

**Министерство образования и науки 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**

### Студент Завойских Евгения Васильевна

*фамилия, имя, отчество*

### Группа ИУ7-33Б

Студент Завойских Е.В.

*подпись, дата фамилия и.о.*

### Руководитель практики Барышникова М.Ю.

*подпись, дата фамилия, и.о.*

### Оценка

*2021 г.*

**Условие задачи**

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

- вектор A содержит значения ненулевых элементов;

- вектор JA содержит номера столбцов для элементов вектора A;

- связный список IA, в элементе Nk которого находится номер компонент в A и JA, с которых начинается описание строки Nk матрицы A.

1. Смоделировать операцию умножения матрицы и вектора-столбца, 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.

**Внешняя спецификация программы**

1. Входные данные

При вызове программы выводится меню, каждый раз пользователь выбирает один из пунктов.

Матрица и вектор-столбец вводятся с клавиатуры или считываются из файла в формате:

* Число строк и столбцов матрицы, количество ненулевых элементов в матрице и векторе;
* Индекс строки, столбца и значение очередного ненулевого элемента матрицы;
* Индекс строки и значение очередного ненулевого элемента вектора.

Матрица и вектор-столбец также могут заполняться случайными числами, формат ввода с клавиатуры:

* Число строк и столбцов матрицы, количество ненулевых элементов в матрице и векторе.

1. Выходные данные

На экран выводятся исходные матрица и вектор-столбец в обычном формате (в виде матрицы), а также в заданной по условию форме. После умножения выводится результат использования двух способов умножения. При вызове вывода со сравнением способов умножения по времени и памяти выводится таблица со столбцами размерности матрицы и вектора-столбца, процента наполнения, времени выполнения обоих способов умножения и затрат по памяти.

В случае некорректного ввода команды меню выводится сообщение об ошибке и предоставляется возможность выбрать команду еще раз. В случае некорректного ввода матрицы или вектора-столбца также выводится соответствующее сообщение об ошибке. При ошибке открытия файла и выделения памяти выводится сообщение об ошибке. При получение нулевой матрицы при умножении выводится соответствующее сообщение.

1. Задача, реализуемая программой

Данная программа предназначена для работы с матрицей и вектором-столбцом и представляет собой приложение со следующими возможными операциями:

1. Ввести матрицу и вектор-столбец с клавиатуры.
2. Считать матрицу и вектор-столбец из файла.
3. Заполнить матрицу и вектор-столбец случайными числами.
4. Произвести умножение двумя способами.
5. Вывести результаты умножения двумя способами.
6. Вывести результаты сравнения двух способов умножения.
7. Вывести матрицу и вектор-столбец в обычном виде.
8. Вывести матрицу и вектор-столбец в разреженном виде.

1. Способ обращения к программе

Программа может быть вызвана из консоли или запущена в любой среде разработки, поддерживающей язык С.

1. Возможные аварийные ситуации и ошибки пользователя

В случае аварийной ситуации программа выводит сообщение о произошедшей ошибке. Ошибка может возникнуть в результате некорректного ввода (не соответствующего формату), при получении нулевой матрицы при умножении, присутствии некорректных данных в файле, открытии файла, выделении динамической памяти.

**Внутренние структуры данных**

В программе используются структуры для хранения матрицы/вектора в обычном виде и разреженном.

// структура для определения обычной матрицы

struct matrix

{

    int rows; // число строк матрицы

    int columns; // число столбцов матрицы

    int not\_zero; // число ненулевых элементов

    int \*\*matrix; // сама матрица

};

typedef struct matrix matrix\_s;

// структура для определения разреженной матрицы

struct sparse\_matrix

{

    int rows; // число строк матрицы

    int columns; // число столбцов матрицы

    int not\_zero; // число ненулевых элементов

    int \*a; // массив ненулевых элементов матрицы

    int \*ja; // массив индексов столбцов ненулевых элементов

    int \*ia; // массив индексов в a и ja, с которых начинается i-ая строка

};

typedef struct sparse\_matrix sparse\_matrix\_s;

