

Лабораторная работа №3: Организация памяти суперскалярных ЭВМ
Студент: Раващдех Ф.Х.
Группа: ИУ7-55Б

CPUID:
Vendor ID: "GenuineIntel"; CPUID level 22

Дополнительные функции Intel:

Версия 000806e9:

Type 0 - Original OEM

Family 6 - Pentium Pro

Model 14 -

Stepping 9

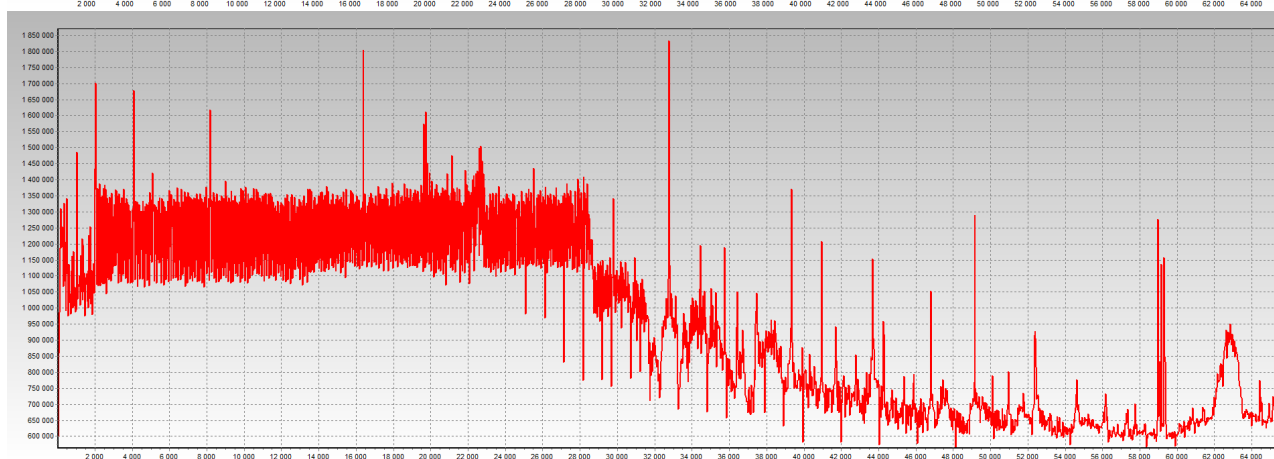
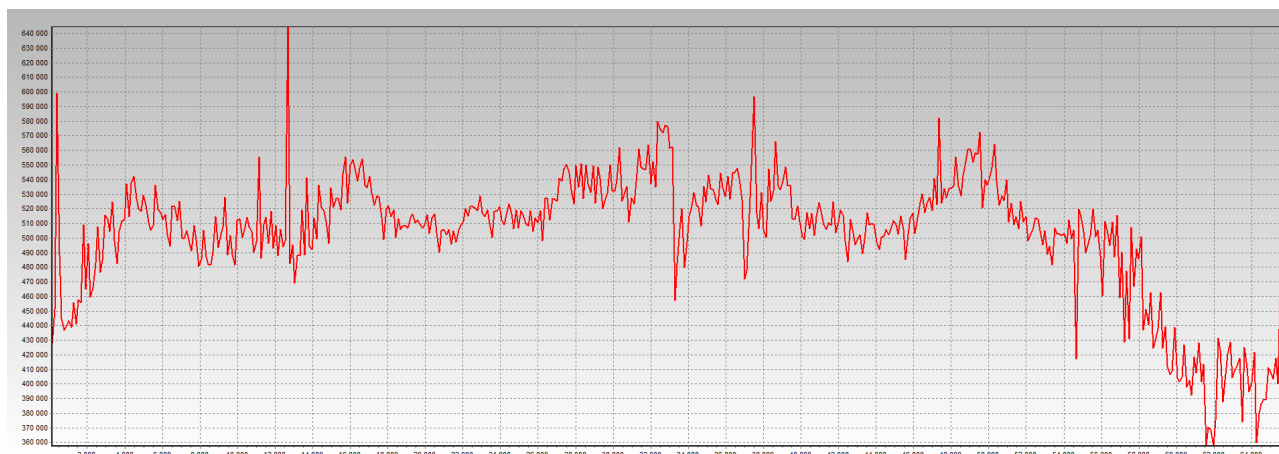
Reserved 32

