|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 2 |

**Название:** Организация памяти конвейерных суперскалярных ЭВМ

**Дисциплина:** Организация ЭВМ и систем

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ6-72Б |  |  | С.В. Астахов | |
|  | (Группа) |  | (Подпись, дата) | | (И.О. Фамилия) |
|  |  |  |  | |  |
| Преподаватель |  |  |  | |  |
|  |  |  | (Подпись, дата) | | (И.О. Фамилия) |

Москва, 2022

**Введение**

**Цель работы:** освоение принципов эффективного использования подсистемы памяти современных универсальных ЭВМ, обеспечивающей хранение и своевременную выдачу команд и данных в центральное процессорное устройство.

**Задание 1**

Ознакомиться с возможностями программы PCLAB в Разделе 2 методических указаний. Запустить программу PCLAB 1.0. Изучить идентификационную информацию на вкладке «Идентификация процессора».

Идентификационная информация приведена в конце отчета.

**Задание 2**

На основании идентификационной информации о микропроцессоре ЭВМ, используемой при проведении лабораторной работы, определить следующие параметры: размер линейки кэш-памяти верхнего уровня и объем физической памяти. Результаты занести в отчет.

Длина строки - 128 Б (64 Б?).

Размер - 32 КБ.

**Задание 3**

Ознакомиться с описанием эксперимента «Исследование расслоения динамической памяти» на вкладке «Описание эксперимента». Провести эксперимент. По результатам эксперимента определить: количество банков динамической памяти; размер одной страницы динамической памяти; количество страниц в динамической памяти. Сделать выводы о использованном способе наращивания динамической памяти. Результаты занести в отчет.

Условия эксперимента:

                  - Единицы измерения по Ох - Байты

                  - Единицы измерения по Оу - такты

                  - Параметр1      :      32

                  - Параметр2      :      128

                  - Параметр3      :      1

Результаты проведенного эксперимента показаны на рисунке 1.

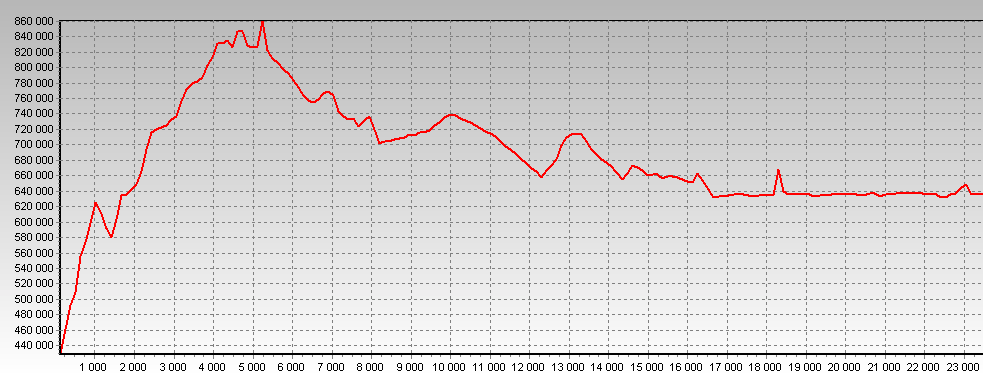


Рисунок 1 — Исследование расслоения динамической памяти

Т1 = 1024 — 1ый экстремум функции

П = 128 — размер линейки кэш-памяти

Б = Т1 / П = 1024 / 128 = 8 — число банков

Т2 = 4096 — глобальный экстремум функции

PC = T2 / Б = 4096 / 8 = 512 — размер одного банка

О = 64 К — полный объем памяти

С = О / (РС\*Б\*П) = 64К / (521\*8\*128) =

Вывод:

**Задание 4**

Ознакомиться с описанием эксперимента «Сравнение эффективности ссылочных и векторных структур данных». Провести эксперимент. По результатам эксперимента определить: отношение времени работы алгоритма, использующего зависимые данные, ко времени обработки аналогичного алгоритма обработки независимых данных. Сделать выводы об эффективности ссылочных и векторных структур данных и способах ее повышения. Результаты занести в отчет.

Условия эксперимента:

                  - Единицы измерения по Ох - Килобайты

                  - Единицы измерения по Оу - такты

                  - Параметр1      :      1

                  - Параметр2      :      32

                  - Параметр3      :      1

Результаты эксперимента представлены на рисунке 2.

