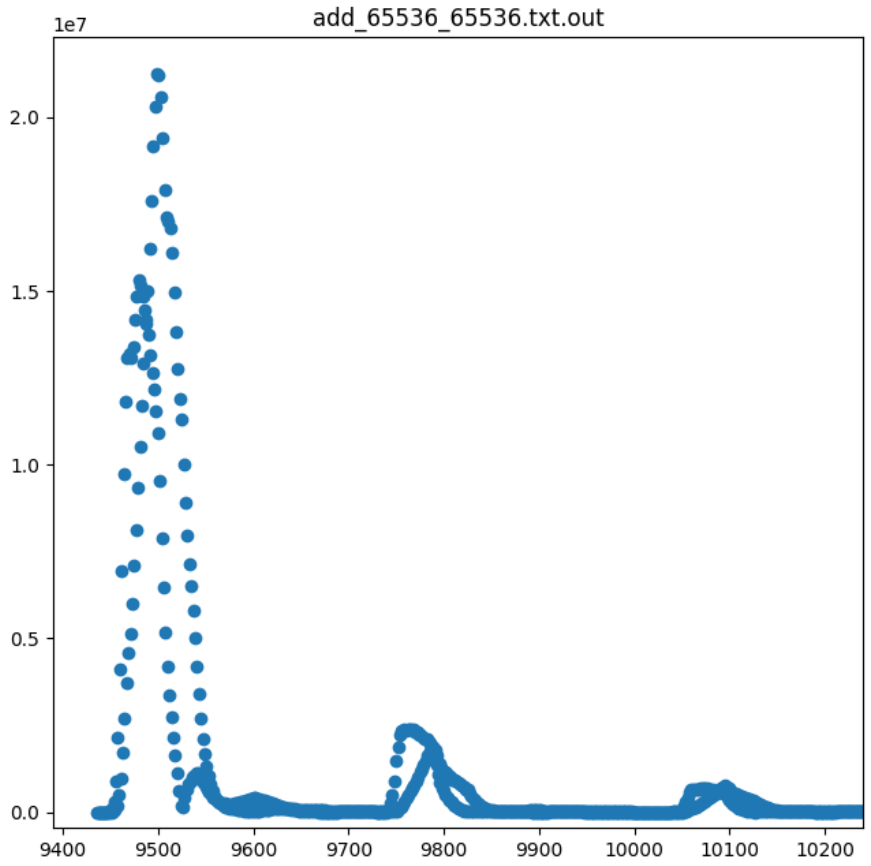
GNU MP

- типы данных:
  - целые числа
  - рациональные числа
  - числа с плавающей запятой
- функции для целых чисел:
  - сложение
  - вычитание
  - сравнение
  - умножение (алгоритм Карацубы)
  - совмещённые операции сложения и умножения
  - взятие модуля
  - деление с получением целого или нецелого результата
  - возведение в степень
  - логические операции

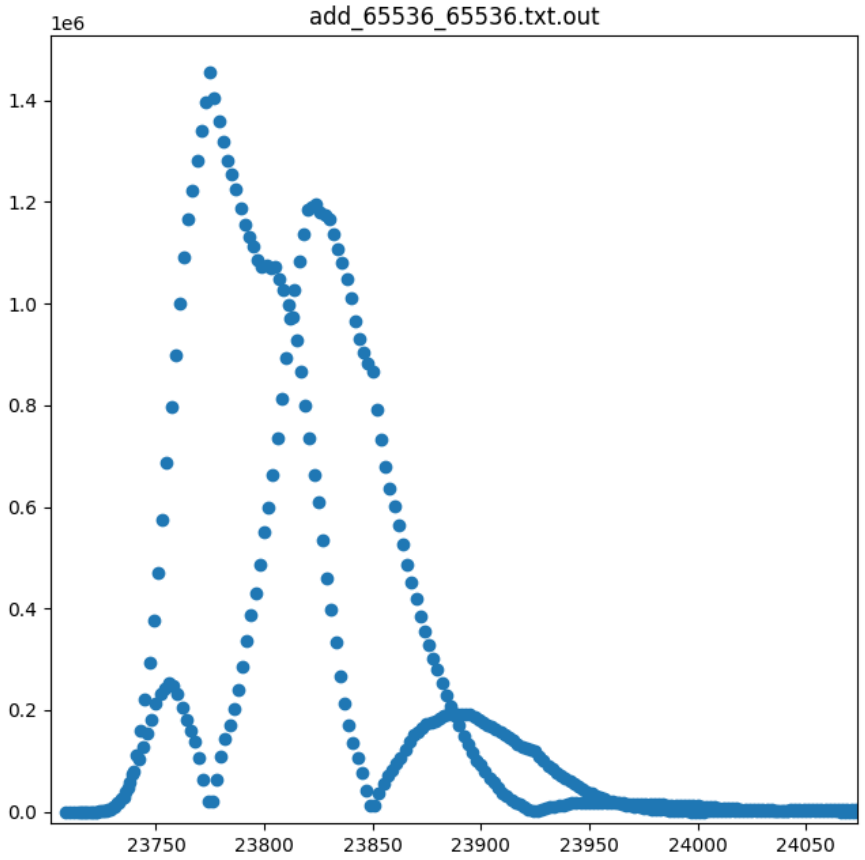
Особенности тестирования

1. Остановка конвейера
2. Замер времени старта
3. Вызов функции с тестируемым алгоритмом
4. Остановка конвейера
5. Замер времени финиша

