



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»

О Т Ч Е Т

Р у б е ж н ы й К о н т р о л ь

Название *Разреженная матрица*

Дисциплина: *Анализ алгоритмов*

Студент

ИУ7И-56Б

(Группа)

Нгуен Ф. С.

(Подпись, дата)

(И.О. Фамилия)

Преподаватель

Волокова Л. Л.

(Подпись, дата)

(И.О. Фамилия)

Москва, 2020

Оглавление

ВВЕДЕНИЕ.....	3
1. АНАЛИТИЧЕСКАЯ ЧАСТЬ:	4
1.1. Понятие и хранение разреженной матрицы в кольцевой схеме Рейнбольдта-Местеньи	4
1.2. Описание алгоритма склейки двух разреженных матриц	4
2. КОНСТРУКТОРСКАЯ ЧАСТЬ	6
2.1. Схемы алгоритмов	6
3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ	9
3.1. Средства реализации.....	9
3.2. Тесты.....	9
4. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ	11
4.1. Сравнение работы алгоритмов.....	11
ЗАКЛЮЧЕНИЕ	14

Введение

Цель лабораторной работы - изучение алгоритма склейки двух разреженных матриц в кольцевой схеме Рейнбольдта-Местеньи

Для того чтобы добиться этой цели, были поставлены следующие задачи:

- изучение понятия разреженной матрицы
- хранение матриц в кольцевой схеме Рейнбольдта-Местеньи
- изучение алгоритма склейки двух разреженных матриц
- оптимизирование работы алгоритма
- сравнение эффективности алгоритмов по времени

1. Аналитическая часть:

1.1. Понятие и хранение разреженной матрицы в кольцевой схеме Рейнбольдта-Местеньи

Разрежённая матрица — это матрица с преимущественно нулевыми элементами.

Ненулевые элементы хранятся в компактной форме и в произвольном порядке в одномерном массиве, скажем AN.

Пусть NR <<следующий ненулевой элемент той же строки>> - Массив столбцовых указателей

Пусть JR JC массивы содержащие указатели входа для строк и столбцов, расположенные в соответствии с порядком строк и столбцов матрицы

$$A = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \left[\begin{array}{cccc} & 6. & & \\ 9. & 4. & & 7. \\ 5. & & & \\ & 2. & & 8. \end{array} \right] \end{matrix}$$

Рисунок 1. Разреженная матрица

		1	2	3	4	5	6	7
AN	=	6.	9.	4.	7.	5.	2.	8.
NR	=	1	3	4	2	5	7	6
NC	=	3	5	6	7	2	1	4
JR	=	1	2	5	6			
JC	=	2	1	0	4			

Рисунок 2. Хранение разреженной матрицы в кольцевой схеме Рейнбольдта-Местеньи

1.2. Описание алгоритма склейки двух разреженных матриц

- AN: добавляет элементы массива 2-й матрицы AN2 к массиву AN1 2-й матрицы

- NC: увеличивает элементы в массиве NC2 второй матрицы на число, которое является количеством частей в первой матрице, затем добавьте массив NC1 массива 1
- JC: увеличивает количество элементов в массиве JC2 второй матрицы на число, которое является количеством частей в первой матрице, затем добавьте массив JC1 1-й матрицы
- Далее вычисляем 2 массива NR и JR:
 - Для каждой строки i :
 - Если 2-я строка пуста (т.е. $JR2[i] = -1$), то $JR[i]$ и $NR[i]$ не нуждаются в изменении.
 - В противном случае, если первая строка пуста, исправить $JC[i] =$ первый элемент второй строки
 - И наоборот, добавить вторую строку к первой строке:
 - Получим первый элемент первой строки, перейдем к концу строки,
 - Последний элемент первой строки указывает на первый элемент второй строки
 - Продолжим в конце второй строки
 - Последний элемент второй строки указывает на первый элемент первой строки

1.3. Оптимизированный алгоритм

Для оптимизации алгоритма:

Объединить циклы с одинаковым количеством итераций (например NC и AC)

Вместо этого для каждой строки с использованием массива JC, указывающего на первый элемент каждой строки, мы используем массив lastrow, который указывает на последний элемент каждой строки, поэтому в процессе объединения двух строк нам не нужно повторять от первого до последнего мы просто используем $JC[i]$, чтобы получить последний элемент строки i , а $NC[JC[i]]$ дает нам первый элемент строки i . Это значительно сокращает время и объем вычислений.

