



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

По дисциплине «Типы и структуре данных»

Название Обработка разреженных матриц

Студент Мамврийский Иван Сергеевич
фамилия, имя, отчество

Группа ИУ7-36Б

Студент

Мамврийский
И.С.

подпись, дата

фамилия, и.о.

Преподаватель

Никульшина Т.А.

подпись, дата

фамилия, и.о.

2022 г.

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

Условие задачи

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

- вектор A содержит значения ненулевых элементов;*
- вектор JA содержит номера строк 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.*

Требование к работе с программой

- Взаимодействие с программой строго по меню.

```
-----Меню-----
1 – Ввести матрицу
2 – Ввести вектор–столбец
3 – Умножить матрицу на вектор
4 – Вывести матрицы и вектор
5 – Замерить эффективность
0 – Выход

Введите команду: █
```

- При неправильно введенном номере программа предлагает выбрать номер пункта.
- Чтобы выйти из программы необходимо ввести «0».
- При вводе существующих пунктов пользователю предоставляются данные, либо выводится подменю, либо выходит.
- Работа в подменю(пункты 1, 2 и 3) осуществляется, так же как и в основном меню

```
Введите команду: 1

1 – Ввести матрицу в обычном виде
2 – Ввести матрицу в разреженном виде
3 – Автозаполнение матрицы
0 – Вернуться в главное меню

Введите команду: █
```

```
Введите команду: 2

1 – Ввести столбец–вектор
2 – Автозаполнение столбца–вектора
0 – Вернуться в главное меню

Введите команду: █
```

```
Введите команду: 3

1 – Умножение простой матрицы на вектор–столбец
2 – Умножение разреженной матрицы на вектор–столбец
0 – Выход в главное меню

Введите команду: █
```

- Чтобы ввести матрицу или вектор в любом виде, заполняются все поля ввода правильно (при некорректных случаях выводится сообщение об ошибке и предоставляется ввод этого поля заново).
- Вывод обычной матрицы осуществляется до размерности столбцов или строк 50 элементов

Алгоритм работы программы:

Создание обычной матрицы:

Для создания обычной матрицы вводится количество строк и столбцов. Заполнение обычной матрицы осуществляется по вводу всех элементов подряд. Тогда для разреженной матрицы при создании обычной считается количество ненулевых элементов. Выделяется память под ненулевые элементы. И заполняется, проходя по обычной матрице, забирая только ненулевые элементы.

Создание разреженной матрицы:

Для создания разреженной матрицы сначала создается обычная матрицы. Далее программа проходит по обычной матрицы. В массив индексов строк записывается номер первого ненулевого элемента в строке, в массив индексов столбцов записывается номер столбца, в котором находится ненулевой элемент. В массив элементов записывается сам ненулевой элемент.

Умножение обычной матрицы происходит следующим образом:

Берется первая строка и вектор-столбец. Происходит умножение элементов строки и вектора, то есть 1 элемент строки умножается на 1 элемент вектора-столбца. Считается сумма произведений. Данная сумма записывается в новую матрицу. Поиск сумм происходит по всем строкам матрицы.

Умножение разреженной матрицы происходит следующим образом:

В начале в JA просматриваются два первых элемента и если они отличаются друг от друга, то есть второй элемент больше первого, что означает, что в строке есть ненулевые элементы, при этом разница между 1 и 2 элементов – количество ненулевых элементов в строке. Затем из массива номеров столбцов JA просматривается индекс столбца нужного ненулевого элемента, после чего по этому индексу столбца смотрится существует ли ненулевой элемент в векторе-столбце такой индекс строки, если да, то происходит умножение элементов. Общий результат умножения строки матрицы на вектор-столбец, отличного от нуля записывается в определенным индексом строки в новый вектор-столбец.

Аварийные выходы из программы

Выходов аварийных нет, в том случае, если программа не будет принудительна закрыта.

Выход осуществляется только пунктом меню «0»

В ошибочных случаях выводится сообщение об ошибке и ввод повторяется, пока программа не получит корректные данные.

Тестовые случаи

Ввод	Вывод	Проверяемый случай
Основное меню: с	Данные введены неверно.	Правильность ввода пункта меню
Основное меню: 15	Данные введены неверно.	Правильность ввода пункта меню
Подменю: с	Данные введены неверно.	Правильность ввода пункта подменю
Подменю: 5	Данные введены неверно.	Правильность ввода пункта подменю
Создание обычной матрицы: -1 100	Данные введены неверно	Правильность ввода размерностей матрицы
Создание обычной матрицы: 100 1001	Данные введены неверно	Правильность ввода размерностей матрицы
Ввод обычной матрицы: d	Число введено неверно	Правильность ввода элементов обычной матрицы
Создание разреженной матрицы: -1 100	Данные введены неверно	Правильность ввода размерностей разреженной матрицы
Создание разреженной матрицы: 100 1001	Данные введены неверно	Правильность ввода размерностей разреженной матрицы
Ввод разреженной матрицы: d	Число введено неверно	Правильность ввода элементов разреженной матрицы
Создание разреженной матрицы: -1	Данные введены неверно	Правильность ввода количества ненулевых элементов разреженной матрицы