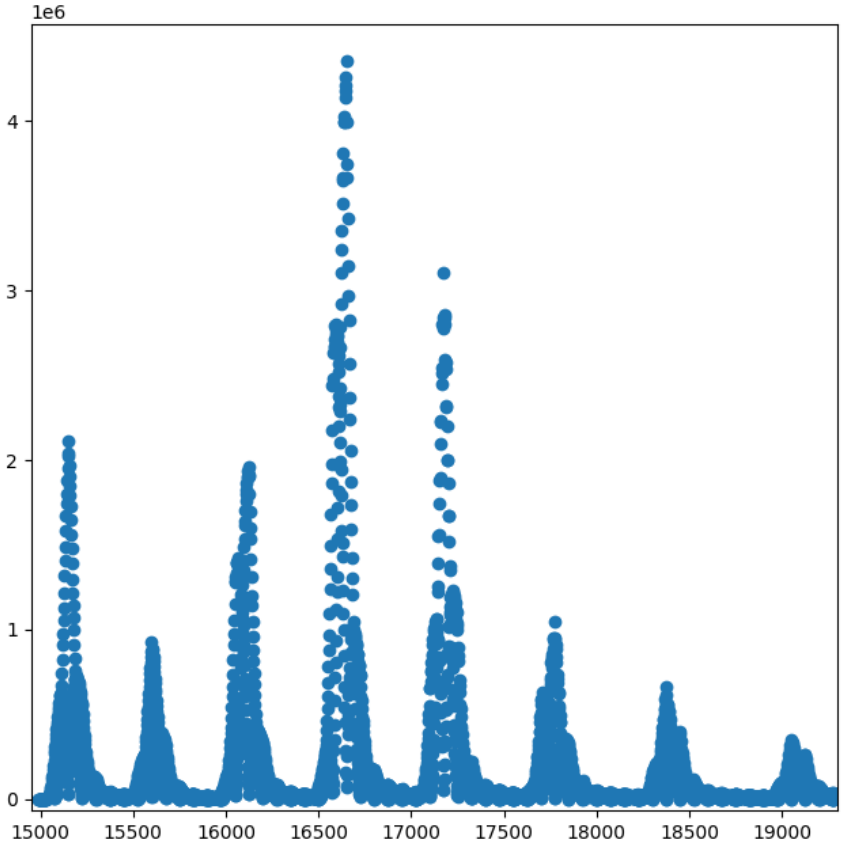
Тест GNU MP сумма



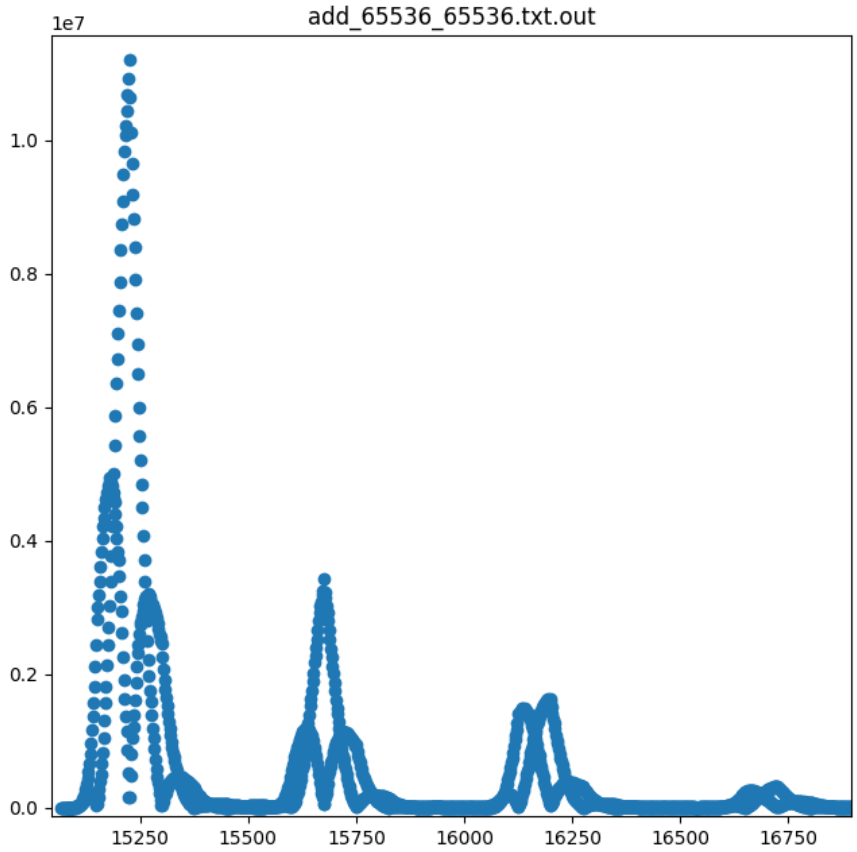
Тест реализации, СИ



Тест реализации, ASM



Тест реализации, ASM выравнивание в памяти

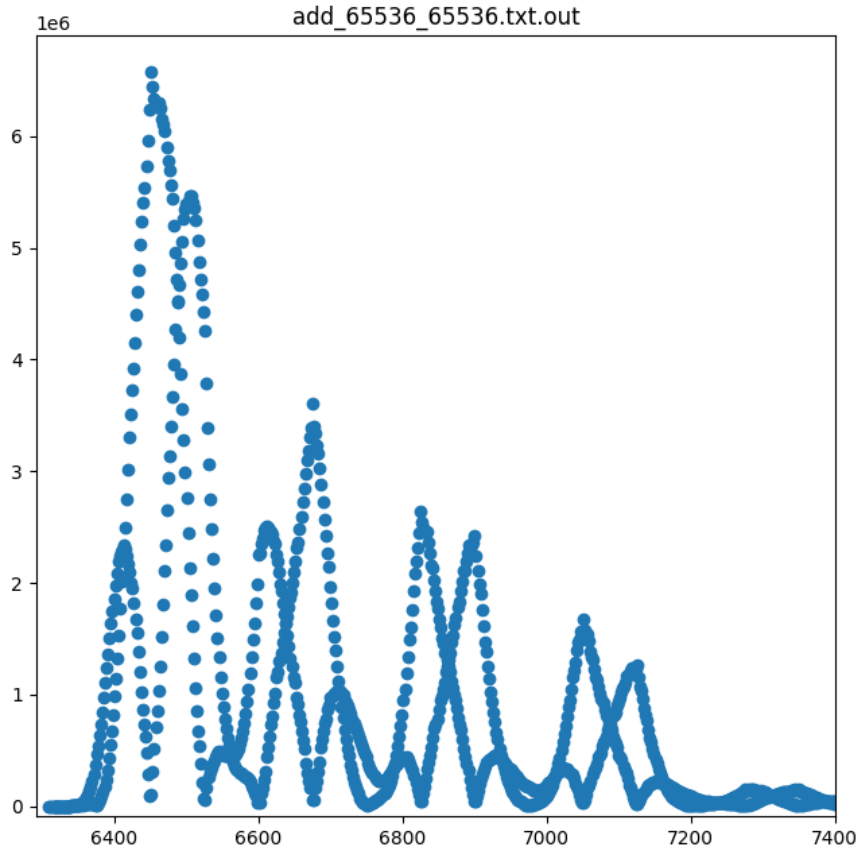


Улучшение алгоритма