2. Конструкторская часть

2.1.Схемы алгоритмов

На рис. 3 приведена схема алгоритма склейки двух разреженных матриц

На рис. 4 приведена схема оптимизированного алгоритма

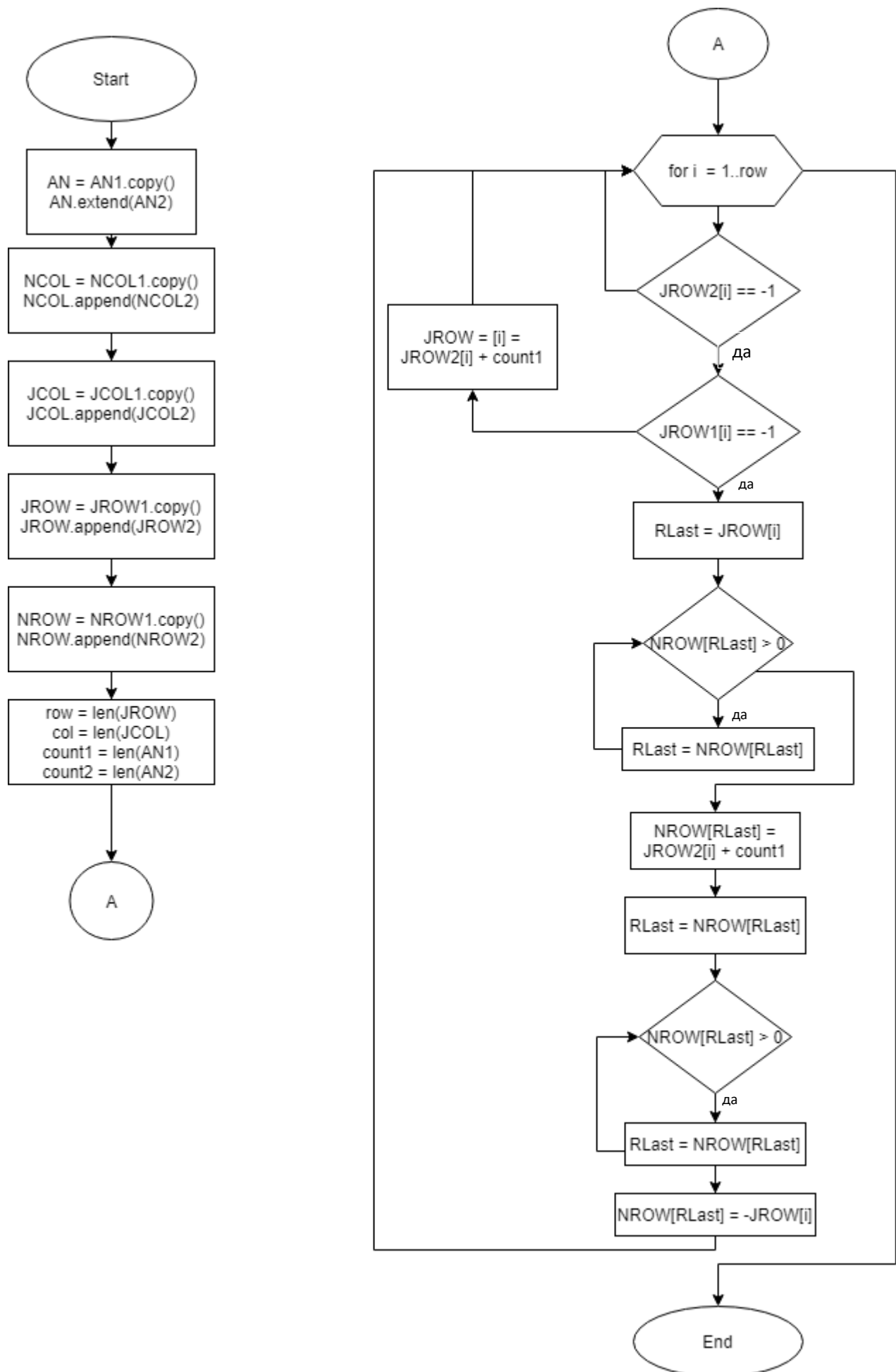


Рисунок 3. Алгоритм склейки двух разреженных матриц

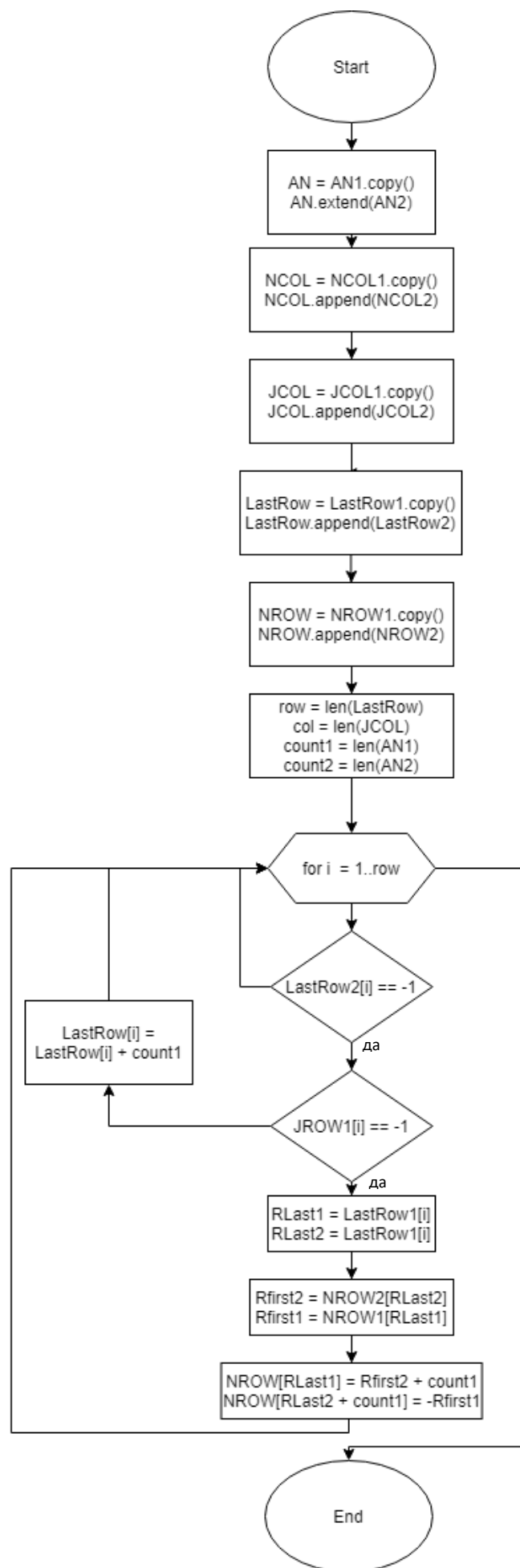


Рисунок 4. Оптимизированный алгоритм

3. Технологическая часть

3.1. Средства реализации

Для реализации программы был использован язык Python. Для замера процессорного времени была использована функция `time()` из библиотеки `time`.

3.2. Тесты

Результаты тестирования показаны на рисунке 5 и 6

```
Matrix A:
Matrix
| 4 | - | 6 | - | -
| - | - | - | - | -
| - | - | - | - | 10
| 8 | - | - | - | 5
| - | - | - | - | 5

Matrix B
Matrix
| - | - | - | - | 9
| 7 | - | - | - | -
| - | 2 | - | 2 | -
| 4 | - | - | - | -
| 5 | - | - | - | -

Waiting Result:
Matrix
| 4 | - | 6 | - | - | - | - | - | - | 9
| - | - | - | - | - | 7 | - | - | - | -
| - | - | - | - | 10 | - | 2 | - | 2 | -
| 8 | - | - | - | 5 | 4 | - | - | - | -
| - | - | - | - | 5 | 5 | - | - | - | -

Connect Resulty
Matrix
| 4 | - | 6 | - | - | - | - | - | - | 9
| - | - | - | - | - | 7 | - | - | - | -
| - | - | - | - | 10 | - | 2 | - | 2 | -
| 8 | - | - | - | 5 | 4 | - | - | - | -
| - | - | - | - | 5 | 5 | - | - | - | -

Test Matrix Connect True
```

