|  |  |
| --- | --- |
| Программа для операций над матрицами | |
| Программа и методика испытаний | |
| Студент | Быков А.А. |
| Преподаватель | Матюшечкин Д.С. |
| Сдано |  |

1. Общие сведения

Наименование программы – «Программа для операций над матрицами».

Для функционирования программы необходима операционная система

Windows 7 или выше.

Программа написана на языке C++.

2. Цель испытаний

Целью испытаний является проверка соответствия программы требованиям к функциональным характеристикам.

3. Требования к программе

Требования к программе изложены во внешней спецификации в 3 пункте.

4. Требования к программной документации

В бумажной форме должны быть предоставлены: техническое задание, технический проект в виде описания программы, программа и методика испытаний и руководство программиста. В электронном формате должны быть предоставлены: копии всех документов бумажной формы, рабочая документация и текст программы. Вся документация должна быть составлена в соответствии с ГОСТ 19.

5. Средства и порядок испытаний

Для запуска тестов необходимо подключение библиотеки CppUnitTest к проекту программы.

6. Методы испытаний

Аспекты тестирования приведены в приложении 1

1.Перечень проверок для функции выделяющий матрицу из файла в массив getMatrixFromFile представлен в приложении 2, для функции выделяющий числа из строки getArrFromRow представлен в приложении 3, для функции выделяющий число из массива символов getNumberFromRow в приложении 4, для функции определяющий количество столбцов в матрцие getNumberOfColumns в приложении 5, для функции определяющий количество строк в матрцие getNumberOfRows в приложении 6, для функции определяющий общий размер для двух матриц getSizeLimitOfMatrix в приложении 7, для функции заполняющий матрицу 0 setZeroMatrix в приложении 8, для функции проверяющий размер матриц на выход за ограничения isSizeCorrect в приложении 9, для функции, которая находит наибольше кол-во столбцов в матрице getMaxRowSize в приложении 10, для функции, которая складывает две матрицы matrixAdd в приложении 11, для функции, которая вычитает из первой матрицы вторую matrixSub в приложении 12, для функции, которая умножает первую матрицу на вторую matrixMul в приложении 13.

# Приложение 1

Аспекты тестирования (Таблица №1)

|  |  |  |
| --- | --- | --- |
| Тестируемая функция | АСПЕКТЫ ТЕСТИРОВАНИЯ | ВАРИАНТЫ ЗНАЧЕНИЙ |
| getMatrixFromFile | Кол-во строк матрицы | Минимальное, обычное, максимальное |
| Кол-во столбцов матрицы | Минимальное, обычное, максимальное |
| Кол-во строк и столбцов матрицы | Минимальное, обычное, максимальное |
| getArrFromRow | Длина строки | Минимальное, обычное, максимальное |
| Некорректные данные | До числа, между чисел, после числа, без числа |
| getNumberFromRow | Отрицательное число | Да, нет |
| Некорректные данные | До числа, между чисел, после числа, без числа |
| getNumberOfColumns | Размер строки | Разные размеры строк, одна большая строка, нулевая строка |
| Пустой файл | Да, нет |
| Длинные цифры | Да, нет |
| Некорректные разделители | Да, нет |
| getNumberOfRows | Кол-во строк матрицы | Минимальное, обычное, максимальное |
| Пустой файл | Да, нет |
| getSizeLimitOfMatrix | Строки | Разные, одинаковые |
|  | Столбцы | Разные, одинаковые |
| setZeroMatrix | Размер матрицы | Минимальное, обычное, максимальное |
| isSizeCorrect | Кол-во строк матрицы | Минимальное, обычное, максимальное, выход за границы |
| Кол-во столбцов матрицы | Минимальное, обычное, максимальное, выход за границы |
| getMaxRowSize | Максимальное значение | В начале, в середине, в конце |
| matrixAdd | Кол-во строк матрицы | Минимальное, обычное, максимальное |
| Кол-во столбцов матрицы | Минимальное, обычное, максимальное |
| matrixSub | Кол-во строк матрицы | Минимальное, обычное, максимальное |
| Кол-во столбцов матрицы | Минимальное, обычное, максимальное |
| matrixMul | Кол-во строк матрицы | Минимальное, обычное, максимальное |
| Кол-во столбцов матрицы | Минимальное, обычное, максимальное |