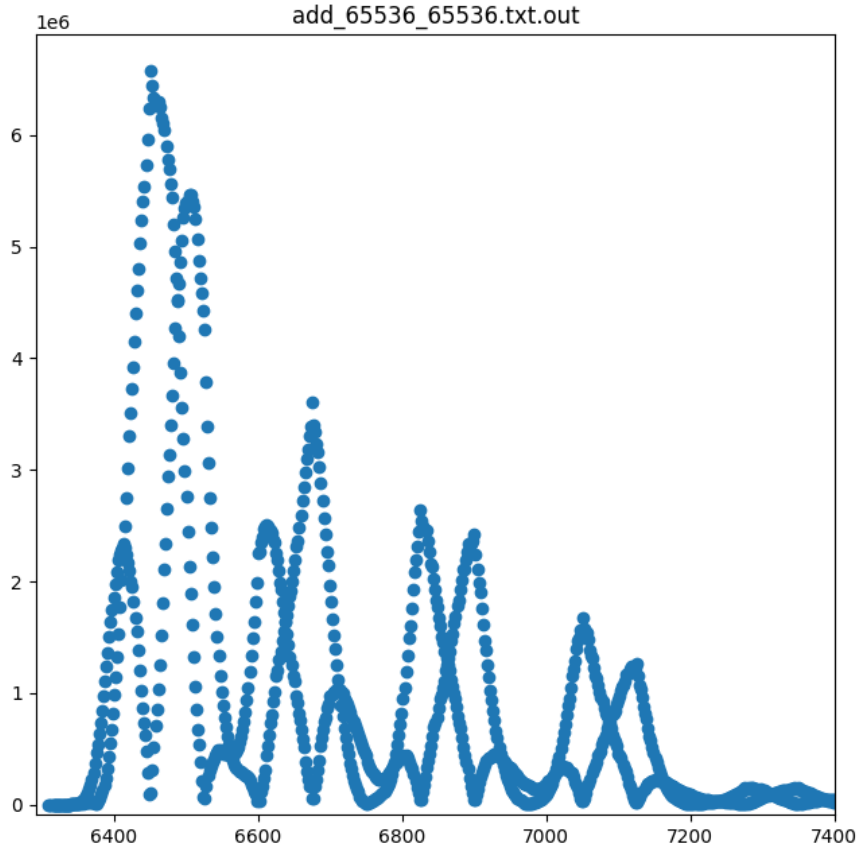
sum_loop:		**Указатель начала цикла**
lodsq		rax = [rsi]; rsi = rsi + 1;
adc	**rax, qword ptr [rdi]**	rax = rax + [rdi] + cf
stosq		[rdi] = rax; rdi = rdi + 1;
loop	**sum_loop**	rcx = rcx – 1; if rcx != 0 jump sum_loop;

