|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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 Программное обеспечение ЭВМ и информационные технологии

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3**

**«ОБРАБОТКА РАЗРЕЖЕННЫХ МАТРИЦ»**

Студент Светличная Алина Алексеевна

Группа ИУ7 – 33Б

Проверил Барышникова Марина Юрьевна

*2021 г.*

# **Описание технического задания**

Условие задачи:

Разреженная (содержащая много нулей) матрица хранится в форме 3-х объектов:

- вектор A содержит значения ненулевых элементов;

- вектор IA содержит номера строк для элементов вектора A;

- связный список JA, в элементе Nk которого находится номер компонент

в A и IA, с которых начинается описание столбца Nk матрицы A.

1. Смоделировать операцию умножения вектора-строки и матрицы, хранящихся в этой форме, с получением результата в той же форме.

2. Произвести операцию умножения, применяя стандартный алгоритм работы с матрицами.

3. Сравнить время выполнения операций и объем памяти при использовании этих 2-х алгоритмов при различном проценте заполнения матриц.

Входные данные:

* Способ заполнения матрицы (самостоятельно и произвольно)
* Количество строк и столбцов матрицы
* Количество ненулевых элементов матрицы
* Количество ненулевых элементов вектора-строки
* Тройки чисел – строка, столбец, значение (при типе заполнения матрицы - самостоятельно)

Выходные данные:

Перемноженная матрица в стандартном виде, в разреженном виде

Действие программы (основное):

• Перемножает матрицы в разреженном виде

• Перемножает матрицы в обычном виде

• Замеряет скорость перемножения и затраты по памяти

Обращение к программе:

Запускается через терминал командой ./main.exe в директории с программой.

Возможные аварийные ситуации:

1. Некорректный способ заполнения матрицы
2. Некорректное количество строк или столбцов
3. Некорректное значение элемента

# **Описание структур данных**

**read\_type** – переменная для выбора заполнения матрицы вручную или произвольно

Поля структуры, описанной ниже, для хранения матрицы в стандартном виде:

matrix – техническая отрасль

x\_size – количество строк матрицы

y\_size – количество столбцов матрицы

unzero\_size – количество ненулевых элементов в матрице

**typedef struct**

**{**

**int \*\*matrix;**

**int x\_size;**

**int y\_size;**

**int unzero\_size;**

**} std\_matrix\_t;**

Поля структуры, описанной ниже, для хранения матрицы в разреженном виде:

\*a – массив указателей на ненулевые элементы матрицы

\*ia – массив указателей на строки элементов матрицы

\*ja – массив указателей на номера компонента в двух предыдущих массивах, с которых начинается описание i-го столбца

x\_size – количество строк матрицы

y\_size – количество столбцов матрицы

unzero\_size – количество ненулевых элементов в матрице

**typedef struct**

**{**

**int \*a;**

**int \*ia;**

**int \*ja;**

**int x\_size;**

**int y\_size;**

**int unzero\_size;**

**} sparse\_matrix\_t;**

Поля структуры, описанной ниже, для хранения вектора-строки:

story – признак сказки

poetry – признак стиха

**typedef struct**

**{**

**int \*arr;**

**int size;**

**} list\_t;**

# **Описание алгоритма**

Программа принимает способ заполнения матрицы (самостоятельно или произвольно), вызывается соответствующая функция заполнения матрицы в стандартном виде.

Вызывается функция преобразования стандартного формата матриц в разреженный.

Алгоритм перемножения стандартной формы матриц: запускается два цикла по столбцам и по строкам, в соответствующий элемент результирующего вектора записывается произведение элемента в данном столбце на элемент данного индекса в векторе.

Алгоритм перемножения разреженной формы матриц: запускается цикл по строкам и цикл от предыдущего элемента в массиве, описывающем начало каждого столбца, до следующего элемента этого же массива (данный цикл показывает, сколько ненулевых элементов есть в конкретном столбце матрицы), в соответствующий элемент результирующего вектора записывается произведение элемента из массива ненулевых элементов матрицы, определенного двумя циклами, и элемента вектора, соответствующего столбца (столбец определяется по массиву описания начала столбцов в разреженной форме вектора, учитывая, что в векторе лишь одна строка, то данный массив однозначно определяет столбец ненулевого элемента).

Оба алгоритма оборачиваются замерами времени, которые потом выводятся на экран (также выводится заполненность матриц и объем занимаемой памяти).