Extended brand string: "Intel(R) Pentium(R) CPU 4415U @ 2.30GHz"

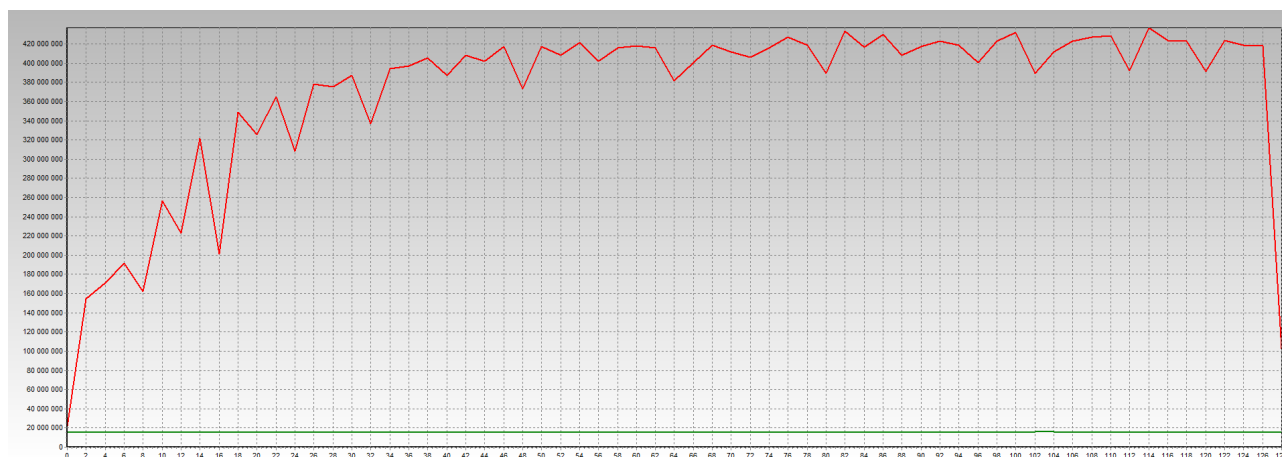
CLFLUSH instruction cache line size: 8

Initial APIC ID: 1

Hyper threading siblings: 16

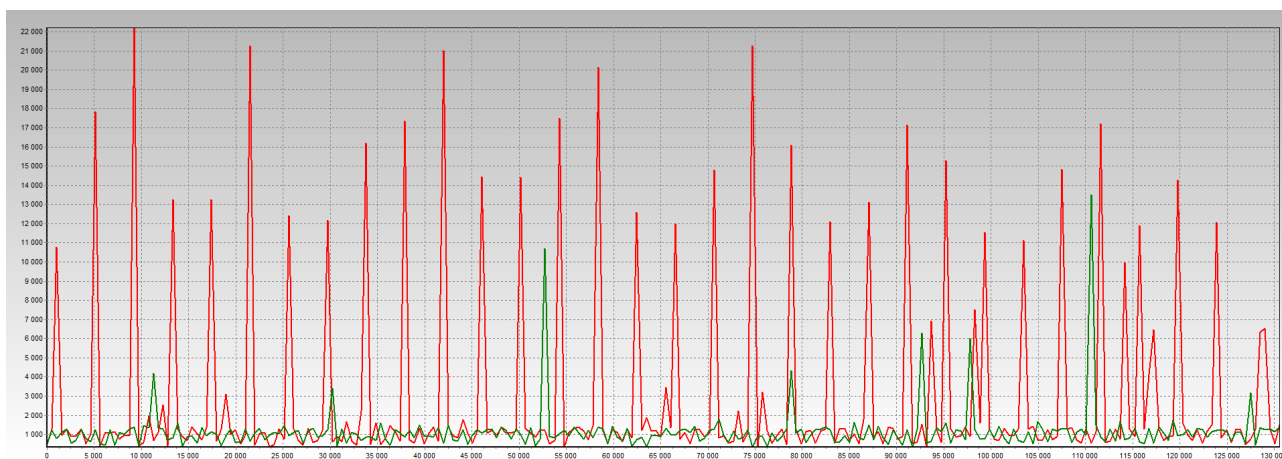