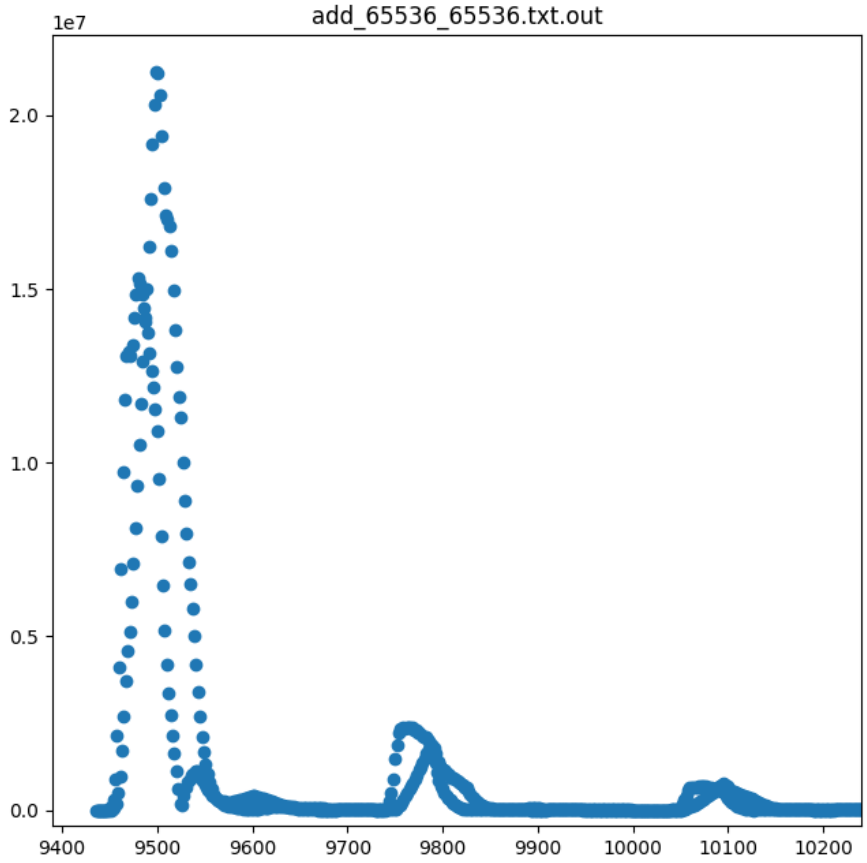
sum_loop:		**Указатель начала цикла**
lodsq		rax = [rsi]; rsi = rsi + 1;
adc	**rax, qword ptr [rdi]**	rax = rax + [rdi] + cf
stosq		[rdi] = rax; rdi = rdi + 1;
dec	**rcx**	rcx = rcx - 1
jnz	**sum_loop**	if rcx != 0 jump sum_loop;