Figure 5. Результат программы

```

Matrix A:
Matrix
| 3| -| -| -| -
| 9| -| -| -| -
| 1| -| -| 2| -
| -| -| -| 2| -
| 4| -| -| -| -

Matrix B
Matrix
| 7| -| -| -| -
| -| -| 8| -| 4
| -| -| -| -| -
| 10| -| -| -| -
| -| 9| -| -| 1

Waiting Result:
Matrix
| 3| -| -| -| -| 7| -| -| -| -
| 9| -| -| -| -| -| -| 8| -| 4
| 1| -| -| 2| -| -| -| -| -| -
| -| -| -| 2| -| 10| -| -| -| -
| 4| -| -| -| -| -| -| 9| -| -| 1

Connect Resuly
Matrix
| 3| -| -| -| -| 7| -| -| -| -
| 9| -| -| -| -| -| -| 8| -| 4
| 1| -| -| 2| -| -| -| -| -| -
| -| -| -| 2| -| 10| -| -| -| -
| 4| -| -| -| -| -| -| 9| -| -| 1

Test Matrix Connect True

```

Figure 6. Результат программы

4. Экспериментальная часть

4.1. Сравнение работы алгоритмов

Для сравнения времени работы алгоритмов были использованы квадратные матрицы размером от 100 до 1000 с шагом 100. Результаты измерений показаны на рисунке 7-10.