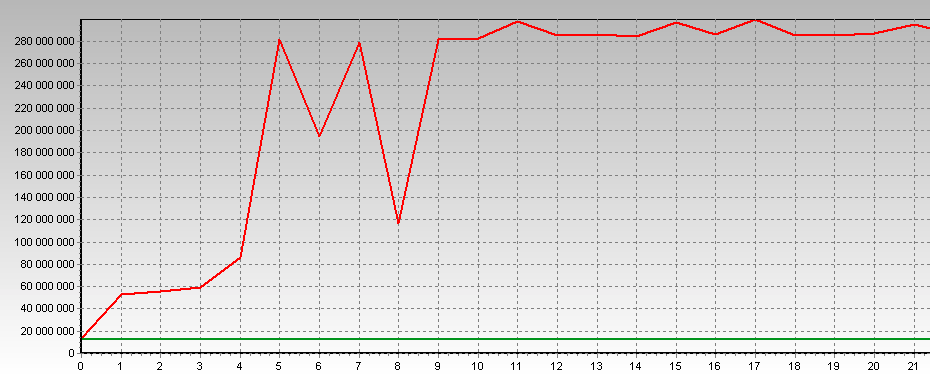


Рисунок 2 — Сравнение эффективности ссылочных и векторных структур данных

Список обрабатывался в 19,645843 раз дольше.

Вывод: по результатам эксперимента видно, что ссылочные структуры (особенно, с ростом фрагментации) требуют большего времени для обработки, чем векторные, так как в случае ссылочных структур происходит обработка зависимых данных, т.е. адрес загружаемого операнда становится известным только после обработки предыдущей команды.

**Задание 5**

Для ЭВМ, используемой при проведении лабораторной работы определить следующие параметры: степень ассоциативности и размер TLB данных. Ознакомиться с описанием и провести эксперимент «Исследование эффективности программной предвыборки». По результатам эксперимента определить: отношение времени последовательной обработки блока данных ко времени обработки блока с применением предвыборки; время и количество тактов первого обращения к странице данных. Сделать выводы об эффективности предвыборки и способах ее повышения. Результаты занести в отчет.

Условия эксперимента:

                  - Единицы измерения по Ох - Байты

                  - Единицы измерения по Оу - такты

                  - Параметр1      :      512

                  - Параметр2      :      64

Результаты эксперимента представлены на рисунке 3.

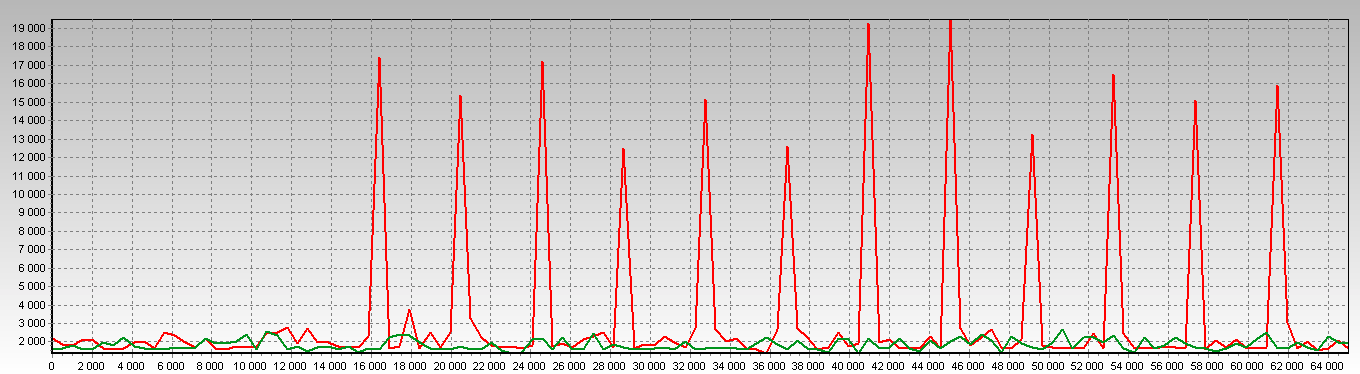


Рисунок 3 — Исследование эффективности программной предвыборки

Обработка без загрузки таблицы страниц в TLB производилась в 1,783955 раз дольше.

Число тактов первого обращения к странице данных — 17412.

Вывод: программная предвыборка позволяет сократить время доступа к оперативной памяти и стабилизировать задержку, исключая двойное обращение к оперативной памяти.