Тест реализации, ASM

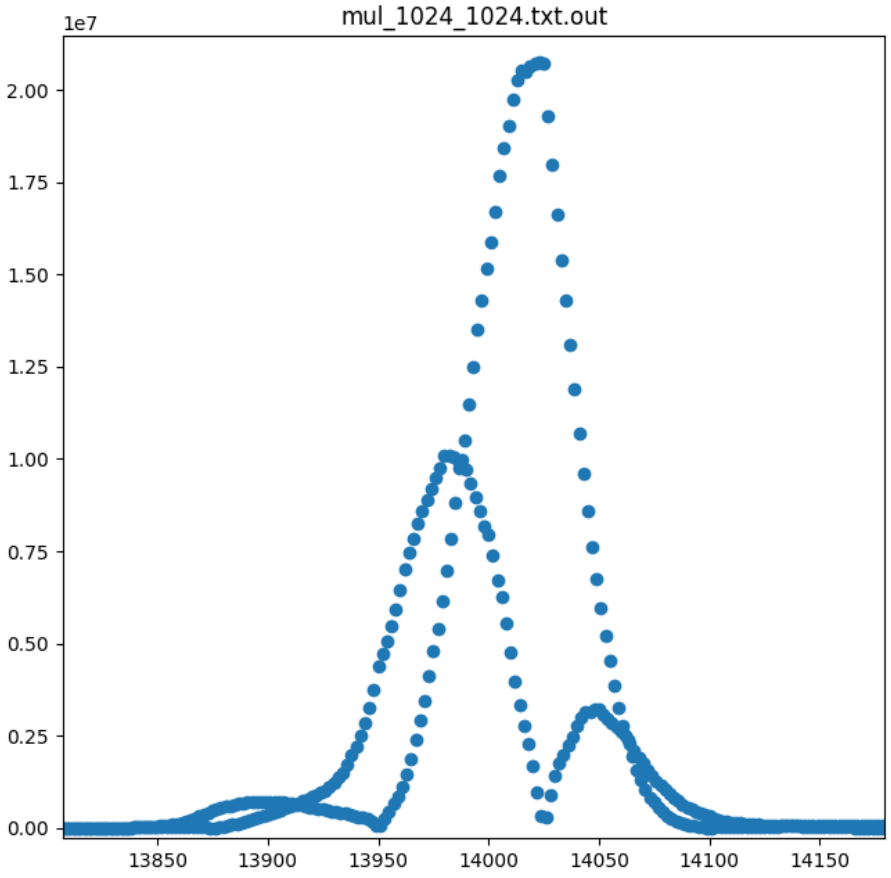


Сравнение

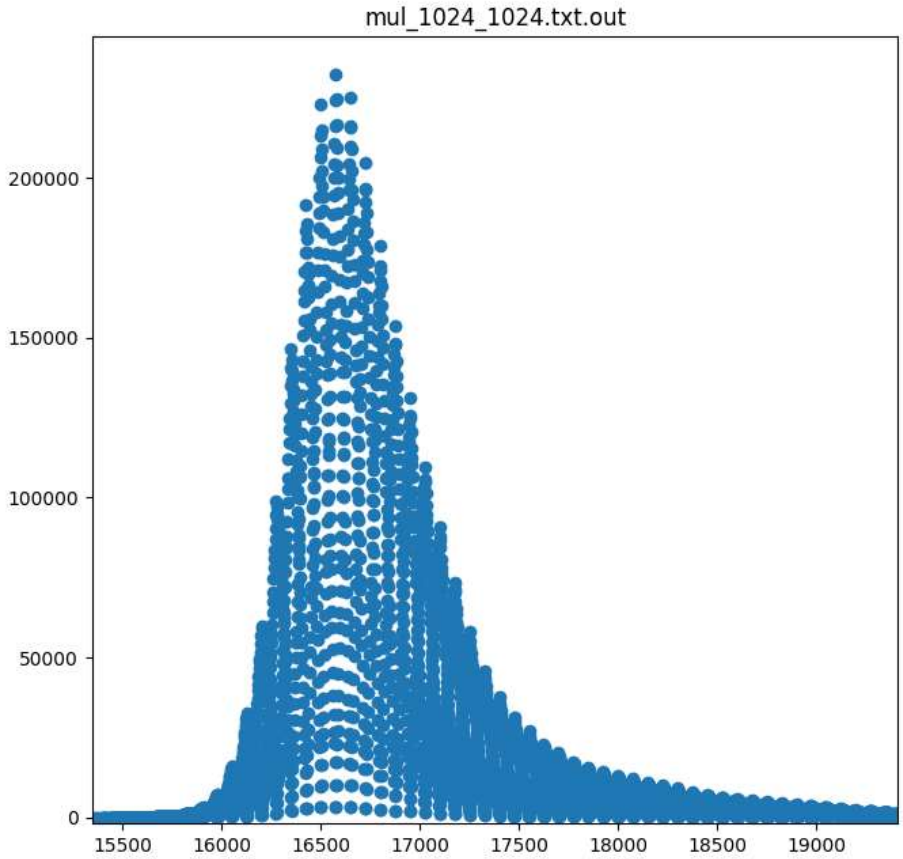

add_65536_65536.txt.out
1e6
0
1
2
3
4
5
6
6400
6600
6800
7000
7200
7400


add_65536_65536.txt.out
1e7
0.0
0.5
1.0
1.5
2.0
9400
9500
9600
9700
9800
9900
10000
10100
10200

