



ФАКУЛЬТЕТ _____ Информатика и системы управления (ИУ)

КАФЕДРА _____ Программное обеспечение ЭВМ и информационные технологии (ИУ7)

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3 **«ОБРАБОТКА РАЗРЕЖЕННЫХ МАТРИЦ»**

Название: Обработка разреженных матриц

Студент Пискунов Панте

Группа ИУ7 – 36Б

Тип лабораторной работы: Учебная

Вариант № 2

Преподаватель Никульшина Татьяна Александровна

2022 г.

Условие задачи

Разработать программу сложения разреженных матриц. Предусмотреть возможность ввода данных, как с клавиатуры, так и использования заранее подготовленных данных. Матрицы хранятся и выводятся в форме трех объектов. Для небольших матриц можно дополнительно вывести матрицу в виде матрицы. Величина матриц - любая (допустим, 1000×1000). Сравнить эффективность (по памяти и по времени выполнения) стандартных алгоритмов обработки матриц с алгоритмами обработки разреженных матриц при различной степени разреженности матриц и различной размерности матриц.

Техническое задание

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 N_k которого находится номер компонент в **A** и **IA**, с которых начинается описание столбца N_k 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.

Входные данные:

1. **Целое число, представляющее собой номер команды:** целое число в диапазоне от 0 до 5.
2. **Командно-зависимые данные:**
 - Количество строк\столбцов матрицы
 - Элементы матрицы в формате “индекс строки, индекс столбца, значение элемента”

Выходные данные:

1. Исходные и результирующие матрицы в стандартном формате и разреженном столбцовом формате
2. Количественная характеристика сравнения времени сложения матриц разного формата
3. Количественная характеристика сравнения занимаемой памяти матриц разного вида

Описание структур данных

// Структура обычной матрицы

```
typedef struct matrix_t
```

```
{
```

```
    int *data; // матрица в виде одномерного массива
```

```
    int rows; // количество строк матрицы
```

```
    int columns; // количество столбцов матрицы
```

```
}matrix_t;
```

// Структура разреженной матрицы

```
typedef struct sparse_matrix_t
```

```
{
```

```
    int *a; // массив, содержащий значения ненулевых элементов
```

```
    int *ai; // массив, содержащий номера строк для элементов массива A
```

```
    int *aj; // массив, содержащий номер компонент
```

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

```
int *pa; // указатель на конец массива, содержащий значения ненулевых
элементов

int *pai; // указатель на конец массива, содержащий номера строк для
элементов массива A

int *paj; // указатель на конец массива, массив, содержащий номер компонент
// в A и IA, с которых начинается описание столбца Nk матрицы A

}sparse_matrix_t;
```

Меню программы:

1. Ввести две матрицы с клавиатуры
2. Сгенерировать две матрицы с заданным процентом разреженности
3. Сложить две матрицы
4. Вывести две матрицы
5. Сравнить время выполнения операций и объем памяти при сложении матриц в разных форматах
0. Выйти

Обращение к программе:

Запускается через терминал с помощью команды ./app.exe

Аварийные ситуации:

1. Некорректный ввод команды.
На входе: число большее 5, либо меньшее 0
На выходе: сообщение "Ошибка: вы ввели матрицу некорректно!"
2. Пустая матрица
На входе: Вызов любой из команд 3-5 при пустых матрицах.
На выходе: "Ошибка: вы не ввели матрицы"
3. Неверный ввод числа

На входе: неверно введенный индекс строки матрицы
На выходе: "Ошибка: вы ввели матрицу некорректно!"

4. Неверный ввод числа

На входе: неверно введенный индекс столбца матрицы
На выходе: "Ошибка: вы ввели матрицу некорректно!"

5. Неверный ввод числа

На входе: неверное значение количества строк\столбцов матрицы\ненулевых элементов матрицы
На выходе: сообщение "Ошибка: вы ввели матрицу некорректно"

6. Неверный ввод числа

На входе: неверное процентное значение ненулевых элементов матрицы
На выходе: "Ошибка: вы ввели неверные данные для генерации матрицы"

Описание алгоритма суммирования двух разреженных матриц

Есть две матрицы. Сначала смотрим в первую матрицу есть ли в первом столбце ненулевых чисел, если нет то переходим в вторую матрицу, если есть сравниваю есть ли в обе матрицы на одном и то же место ненулевых чисел, если есть то, их суммирую, если нет, тогда просто их переписываю. И так проверяю все столбцы.

Тесты

	Тест	Пользовательский ввод	Результат
1	Некорректный ввод команды	-1	Ошибка: вы ввели матрицу некорректно!
2	Некорректный ввод команды	f	Ошибка: вы ввели матрицу некорректно!
3	Ввод корректной команды (3-5) при пустых матрицах	любая команда 3-5	Ошибка: вы не ввели матрицы!
4	Неверное значение количества строк матрицы	-1	Ошибка: вы ввели матрицу некорректно

5	Неверное значение количества строк матрицы	iu	Ошибка: вы ввели матрицу некорректно
6	Неверное значение количества столбцов матрицы	0.228	Ошибка: вы ввели матрицу некорректно
7	Неверное значение количества столбцов матрицы	rk	Ошибка: вы ввели матрицу некорректно
8	Неверное процентное значение ненулевых элементов матрицы	100	Ошибка: вы ввели неверные данные для генерации матрицы
9	Неверное процентное значение ненулевых элементов матрицы	-1	Ошибка: вы ввели неверные данные для генерации матрицы