Создание разреженной матрицы (повторный ввод элемента): 1 1 2	Данные уже были введены	Повторение вводимых данных по строке и столбцу
Создание разреженной матрицы (n = 10, m = 10): 11 10 6	Номер строки или столбца введен неверно	Проверка на правильность ввода номеров строки и столбца
Создание вектора-столбца: 1001	Данные введены неверно	Правильность ввода размерностей вектора-столбца
Создание вектора-столбца: -1	Данные введены неверно	Правильность ввода размерностей вектора-столбца
Ввод вектора-столбца: d	Число введено неверно	Правильность ввода элементов вектора-столбца
Умножение матриц: Размерности матрицы: 10 100 Размерности вектора столбца: 101	Умножение провести невозможно разные размерности.	
Вывод данных: не введен вектор-столбец	Вектор-столбец не был введен	Вывод данных при их отсутствии
Вывод данных: не введена матрицы	Матрицы не была введена	Вывод данных при их отсутствии
Вывод данных: размерность > 50	Размер матрицы слишком большой. Выведена не будет.	
Умножение матриц: Матрица: 1 2 3 4 Вектор-столбец: 1 2	5 11	Правильность умножения матриц.
Умножение матриц: Матрица: 0 2 3 0 Вектор-столбец: 0 2	4 0	Правильность умножения матриц.

Матрицы была введена	Вывод обычной матрицы: 7 49 73 58	Вывод данных
Ввод матрицы: Введите количество строк матрицы: 2 Введите количество столбцов матрицы: 2 Ввод элементов матрицы: Введите 1 строку 1 столбец. 1 Введите 1 строку 2 столбец. 2 Введите 2 строку 1 столбец. 3 Введите 2 строку 2 столбец. 4	Вывод обычной матрицы: 1 2 3 4	Правильность ввода данных

Заполненность 90%:

```
Размерность: 500
Время умножения простой матрицы: 0.001481с
Память затраченная на простую матрицу: 1000000
Память затраченная на простую матрицу-столбец: 2000

Время умножения разреженной матрицы: 0.002243с
Память затраченная на разреженную матрицу: 1802000
Память затраченная на разреженную матрицу-столбец: 3776
```

```
Размерность: 1000
Время умножения простой матрицы: 0.004163с
Память затраченная на простую матрицу: 4000000
Память затраченная на простую матрицу-столбец: 4000

Время умножения разреженной матрицы: 0.005652с
Память затраченная на разреженную матрицу: 7204000
Память затраченная на разреженную матрицу-столбец: 7544
```

Заполненность 68%:

```
Размерность: 500
Время умножения простой матрицы: 0.001524с
Память затраченная на простую матрицу: 1000000
Память затраченная на простую матрицу-столбец: 2000

Время умножения разреженной матрицы: 0.001436с
Память затраченная на разреженную матрицу: 1362000
Память затраченная на разреженную матрицу-столбец: 318
```

```
Размерность: 1000
Время умножения простой матрицы: 0.003871с
Память затраченная на простую матрицу: 4000000
Память затраченная на простую матрицу-столбец: 4000

Время умножения разреженной матрицы: 0.003589с
Память затраченная на разреженную матрицу: 5444400
Память затраченная на разреженную матрицу-столбец: 6372
```


В результате измерений, были получены выше предоставленные данные, по которым можно сделать вывод, что каждый из 2-х методов эффективный в разных ситуациях, так:

- *Разреженный метод*, эффективнее использовать по времени выполнения операции (в данном случае умножение) при разреженности матрицы более, чем 32% (до 68%наполненности). Разреженную матрицу и вектор-столбец эффективнее хранить, когда нулевых элементов матрицы больше, чем ненулевых элементов. По результатам такая эффективность достигает при разреженности матрицы и вектора-столбца больше, чем 50%. (50% заполненности).
- *Простой метод*, эффективнее использовать при заполненности матрицы больше, чем 68% ненулевых элементов, по времени выполнения операции, так по хранению матриц и векторов-столбцов.

Контрольные вопросы

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

Разреженная матрица – матрица с преимущественно нулевыми элементами. Число ненулевых элементов в матрице порядка n может выражаться как n^{1+g} , где $g < 1$. Значения g лежат в интервале $0.2 \dots 0.5$.

Существуют различные методы хранения элементов матрицы в памяти.

Линейный связный список, т.е. последовательность ячеек, связанных в определенном порядке. Каждая ячейка списка содержит элемент списка и указатель на положение следующей ячейки.

Можно хранить матрицу, используя кольцевой связный список, двусторонние стеки и очереди.

Существует диагональная схема хранения симметричных матриц, а также связные схемы разреженного хранения.

Связная схема хранения матриц, предложенная Кнудом, предлагает хранить в массиве (например, в AN) в произвольном порядке сами элементы, индексы строк и столбцов соответствующих элементов (например, в массивах I и J), номер (из массива AN) следующего ненулевого элемента, расположенного в матрице по строке (NR) и по столбцу (NC), а также номера элементов, с которых начинается строка (указатели для входа в строку – JR) и номера элементов, с которых начинается столбец (указатели для входа в столбец – JS).

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

Для хранения обычной матрицы: $N * M * \text{sizeof}(\text{elem})$. Память под разреженную матрицу ($K * \text{sizeof}(\text{elem}) + (N + 1) * \text{sizeof}(\text{elem})$) выделяется в зависимости от схемы хранения. Память выделяется по мере заполнения ненулевыми элементами.

Пояснение:

- N – количество строк
- M – количество столбцов
- K – количество ненулевых элементов
- elem – тип данных

3. Каков принцип обработки разреженной матрицы?

Обработка разреженной матрицы предполагает работу только с ненулевыми элементами (таким образом, количество операций пропорционально количеству ненулевых элементов).

4. 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.

При достижении определенного процента заполнения ненулевыми элементами происходит значительное падение эффективности по времени.