**Тесты**

|  |  |  |
| --- | --- | --- |
| Аварийная ситуация | Код ошибки | Уведомление |
| Тип чтения - 2 | ERROROR\_READ\_TYPE | Ошибка: тип заполнения матрицы введён неверно |
| Количество строк - ert | ERROR\_INPUT\_SIZES | Ошибка: неверно введены размеры матрицы |
| Количество столбцов - № | ERROR\_INPUT\_SIZES | Ошибка: неверно введены размеры матрицы |
| Количество ненулевых элементов матрицы – % | ERROR\_UNZERO\_SIZE | Ошибка: количество ненулевых элементов в матрице введено неверно |
| Количество ненулевых элементов вектора – ! | ERROR\_UNZERO\_SIZE | Ошибка: количество ненулевых элементов в векторе столбце введено неверно |
| При самостоятельном заполнении матрицы | | |
| Элемент матрицы - 89\* | ERROR\_ELEMENT | Ошибка: число введено неверно |
| Строка матрицы за границами размерности матрицы | ERROR\_X\_POS | Ошибка: строка матрицы введена неверно |
| Столбец матрицы за границами размерности матрицы | ERROR\_Y\_POS | Ошибка: столбец матрицы введен неверно |
| Элемент вектора - w | ERROR\_ELEMENT | Ошибка: элемент вектора введён неверно |
| Столбец вектора за границами размерности матрицы | ERROR\_Y\_POS | Ошибка: позиция элемента вектора введена неверно |

**Обработка результатов умножения**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Сравнение характеристик стандартной и разреженной матриц:** | | | | | | | | |
| Размеры | Плотность в % | | Время выполнения (в тактах) | | | Объем занимаемой памяти (в байтах) | | |
| Стандартная матрица | Разреженная матрица | Процентное отношение | Стандартная матрица | Разреженная матрица | Процентное отношение |
| 10x10 | 5 | | 1372 | 633 | 216% | 400 | 80 | 500% |
| 10 | | 1366 | 812 | 168% | 400 | 120 | 333% |
| 20 | | 1377 | 1069 | 128% | 400 | 200 | 200% |
| 50 | | 1538 | 1609 | 95% | 400 | 440 | 90% |
| 60 | | 1719 | 1776 | 96% | 400 | 520 | 77% |
|  | 75 | | 1467 | 1949 | 75% | 400 | 640 | 63% |
| 50х50 | 5 | | 25816 | 4547 | 567% | 10000 | 1200 | 833% |
| 10 | | 26991 | 6467 | 417% | 10000 | 2200 | 455% |
| 20 | | 28884 | 12076 | 239% | 10000 | 4200 | 238% |
| 50 | | 27122 | 20556 | 132% | 10000 | 10200 | 98% |
| 60 | | 25109 | 22696 | 111% | 10000 | 12200 | 82% |
| 75 | | 24212 | 26053 | 93% | 10000 | 15200 | 66% |
| 100х100 | 5 | | 120153 | 13893 | 865% | 40000 | 4400 | 909% |
| 10 | | 106760 | 19390 | 551% | 40000 | 8400 | 476% |
| 20 | | 113343 | 34992 | 324% | 40000 | 16400 | 244% |
| 50 | | 108122 | 76190 | 142% | 40000 | 40400 | 99% |
| 60 | | 121029 | 99618 | 121% | 40000 | 44400 | 90% |
| 75 | | 107082 | 113294 | 95% | 40000 | 60400 | 66% |
| 500х500 | 5 | | 3656579 | 189215 | 1932% | 1000000 | 102000 | 980% |
| 10 | | 3808161 | 377912 | 1007% | 1000000 | 202000 | 495% |
| 20 | | 3166628 | 618186 | 512% | 1000000 | 402000 | 249% |
| 50 | | 3660177 | 1644502 | 223% | 1000000 | 1002000 | 100% |
| 60 | | 3110544 | 1876220 | 166% | 1000000 | 1202000 | 83% |
| 75 | | 3469274 | 2396670 | 145% | 1000000 | 1502000 | 67% |
|  | | | | |  |  | |  |
| **Обработка результатов использования разреженной матрицы:** | | | | | | | | |
| Заполненность матрицы | | Выгодно по времени | | | | Выгодно по памяти | | |
| <50% | | + | | | | + | | |
| 50% | | + | | | | + | | |
| 75% | | для таблиц большой размерности | | | | - | | |

**Контрольные вопросы**

**1. Что такое разреженная матрица, какие способы хранения вы знаете?**

Разреженная матрица – это матрица, содержащая большое количество нулей. Способы хранения: связная схема хранения, строчный формат, линейный связный список, кольцевой связный список, двунаправленные стеки и очереди.

**2. Каким образом и сколько памяти выделяется под хранение разреженной и обычной матрицы?**

Под обычную матрицу выделяет N \* M ячеек памяти, где N – количество строк, а M – количество столбцов. Для разреженной матрицы – зависит от способа, но не менее 3\*K, где K – количество ненулевых элементов.

**3. Каков принцип обработки разреженной матрицы?**

Смысл всех алгоритмов работы с разреженными матрицами – обработка только ненулевых элементов, что значит, что и время работы, и память пропорциональна заполненности матрицы.

**4. В каком случае для матриц эффективнее применять стандартные алгоритмы обработки матриц? От чего это зависит?**

Как можно заметить по двум таблицам обработки разреженных матриц, представленных выше, алгоритмы обработки разреженных матриц безусловно выгодны, если заполненность матриц до 50%. Однако, если в программе не важна память, то можно использовать такие алгоритмы до заполненности 75%, так как скорость остается выгодной именно до этого рубежа.

# **Вывод**

Выводы об алгоритмах обработки разреженных матриц представлены в таблице№2 и контрольном вопросе№4.