# Приложение 2

Задача функции

Записать матрицу из файла в массив.

bool getMatrixFromFile(int arr[][MAX\_SIZE], ifstream& file, string path)

Входные данные:

file - файл из которого будем брать матрицу

path - путь до файла

Выходные данные:

arr – массив с матрицей из файла

Таблица №2. Тесты для функции getMatrixFromFile

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | 21,2,5  11,23,4 | {{21,2,5}, {11,23,4}}; |
| 2 | Максимальное кол-во строк | 21,2,5  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4 | {{21,2,5}, {11,23,4},{11,23,4}, {11,23,4},{11,23,4},{11,23,4},  {11,23,4}, {11,23,4},{11,23,4}, {11,23,4},{11,23,4}, {11,23,4}, {11,23,4}, {11,23,4},{11,23,4}, {11,23,4}, {11,23,4},{11,23,4}, {11,23,4}, {11,23,4}}; |
| 3 | Минимальное кол-во строк | 21,2,5 | {{21,2,5}}; |
| 4 | Максимальное кол-во столбцов | 21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5 | {{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5}}; |
| 5 | Минимальное кол-во столбцов | 21  12 | {{21},{12}}; |
| 6 | Максимальное кол-во столбцов и строк | 21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5  21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5 | {{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},{21,2,5,5,3,4,5,6,7,8,9,9,0,4,5,67,1,23,4,5},}; |
| 7 | Минимальное кол-во столбцов и строк | 21 | {{21}}; |

Приложение 3

Задача функции

Выделить числа из строки.

bool getArrFromRow(int arr[][MAX\_SIZE], int row, string line)

Входные данные:

row - кол-во строк в массиве

line - строка, в которой будем искать числа

Выходные данные:

Массив с числами из строки

return – значение true, если в матрице только целые числа; false, если в матрице не только целые числа

Таблица №3. Тесты для функции getArrFromRow

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | "12,5,67,787" | {12,5,67,787} |
| 2 | Много чисел | "12,5,67,787,3214,21,345,56,567,77,213,23,45,6,7,8,90,12,2" | {12,5,67,787,3214,21,345,56,567,77,213,23,45,6,7,8,90,12,2} |
| 3 | Одно число | "12" | {12} |
| 4 | Некорректные данные после цифры | "12,5fdsf,67" | false |
| 5 | Некорректные данные перед цифрой | "12,fdsf5,67" | false |
| 6 | Некорректные данные между цифрами | "12,32ff5,67" | false |
| 7 | Некорректные данные без цифр | "12,fffa,67" | false |

Приложение 4

Задача функции

Выделить числа из массива символов.

bool getNumberFromRow(char\* value, int arr[][MAX\_SIZE], int &row, int& col)

Входные данные:

value – массив символов с цифрами

row – строка в который ищем цифру

col – столбец в который ищем цифру

Выходные данные:

Массив цифр

return – значение true, если в матрице только целые числа; false, если в матрице не только целые числа

Таблица №4. Тесты для функции getNumberFromRow

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | {'1','2'} | {12} |
| 2 | Отрицательное число | {'-','2'} | {-2} |
| 3 | Некорректные данные перед цифрой | { 'a','2' } | false |
| 4 | Некорректные данные после цифрой | { '2','a' } | false |
| 5 | Некорректные данные между цифрами | { '2','a', '3' } | false |

Приложение 5

Задача функции:

Разбить строку на подгруппы и поместить их в вектор строк

int getNumberOfColumns(ifstream& file, string path)

Входные данные:

file - файл из которого будем брать матрицу

path - путь до файла

Выходные данные:

return - кол-во столбцов в матрице

Таблица №5. Тесты для функции getNumberOfColumns

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | 21,2,5  11,23,4 | 3 |
| 2 | Разные длины строк | 21,2  11,23,4,5  11,23,4 | 4 |
| 3 | Одна длинная строка | 11,23,4,5,23,4,5,23,4,5,23,4,5,23,4,5,23,4,5 | 19 |
| 4 | Одна нулевая строка | 21,2  11,23,4 | 3 |
| 5 | Пустой файл |  | 0 |
| 6 | Некорректные разделители | 21 – 2 - 5 | 1 |