**Задание 6**

Ознакомиться с описанием и провести эксперимент «Исследование способов эффективного чтения оперативной памяти». По результатам эксперимента определить: отношение времени обработки блока памяти неоптимизированной структуры ко времени обработки блока структуры, обеспечивающей эффективную загрузку и параллельную обработку данных. Сделать выводы о способах повышения эффективности чтения оперативной памяти.

Условия эксперимента:

                  - Единицы измерения по Ох - Количество параллельных потоков

                  - Единицы измерения по Оу - такты

                  - Параметр1      :      2

                  - Параметр2      :      64

Результаты эксперимента представлены на рисунке 4.

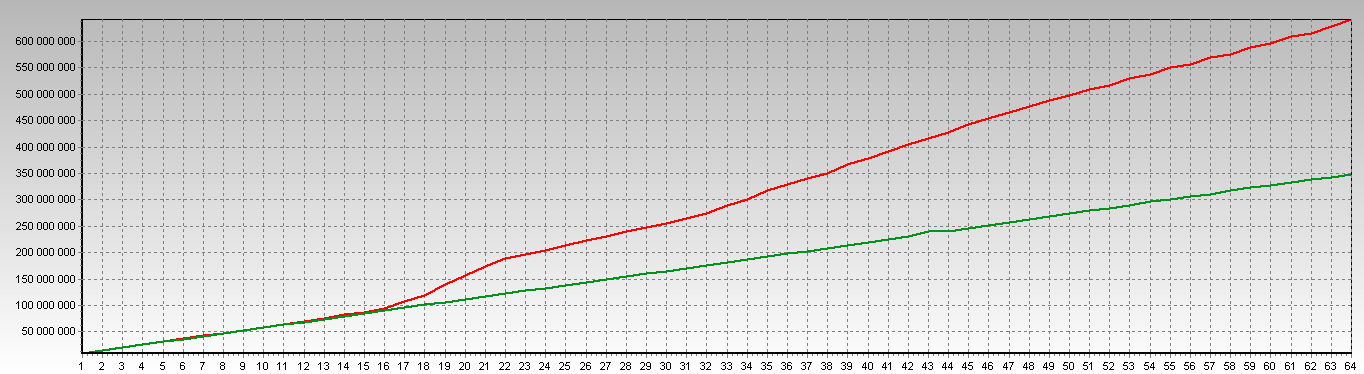


Рисунок 4 — Исследование способов эффективного    чтения оперативной памяти

Неоптимизированная структура обрабатывалась в 1,6762589 раз дольше.

Вывод: при использовании неоптимизированных структур данных они распределяются по пакетам некомпактно, что вызывает задержки в работе программы. Оптимизация структур данных, например, более «плотное» распределение данных по пакетам, позволяет ускорить программу.

**Задание 7**

Для ЭВМ, используемой при проведении лабораторной работы определить следующие параметры: размер банка кэш-памяти данных первого и второго уровня, степень ассоциативности кэш-памяти первого и второго уровня, размер линейки кэш-памяти первого и второго уровня. Ознакомиться с описанием и провести эксперимент «Исследование конфликтов в кэш-памяти». По результатам эксперимента определить: отношение времени обработки массива с конфликтами в кэш-памяти ко времени обработки массива без конфликтов. Сделать выводы о способах устранения конфликтов в кэш-памяти.

Условия эксперимента:

                  - Единицы измерения по Ох - Смещение от начала блока

                  - Единицы измерения по Оу - такты

                  - Параметр1      :      128

                  - Параметр2      :      128

                  - Параметр3      :      32

Результаты эксперимента представлены на рисунке 5.