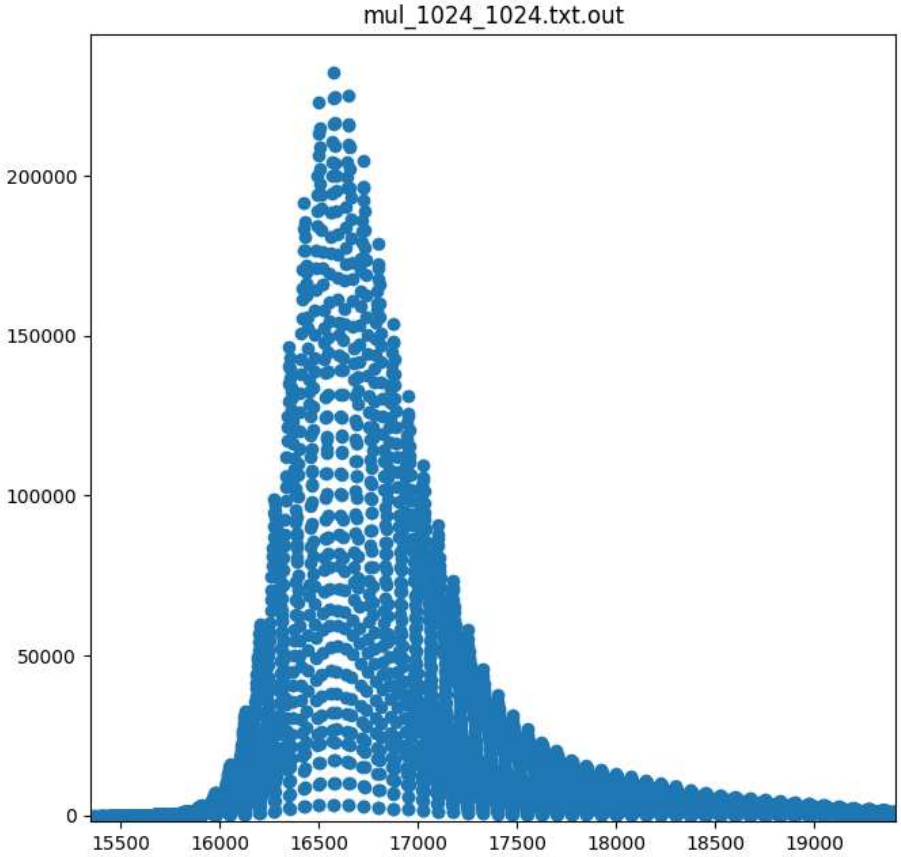
Тест GNU MP умножение

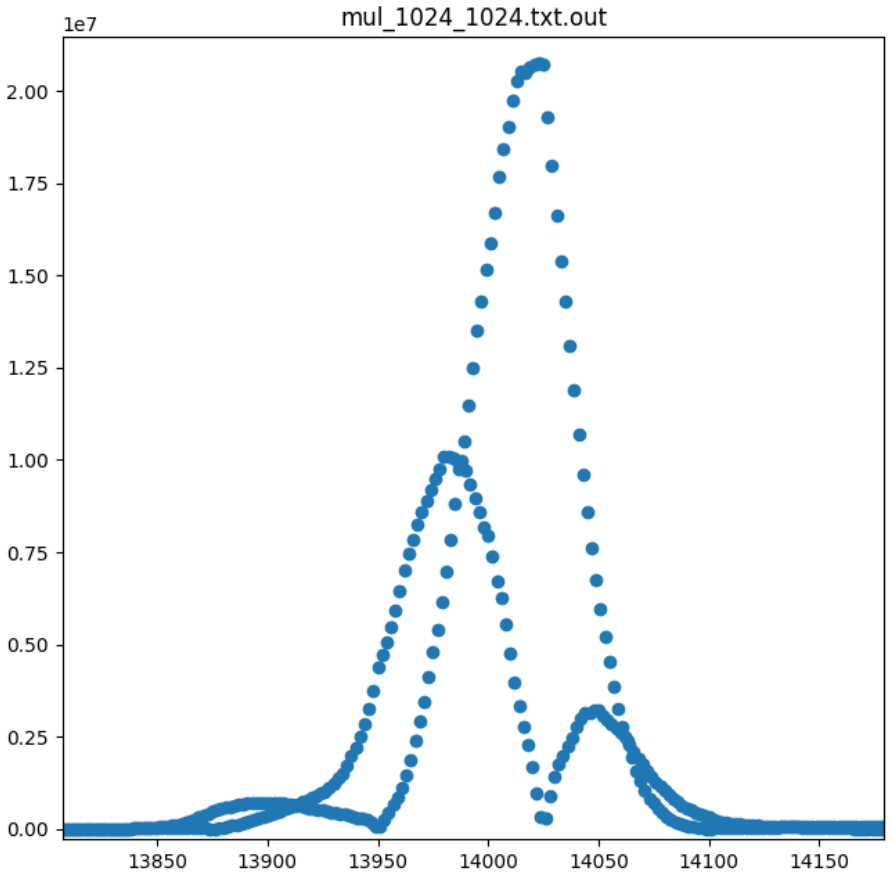


Тест умножение “в столбик”, ASM

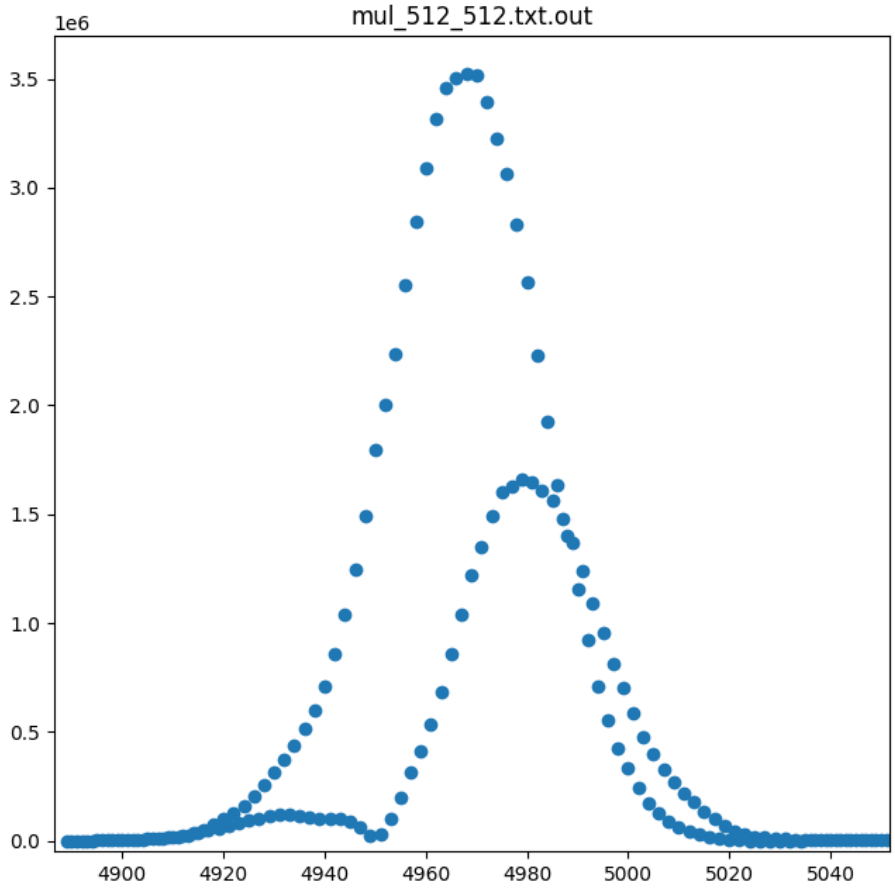


Сравнение

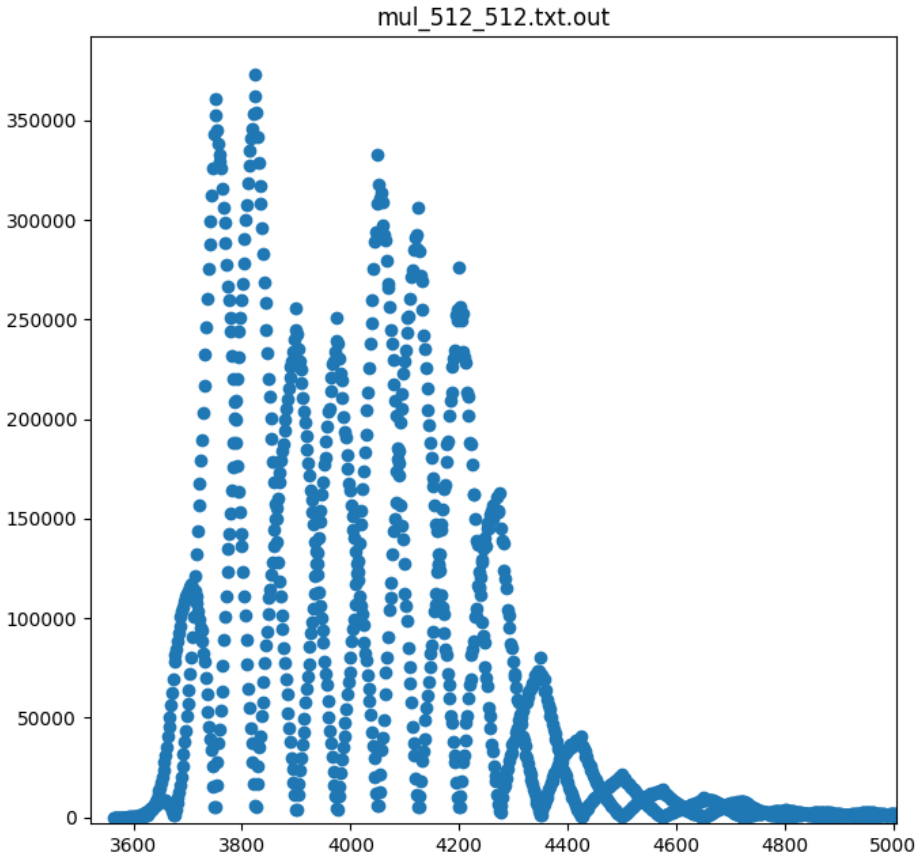

mul_1024_1024.txt.out
200000
150000
100000
50000
0
15500
16000
16500
17000
17500
18000
18500
19000


mul_1024_1024.txt.out
1e7
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
13850
13900
13950
14000
14050
14100
14150