**Алгоритм**

Пока пользователь не решит завершить программу, выполняется действие, выбранное из меню:

1. Ввести матрицу и вектор-столбец с клавиатуры: с клавиатуры вводятся размеры матрицы, количество ненулевых элементов в матрице и векторе, далее в каждой строке вводится сначала индекс строки, столбца и значение очередного ненулевого элемента матрицы, потом – индекс строки и значение очередного ненулевого элемента вектора; затем по обычной матрице и вектору формируются разреженные матрица и вектор
2. Считать матрицу и вектор-столбец из файла: вводится имя файла, из него считываются размеры матрицы, количество ненулевых элементов в матрице и векторе, далее в каждой строке считываются сначала индекс строки, столбца и значение очередного ненулевого элемента матрицы, потом – индекс строки и значение очередного ненулевого элемента вектора; затем по обычной матрице и вектору формируются разреженные матрица и вектор
3. Заполнить матрицу и вектор-столбец случайными числами: с клавиатуры вводятся размеры матрицы, количество ненулевых элементов в матрице и векторе, соответствующее количество элементов в матрице и векторе заполняется случайными числами, потом каждый второй элемент меняется с симметричным; затем по обычной матрице и вектору формируются разреженные матрица и вектор
4. Произвести умножение двумя способами: если матрица и вектор уже заполнены, выполняется сначала умножение обычным способом (каждая строка матрицы умножается на вектор-столбец поэлементно, произведения складываются), затем – в разреженном формате (просматриваются все ненулевые элементы очередной строки матрицы, смотрят индекс столбца очередного элемента, если в строке вектора с таким же индексом есть ненулевой элемент – элементы перемножаются и добавляются к сумме в соответствующем элементе вектора результата)
5. Вывести результаты умножения двумя способами: если умножение было выполнено, выводится сначала результат обычного способа (в виде матрицы), затем – в разреженном формате
6. Вывести результаты сравнения двух способов умножения: выводится таблица со столбцами размерности матрицы и вектора-столбца, процента наполнения, времени выполнения обоих способов умножения и затрат по памяти. Матрица и вектор заполняются случайными числами
7. Вывести матрицу и вектор-столбец в обычном виде: матрица выводится в виде обычной матрицы, вектор тоже (если они заполнены)
8. Вывести матрицу и вектор-столбец в разреженном виде: матрица и вектор выводятся в разреженном формате (если они заполнены)

**Тесты**

Негативы:

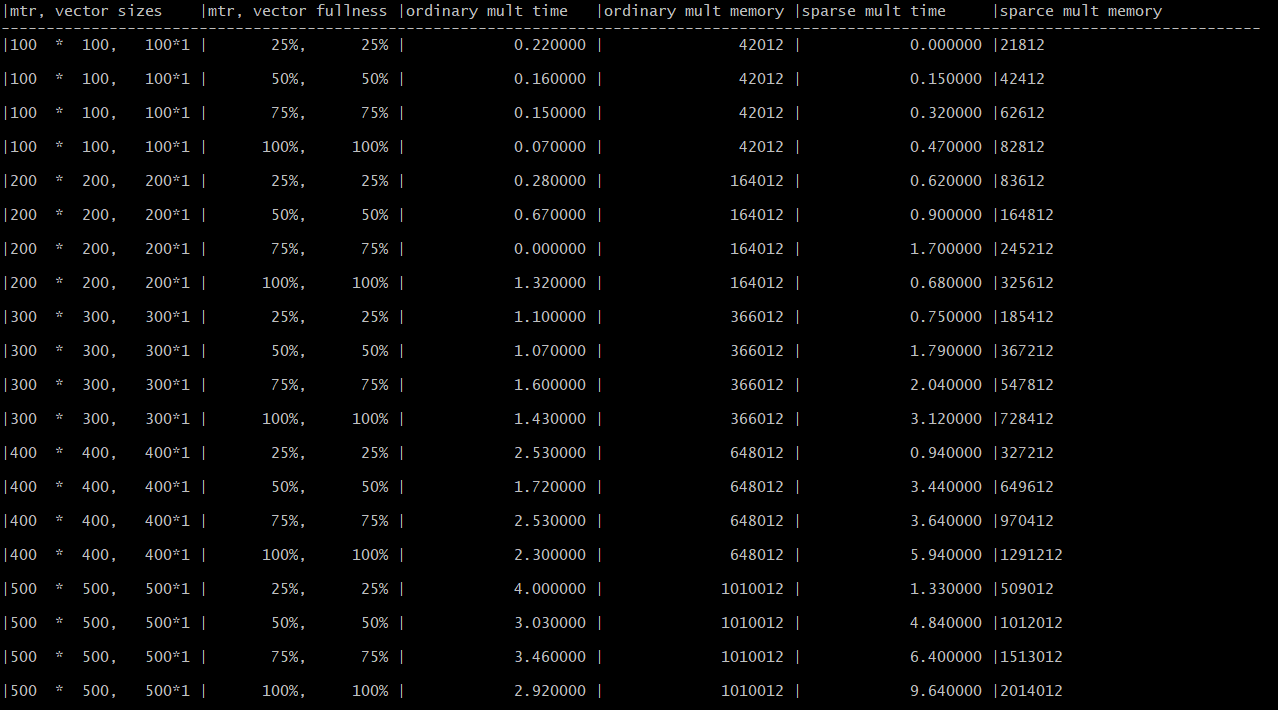
|  |  |  |
| --- | --- | --- |
| Входные данные | Вывод | Что проверяется |
| Возможные действия (введите номер нужной команды):  0. Завершить работу  1. Ввести матрицу и вектор-столбец с клавиатуры  2. Считать матрицу и вектор-столбец из файла  3. Заполнить матрицу и вектор-столбец случайными числами  4. Произвести умножение двумя способами  5. Вывести результаты умножения двумя способами  6. Вывести результаты сравнения двух способов умножения  7. Вывести матрицу и вектор-столбец в обычном виде  8. Вывести матрицу и вектор-столбец в разреженном виде  -10 | Нет команды с введенным номером | Неверно введена команда меню |
| 1  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: -9 9 3 3 | Ошибка чтения | Некорректный размер |
| 1  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: 2 3 0 2 | Ошибка чтения | Некорректное количество ненулевых элементов |
| 1  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: 2 3 4 2  Введите индекс строки, столбца и значение элемента матрицы: 0 6 2 | Ошибка чтения | Некорректный индекс |
| 1  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: 1 2 1 1  Введите индекс строки, столбца и значение элемента матрицы: 0 0 0 | Ошибка чтения | Некорректное значение элемента |
| 2  Введите название файла (не более 20 символов): 2.txt | Ошибка чтения | Пустой файл |
| 2  Введите название файла (не более 20 символов): 3.txt | Не удалось открыть указанный файл | Файл не существует |
| 2  Введите название файла (не более 20 символов): aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa.txt | Ошибка чтения | Название файла слишком длинное |
| 1  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: 1 2 2 2  Введите индекс строки, столбца и значение элемента матрицы: 0 0 -1  Введите индекс строки, столбца и значение элемента матрицы: 0 1 1  Введите индекс строки и значение элемента вектора-столбца: 0 1  Введите индекс строки и значение элемента вектора-столбца: 1 1  4 | Матрица получилась нулевой | В результате умножения получается нулевая матрица |

Позитивы:

|  |  |  |
| --- | --- | --- |
| Входные данные | Вывод | Что проверяется |
| 1  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: 2 3 4 1  Введите индекс строки, столбца и значение элемента матрицы: 0 0 1  Введите индекс строки, столбца и значение элемента матрицы: 0 2 45  Введите индекс строки, столбца и значение элемента матрицы: 1 1 1  Введите индекс строки, столбца и значение элемента матрицы: 1 2 2  Введите индекс строки и значение элемента вектора-столбца: 0 12 | Матрица  1 0 45  0 1 2  Вектор-столбец  12  0  0 | Создание матрицы и вектора с клавиатуры |
| 2  Введите название файла (не более 20 символов): 1.txt  В файле:  3 3 6 2  1 0 6  1 1 13  1 2 -8  0 1 2  2 1 4  2 2 5  0 2  2 6 | Матрица  0 2 0  6 13 -8  0 4 5  Вектор-столбец  2  0  6 | Считывание матрицы и вектора из файла |
| 3  Введите размеры матрицы, количество ненулевых элементов в матрице и векторе-столбце: 4 5 10 3 | Матрица  0 68 0 1 0  25 0 59 0 65  0 63 0 79 0  70 0 35 0 42  Вектор-столбец  0  46  82  0  6 | Заполнение матрицы случайными числами |
| 4  Возможные действия (введите номер нужной команды):  0. Завершить работу  1. Ввести матрицу и вектор-столбец с клавиатуры  2. Считать матрицу и вектор-столбец из файла  3. Заполнить матрицу и вектор-столбец случайными числами  4. Произвести умножение двумя способами  5. Вывести результаты умножения двумя способами  6. Вывести результаты сравнения двух способов умножения  7. Вывести матрицу и вектор-столбец в обычном виде  8. Вывести матрицу и вектор-столбец в разреженном виде  5 | Результат обычного умножения:  3128  5228  2898  3122  Результат в разреженном виде:  3128 5228 2898 3122  0 0 0 0  0 1 2 3 4 | Умножение двумя способами |
| 7 | Матрица  0 68 0 1 0  25 0 59 0 65  0 63 0 79 0  70 0 35 0 42  Вектор-столбец  0  46  82  0  6 | Вывод матрицы и вектора в обычном виде |
| 8 | Матрица  68 1 25 59 65 63 79 70 35 42  1 3 0 2 4 1 3 0 2 4  0 2 5 7 10  Вектор-столбец  46 82 6  0 0 0  0 0 1 2 2 3 | Вывод матрицы и вектора в разреженном виде |
| 6 | 1.PNG | Вывод таблицы по сравнению двух способов умножения |

**Вывод**

Замеры времени (в тактах) и замеры памяти (в байтах) двух способов умножения:



При разреженности в 75% требуется в 2 раза меньше памяти на хранение разреженных матриц, чем на хранение обычных; по времени умножение разреженных происходит быстрее, чем обычных.

При разреженности в 50% затраты по памяти для обоих способов примерно равны, как и по времени.

При разреженности в 25% памяти для хранения разреженных матриц нужно в 1,5 раза больше, чем для обычных; умножение разреженных матриц занимает больше времени, чем умножение обычных.

При отсутствии нулевых элементов памяти для хранения разреженных матриц нужно в 2 раза больше, чем для обычных; умножение разреженных матриц занимает больше времени, чем умножение обычных.

По мере увеличения размера матриц и уменьшения разреженности затраты по времени для обычного метода становятся все меньше затрат на разреженный метод.

По итогу, разреженный метод хранения и умножения выгодно использовать при разреженности более 50%, иначе и по времени, и по памяти выгоднее использовать обычный.

**Ответы на вопросы**

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

Разреженная матрица – матрица с большим числом нулевых элементов (число ненулевых элементов в матрице порядка n может выражаться как n в степени 1+g , где g лежит в интервале 0.2 … 0.5). Способы хранения: линейный и кольцевой связные списки, связные схемы разреженного хранения, разреженный строчный формат, двунаправленные стеки и очереди.

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

Под обычную матрицу выделяется n\*m\*sizeof(element) байт, где n – строки, m – столбцы, а element – тип данных матрицы. Для разреженной матрицы память выделяется в зависимости от способа. В моей задаче требуется (2\*k + n + 1)\*sizeof(element) байт, где n – число строк матрицы, k – количество ненулевых элементов.

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

Алгоритмы обработки разреженных матриц предусматривают действия только с ненулевыми элементами и, таким образом, количество операций будет пропорционально количеству ненулевых элементов.

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

Разреженность матрицы следует учитывать только в том случае, если из этого можно извлечь выгоду за счёт игнорирования нулевых элементов. В моей задаче стандартный алгоритм эффективнее при разреженности менее 50%.