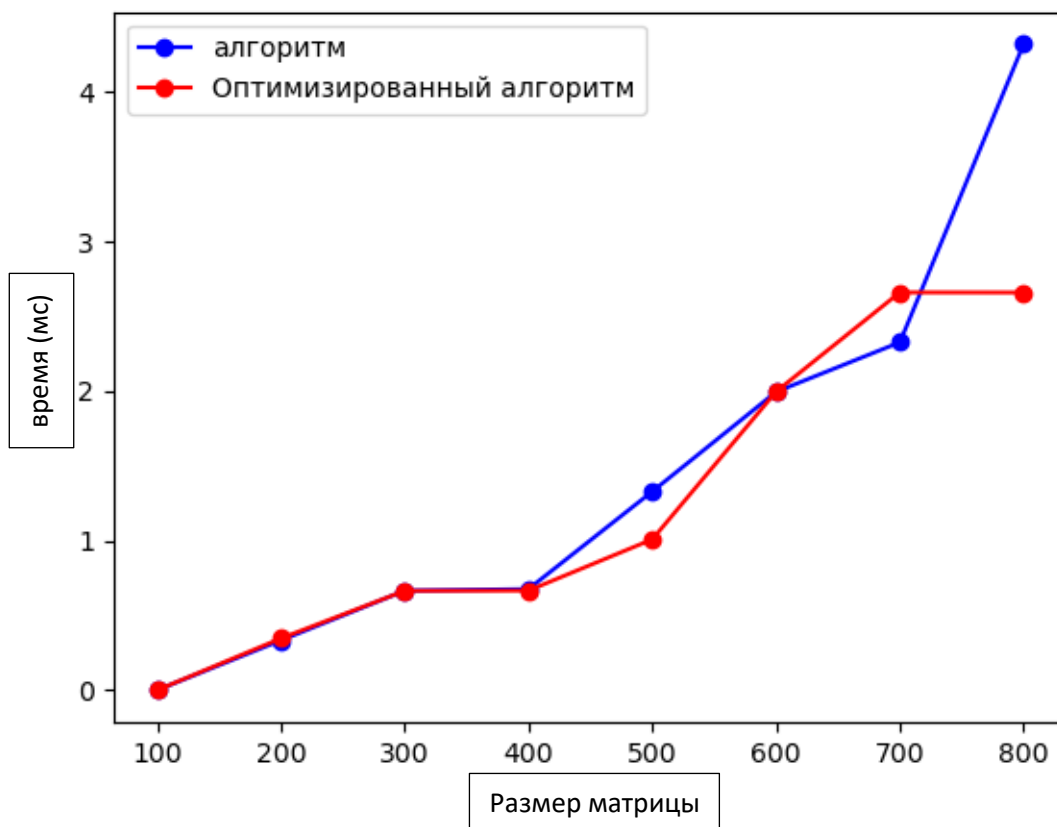


Рисунок 7. Матрица с заполнением 0,5%

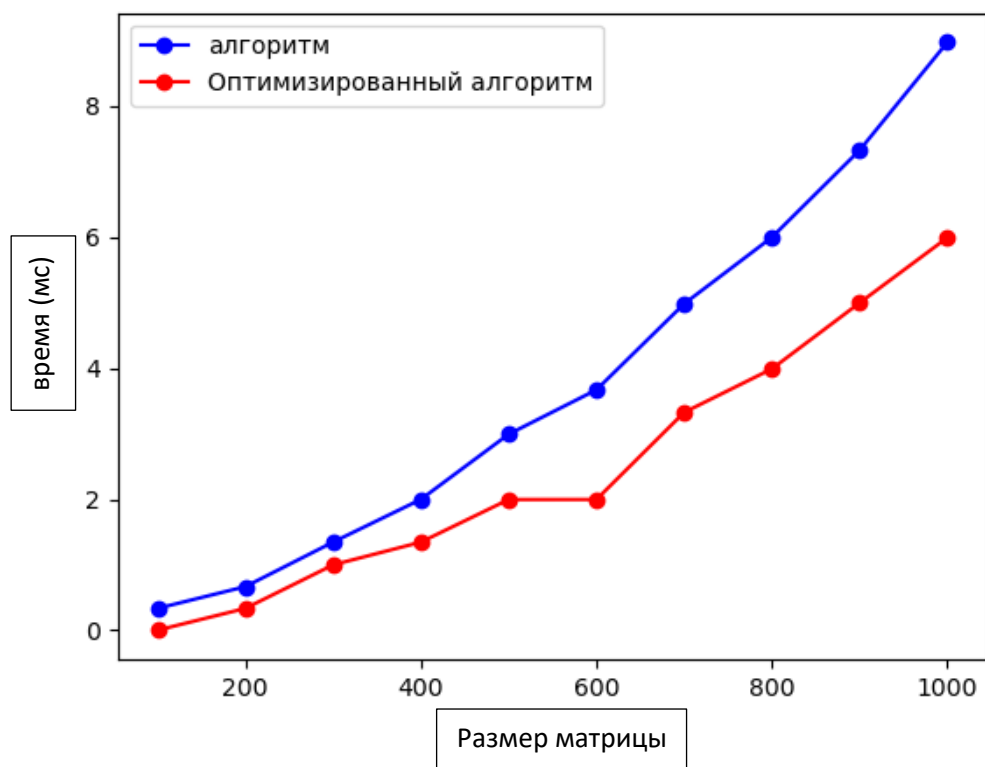


Рисунок 8. Матрица с заполнением 1%

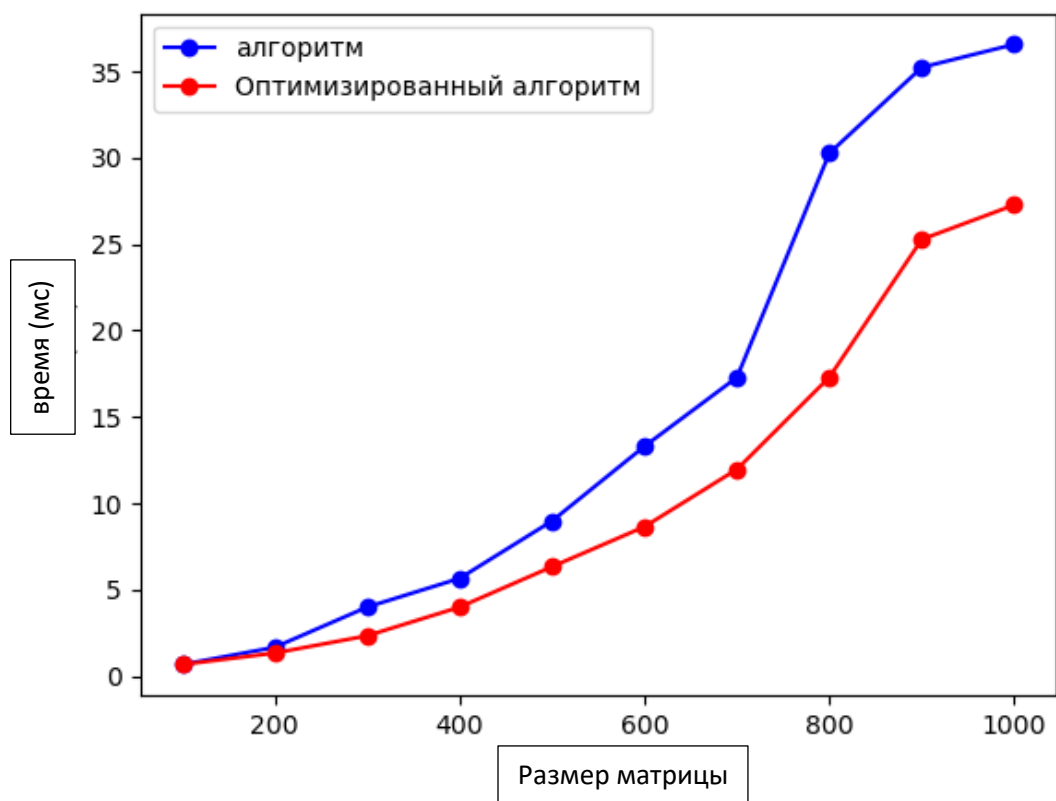


Рисунок 9. Матрица с заполнением 5%

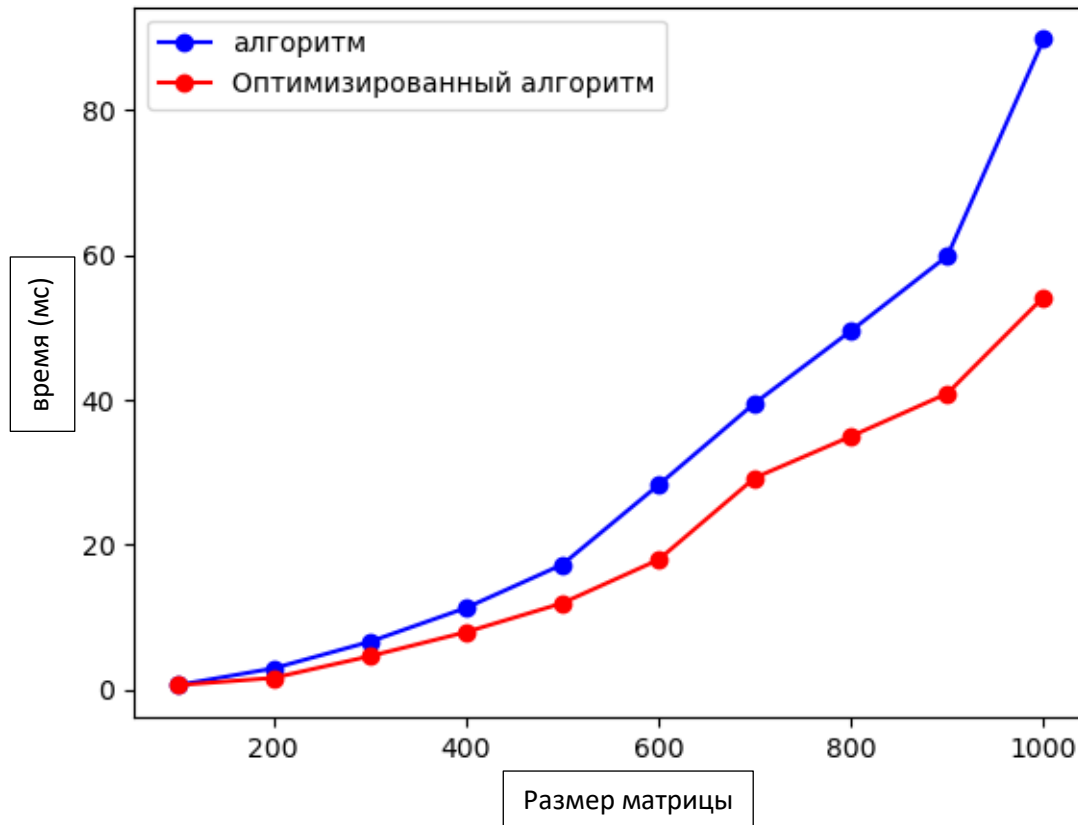


Рисунок 10. Матрица с заполнением 10%

❖ Вывод

По результатам исследования:

Когда матрица разреженная, время работы двух алгоритмов почти одинаково.

В случае высокого заполнения матрицы, тем выше разница между двумя алгоритмами.

Это выполняется путем прокрутки до конца каждого списка. Чем длиннее список (высокое заполнение), тем больше времени.

Заключение

В рамках лабораторной работы было выполнены следующие задачи:

- изучение понятия разреженной матрицы
- хранение матриц в кольцевой схеме Рейнбольдта-Местеньи
- изучение алгоритма склейки двух разреженных матриц
- оптимизирование работы алгоритма
- сравнение эффективности алгоритмов по времени