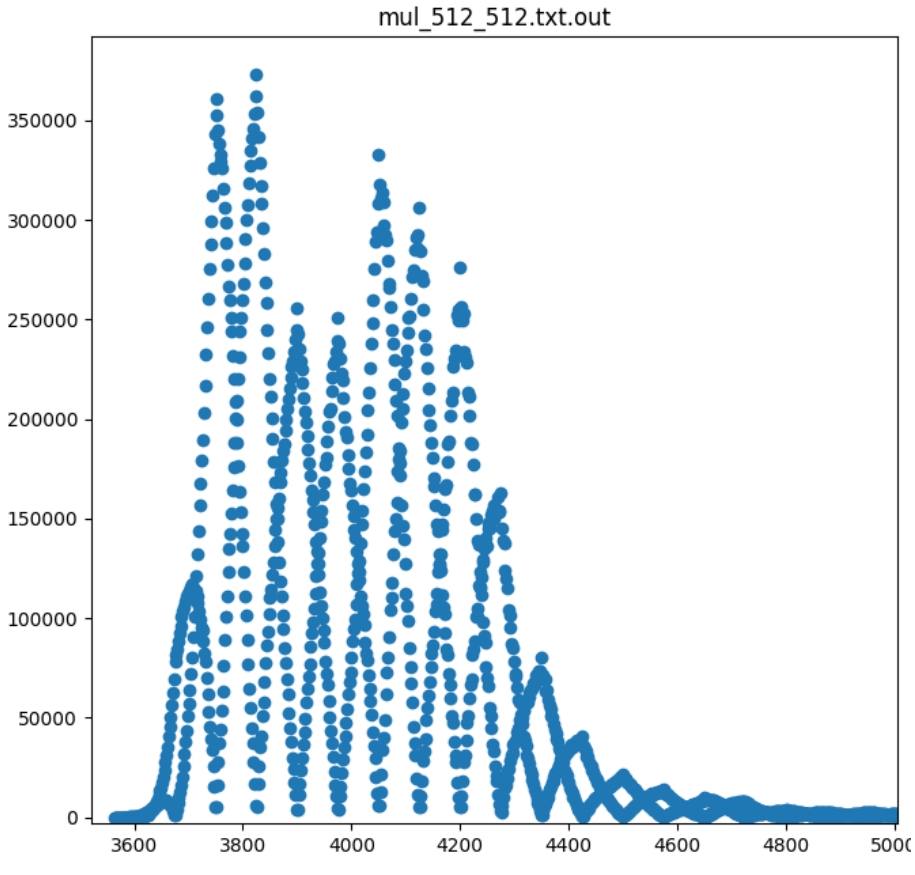
Тест GNU MP умножение

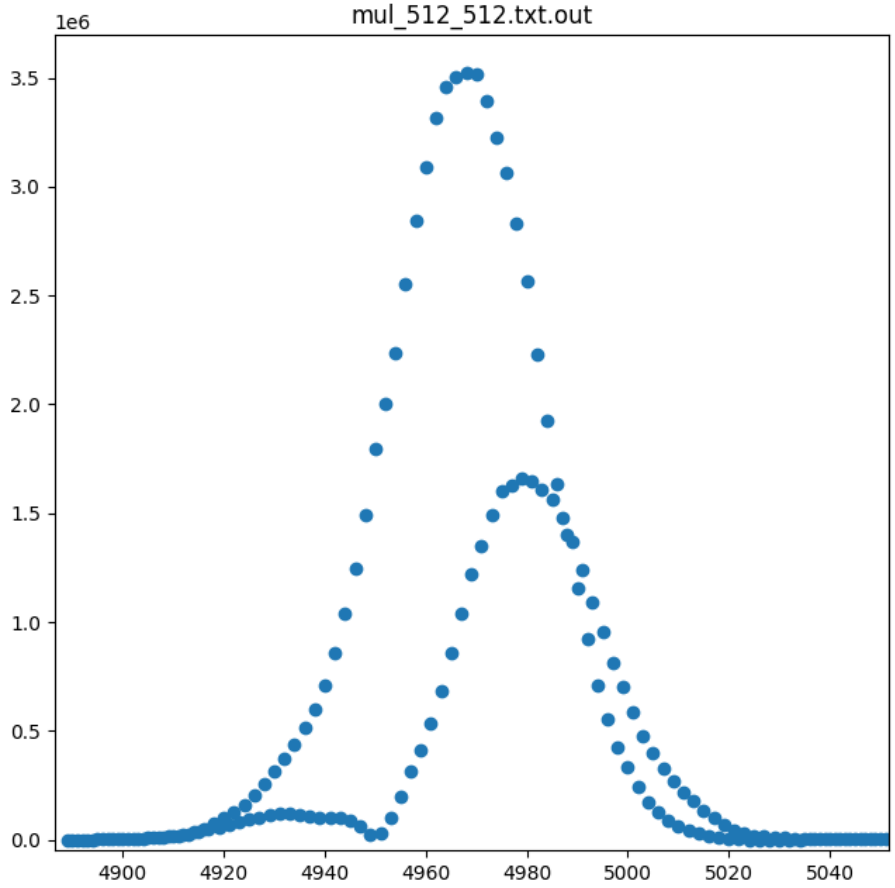


Тест умножение “в столбик”, ASM



Сравнение


mul_512_512.txt.out
350000
300000
250000
200000
150000
100000
50000
0
3600
3800
4000
4200
4400
4600
4800


mul_512_512.txt.out
1e6
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
4900
4920
4940
4960
4980
5000
5020
5040

Умножение алгоритмом Карацубы

Явной зависимости вывести не удалось, но примерно 92% всех делений при умножении чисел от 1 до 3000 разрядов были связаны со степенью двойки. Алгоритм предпочитал выбирать либо 2^n старших разрядов большего из перемножаемых чисел, либо 2^n младших. Остальные 8% делений не были связаны со степенью двойки и встречались при умножении как чисел маленького размера, так и большого.

Вывод

В ходе работы была изучена скорость необходимых алгоритмов существующей и популярной реализации длинной арифметики.
Был выбран, частично реализован и протестирован на корректность и скорость функционал будущей библиотеки для осуществления быстрых операций длинной арифметики.
В работе представлено подробное сравнение скоростей алгоритмов.
Определены пути развития работы.

Список литературы

[1] GNU MP The GNU Multiple Precision Arithmetic Library Edition 6.0.0 25 March 2014: https://gmplib.org/gmp-man-6.0.0a.pdf
[2] Intel® 64 and IA-32 Architectures Software Developer's Manual

Спасибо за внимание

Санкт-Петербургский государственный университет 2022