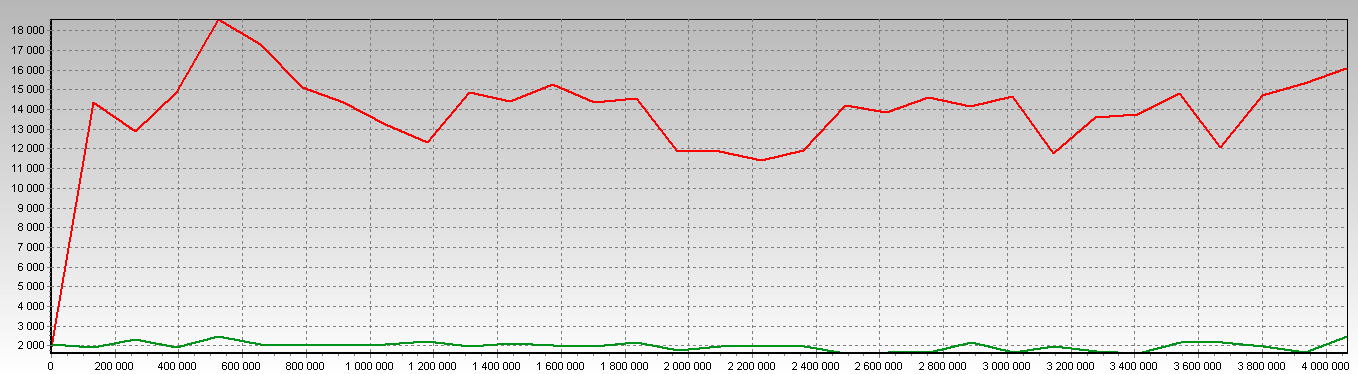


Рисунок 5 — Исследование конфликтов в кэш-памяти

Чтение с конфликтами банков производилось в 6,9114675 раз дольше.

Вывод: программа без конфликтов кеш-памяти (кеширующая данные не в один и тот же набор) работает значительно быстрее. Для оптимизации программы следует обращаться к памяти на расстоянии не кратном размеру банка.

**Задание 8**

Ознакомиться с описанием и провести эксперимент «Исследование алгоритмов сортировки». По результатам эксперимента определить: отношение времени сортировки массивов алгоритмом QuickSort ко времени сортировки алгоритмом Counting- Radix, а также ко времени сортировки Counting-Radix алгоритмом, оптимизированным под 8-процессорную вычислительную систему. Сделать выводы о наиболее эффективном алгоритме сортировки.

Условия эксперимента:

                  - Единицы измерения по Ох - Размер массива

                  - Единицы измерения по Оу - такты

                  - Параметр1      :      1

                  - Параметр2      :      128

Результаты эксперимента приведены на рисунке 6.

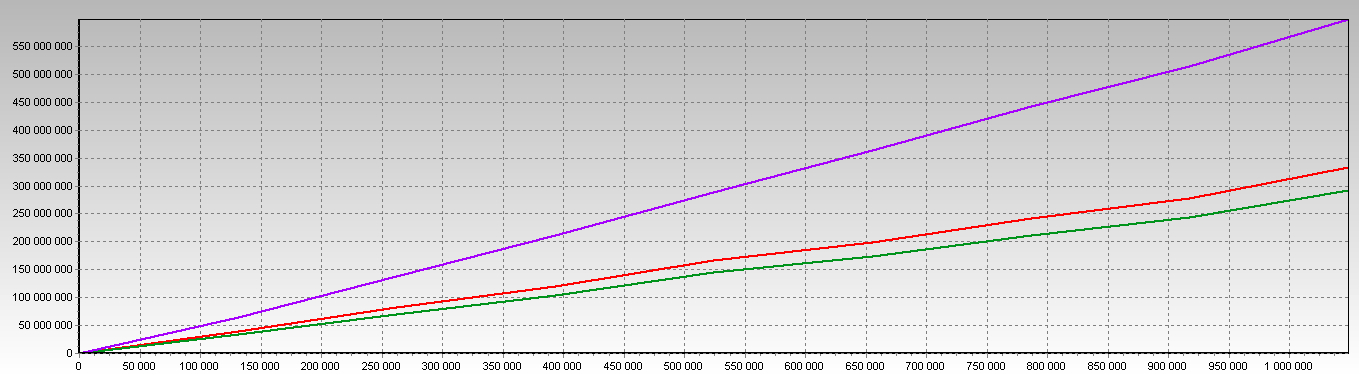


Рисунок 6 — Исследование алгоритмов сортировки

QuickSort работал в 1,7959224 раз дольше Radix-Counting Sort.

QuickSort работал в 2,0576808 раз дольше Radix-Counting Sort, оптимизированного под 8-процессорную ЭВМ.

Вывод: метод Radix-Counting имеет меньшую вычислительную сложность, чем QuickSort, так как работает без использования парных сравнений, однако, вычислительная сложность алгоритма существенно зависит от его настройки на определенную разрядность чисел и размерность массивов.

**Вывод:** результатом выполнения данной работы является освоение принципов эффективного использования подсистемы памяти современных универсальных ЭВМ, обеспечивающей хранение и своевременную выдачу команд и данных в центральное процессорное устройство.