Вывод: неправильный выбор шага приведет к обращению к одному и тому же банку памяти, что медленнее, чем обращения к разным банкам памяти.

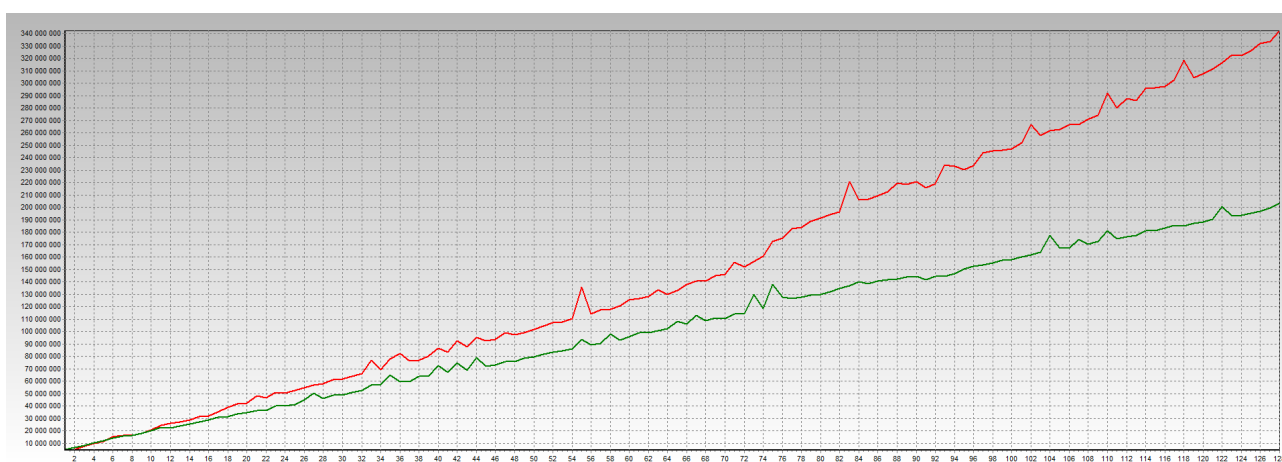


Список обрабатывался в 23,767485 раз дольше.

