МГТУ им. Н.Э. Баумана

**Лабораторная Работа № 3**

**Обработка разреженных матриц**

**Цель работы:** реализация алгоритмов обработки разреженных матриц, сравнение этих алгоритмов со стандартными алгоритмами обработки матриц при различном размере матриц и степени их разреженности.

Студент: Нгуен Ань Тхы

Группа : ИУ7 И - 36Б

1) описание условия 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 сложения двух матриц, хранящихся в этой форме, с получением результата в той же форме.

2. Произвести операцию сложения, 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.

2) описание ТЗ:

Исходные данные:

Размеры матрицы (2 натуральные числа n, m)

Список индексы строк и столбцов и значения соответствующих элементов в 2 матрицах (3 положительных целых числа по очереди i, j, value), индексы строк и столбцов в диапазон (0..n, 0..m)

Выходные данные:

Результат сложения 2-х матриц в форме матрицы и в форме 3-х объектов: вектор A, вектор IA и связный список JA.

Таблицы с результатами измерений времени и памяти.

Соответствующее сообщение об ошибке

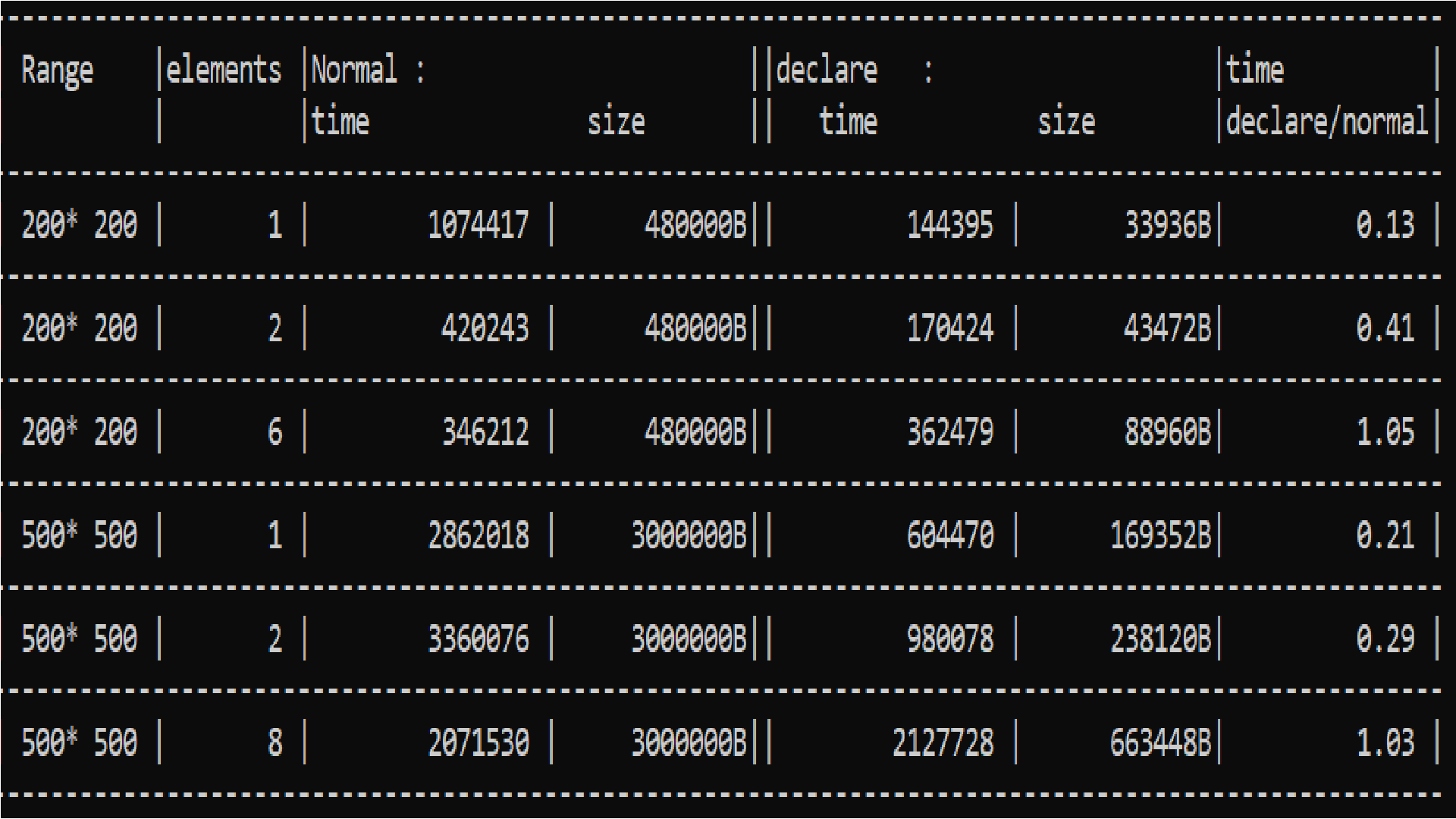
3) описание алгоритма:

Использовать линейный связный список для хранения положения и значения ненулевых элементов в матрице.

Выбрать кажыдй ненулевый элемент из 2 списки в очереди, проверять если их положение в матрице совпадать то вычислить их сумму и добавить в списке, иначе просто добавить элемент в списке

4) набор тестов, с указанием, что проверяется:

|  |  |
| --- | --- |
| Size of matrix: -1 2 | Input ERR |
| (i j value) : -1 2 3 | ERR wrong index |
| (i j value) : 2 2 e | Input ERR |
| Size of matrix: 5 6  1 0 4  3 0 5  2 0 4  3 2 4  4 2 5  2 2 6  4 3 5  2 3 6  3 5 3  2 5 8  0 0 0  1 3 4  2 3 5  3 3 6  1 1 6  2 1 7  3 1 8  3 4 5  4 5 6  3 5 7  2 5 8  0 0 0 | Result:  0 0 0 0 0 0  4 6 0 4 0 0  4 7 6 11 0 16  5 8 4 6 5 10  0 0 5 5 0 6  A: 4 4 5 6 7 8 6 4 5 4 11 6 5 5 16 10 6  IA: 1 2 3 1 2 3 2 3 4 1 2 3 4 3 2 3 4  ANj: 0 1 2 3 4 5  AN: 0 3 6 9 13 14 |

5) таблицы с результатами измерений времени и памяти:

Вывод:

Для матрицы размер 200 \* 200:

Операция сложения над матрицами с заполнением до 1% элементов выполняется быстрее при хранении ее как разреженной. При этом требуется в 14 раз меньше памяти.

Операция сложения над матрицами с заполнением до 2% элементов выполняется быстрее при хранении ее как разреженной. При этом требуется в 11 раз меньше памяти.

Операция сложения над матрицами с заполнением до 6% элементов выполняется равно при хранении ее как разреженной. При этом требуется в 5 раз меньше памяти.

Для матрицы размер 500 \* 500:

Операция сложения над матрицами с заполнением до 1% элементов выполняется быстрее при хранении ее как разреженной. При этом требуется в 17 раз меньше памяти.

Операция сложения над матрицами с заполнением до 2% элементов выполняется быстрее при хранении ее как разреженной. При этом требуется в 12 раз меньше памяти.

Операция сложения над матрицами с заполнением до 8% элементов выполняется равно при хранении ее как разреженной. При этом требуется в 4 раз меньше памяти.

Алгоритмы работы с разреженными матрицами дают заметный выигрыш во времени при обработке достаточно больших матриц с малым заполнением.

6) Контрольные вопросы:

1. Что такое разреженная матрица, какие схемы хранения таких матриц вы знаете?

Разреженная матрица – матр., большая часть которой заполнена нулями.

Можно хранить индекс (i,j) ненулевого элемента и его значение .

Можно хранить значения ненулевых элементов, их индексы строки/столбца, номера элементов, с которых начинается очередная строка.

2. Каким образом и сколько памяти выделяется под хранение разреженной и обычной матрицы.  
Под обычную матрицу выделяется n\*m\*sizeof(element) байт памяти.   
Требуемая память под хранение разреженной матрицы зависит от выбранного типа хранения, количества ненулевых элементов.

3. Каков принцип обработки разреженной матрицы?

Обрабатываются только ненулевые элементы, что позволяет сократить время обработки.

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

Эффективнее применять стандартные алгоритмы обработки матриц при достижении определенного уровня заполненности матрицы.