10	Неверное значение количества ненулевых элементов матрицы	5 (при размере матрицы 2 x 2)	Ошибка: вы ввели матрицу некорректно
11	Неверное значение количества ненулевых элементов матрицы	-1	Ошибка: вы ввели матрицу некорректно
12	Неверное значение номера строки	-1	Ошибка: вы ввели матрицу некорректно
13	Неверное значение номера строки	6 (при размере матрицы 5 x 5)	Ошибка: вы ввели матрицу некорректно
14	Неверное значение номера столбца	-6	Ошибка: вы ввели матрицу некорректно
15	Неверное значение номера столбца	6 (при размере матрицы 5 x 5)	Ошибка: вы ввели матрицу некорректно
16	Неверное значение элемента матрицы	warcraft	Ошибка: вы ввели матрицу некорректно
17	Неверное значение элемента матрицы	1.488	Ошибка: вы ввели матрицу некорректно
18	Ввод матриц вручную (команда 1)	корректно введенные матрицы	Заполненные матрицы
19	Заполнение матриц случайными числами (команда 2)	корректные размерности матриц и процент разреженности матриц	Матрицы заполненные случайными элементами
20	Сложение матриц (команда 3)	3	Результирующая матрица, полученная поэлементным сложением 1 и 2 матриц
22	Вывод исходных матриц в стандартном формате на экран (команда 4)	4	Исходные матрицы в стандартном формате
24	Вывести сравнение времени сложения и объема занимаемой памяти обычной и	5	Вывод сравнения времени сложения и объема занимаемой памяти матриц разных форматов

	разреженной матриц (команда 5)		
25	Выход из программы (команда 0)	0	Программа завершена!

Оценка эффективности

Измерения эффективности сложения матриц в разных форматах будут производиться в секундах. Матрицы будут заполняться случайными числами.

Время сложения:

10% разреженности

Размер	Стандартный формат(секунды)	Столбцовый формат(секунды)
10x10	0.000007	0.000035
100x100	0.000183	0.007568
500x500	0.004090	0.363914

30% разреженности

Размер	Стандартный формат(секунды)	Столбцовый формат(секунды)
10x10	0.000002	0.000007
100x100	0.000189	0.006008
500x500	0.004088	0.223072

50% разреженности

Размер	Стандартный формат(секунды)	Столбцовый формат(секунды)
--------	--------------------------------	-------------------------------

10x10	0.000005	0.000008
100x100	0.000183	0.004943
500x500	0.004102	0.116622

70% разреженности

Размер	Стандартный формат(секунды)	Столбцовый формат(секунды)
10x10	0.000007	0.000012
100x100	0.000189	0.001840
500x500	0.004005	0.075277

80% разреженности

Размер	Стандартный формат(секунды)	Столбцовый формат(секунды)
10x10	0.000005	0.00006
100x100	0.000188	0.000900
500x500	0.004083	0.020240

95% разреженности

Размер	Стандартный формат(секунды)	Столбцовый формат(секунды)
10x10	0.000008	0.00005
100x100	0.000197	0.00170
500x500	0.004085	0.04037

Объем занимаемой памяти:

10% разреженности

Размер	Стандартный формат	Столбцовый формат
10x10	1200	2292
100x100	120000	217212
500x500	3000000	5406012

30% разреженности

Размер	Стандартный формат	Столбцовый формат
10x10	1200	1812
100x100	120000	169212
500x500	3000000	4206012

50% разреженности

Размер	Стандартный формат	Столбцовый формат
10x10	1200	1332
100x100	120000	121212
500x500	3000000	3016012

55% разреженности

Размер	Стандартный формат	Столбцовый формат
10x10	1200	1190
100x100	120000	109212
500x500	3000000	2706012

Контрольные вопросы

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

Разреженная матрица — это матрица, содержащая большое количество нулей. Схемы хранения матрицы: связанная схема хранения, строчный формат, столбцовый формат, кольцевой связанный список, двунаправленные стеки и очереди.

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

Под обычную матрицу (N – количество строк, M – количество столбцов) выделяет $N \cdot M$ ячеек памяти. Для разреженной матрицы количество ячеек памяти зависит от способа. В случае разреженного формата требуется $3 \cdot K$ ячеек памяти, где K – количество ненулевых элементов.

3. Каков принцип обработки разреженной матрицы?

Алгоритмы обработки разреженных матриц предусматривают работу только с ненулевыми элементами.

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

Стандартные алгоритмы обработки матриц эффективнее применять при большом количестве ненулевых элементов (начиная примерно с 10%). Зависит от количества ненулевых элементов.

Вывод

Алгоритмы для работы с разреженной матрицей имеют смысл только, если количество ненулевых элементов в обычной матрице не много (в зависимости от алгоритма, в данном случае только при заполненности выше 95 процентов обычный алгоритм медленнее разреженного)

Стоит отметить, что когда процент ненулевых элементов матрицы 95% использование алгоритмов работы с разреженными матрицами будет выигрывать по времени, а при разреженности выше 55% будет выигрывать по памяти.