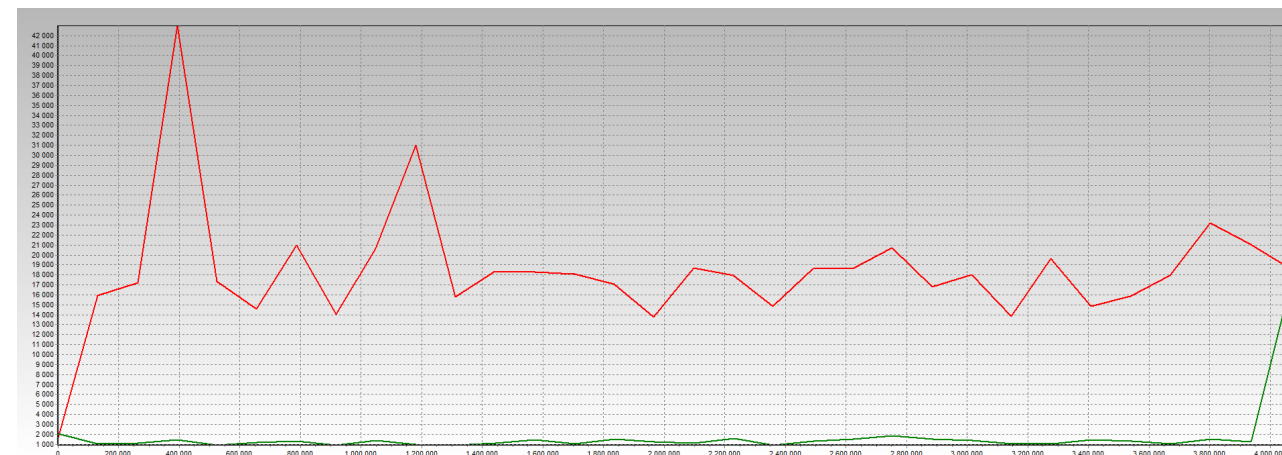
Вывод: векторные структуры данных обрабатываются процессором быстрее ссылочных.



Обработка без загрузки таблицы страниц в TLB производилась в 2,4999698 раз дольше.
Вывод: цикл предвыборки данных ускоряет обработку.



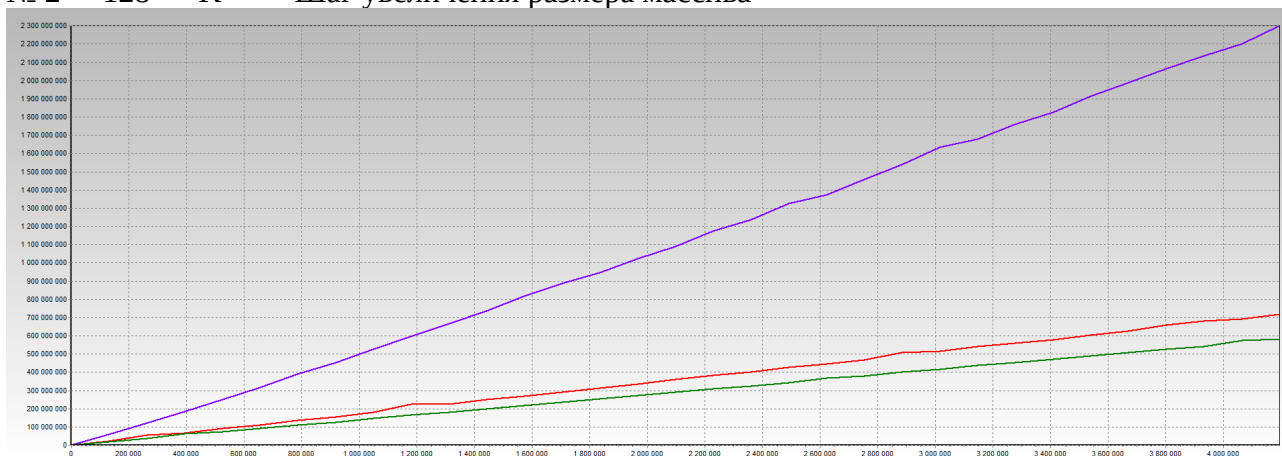
Неоптимизированная структура обрабатывалась в 1,4674585 раз дольше.
Вывод: пакетно-ориентированное программирование эффективнее.



Чтение с конфликтами банков производилось в 10,355072 раз дольше.
Вывод: не стоит ходить по памяти с большим шагом.

Параметры:

№ 1	4	М	Количество 64-х разрядных элементов массивов
№ 2	128	К	Шаг увеличения размера массива



QuickSort работал в 3,0835462 раз дольше Radix-Counting Sort.

QuickSort работал в 3,8176538 раз дольше Radix-Counting Sort, оптимизированного под 8-процессорную ЭВМ.

Вывод: Radix-Counting Sort работает за $O(N/\log N)$ и быстрее чем QuickSort.