Приложение 6

Задача функции:

Написать группы строк в обратном порядке

int getNumberOfRows(ifstream& file, string path)

Входные данные:

file - файл из которого будем брать матрицу

path - путь до файла

Выходные данные:

return - кол-во строк в матрице

Таблица №6. Тесты для функции getNumberOfRows

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | 21,2,5  11,23,4 | 2 |
| 2 | Максимальное кол-во строк | 21,2,5  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4  11,23,4 | 20 |
| 3 | Минимальное кол-во строк | 21,2,5 | 1 |
| 4 | Пустой файл |  | 0 |

Приложение 7

Задача функции:

Определить общий размер для двух матриц

void getSizeLimitOfMatrix(int w1, int h1, int w2, int h2, int result[]);

Входные данные:

w1 – кол-во столбцов первой матрицы

h1 – кол-во строк первой матрицы

w2 – кол-во столбцов первой матрицы

h2 – кол-во строк первой матрицы

Выходные данные:

return – общий размер для двух матриц

Таблица №7. Тесты для функции getSizeLimitOfMatrix

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | 1, 4, 3, 2 | 3,4 |
| 2 | Одинаковое кол-во столбцов | 10, 4, 10, 2 | 10,4 |
| 3 | Одинаковое кол-во строк | 1, 4, 3, 4 | 3, 4 |

Приложение 8

Задача функции:

Определить общий размер для двух матриц

void setZeroMatrix(Matrix\* matr, int size[])

Входные данные:

matr – матрица

size – размер матрицы

Выходные данные:

Матрица заполненная нулями

Таблица №8. Тесты для функции setZeroMatrix

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | 2,1 | {{0},{0}} |
| 2 | Максимальный размер | 20, 20 | {{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}} |
| 3 | Минимальный размер | 1,1 | {{0}} |

Приложение 9

Задача функции:

Определить общий размер для двух матриц

bool isSizeCorrect(int size[])

Входные данные:

return – значение true, если в размер не выходит за границы; false, если размер выходит за границы

Таблица №9 Тесты для функции isSizeCorrect

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | { 2,10 } | True |
| 2 | Максимальный размер столбцов | { 2,20 } | True |
| 3 | Минимальный размер столбцов | { 2,1 } | True |
| 4 | Максимальный размер строк | { 20,5 } | True |
| 5 | Минимальный размер строк | { 1,5 } | True |
| 6 | Выход за границы по строкам | { 1,100 } | False |
| 7 | Выход за границы по столбцам | { 100,1 } | False |

Приложение 10

Задача функции:

Найти наибольше кол-во столбцов в матрице

int getMaxRowSize(int arr[], int size);

arr – длины всех строк в матрице

size – кол-во строк в матрице

Входные данные:

return – наибольшее кол-во столбцов

Таблица №10 Тесты для функции getMaxRowSize

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | { 1,3,6,10,7 } | 10 |
| 2 | Максимальная длина в начале | { 19,3,6,10,7 } | 19 |
| 3 | Максимальная длина в середине | { 19,3,6,10,20,11,3,2,10 } | 20 |
| 4 | Максимальная длина в конце | { 19,3,6,10,1,11,3,2,20 } | 20 |

Приложение 11

Задача функции:

Найти сумму двух матриц

void matrixAdd(int matr1[][MAX\_SIZE], int matr2[][MAX\_SIZE], int result[][MAX\_SIZE], int size[])

matr1– первая слагаемая матрица

matr2– вторая слагаемая матрица

size – общий размер для матриц

Входные данные:

Сумма двух матриц

Таблица №11 Тесты для функции matrixAdd

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | {{1,2,3},{1,5,12}};  {{1,2,3,10},{1,5,12,16}}; | {{2,4,6,10},{2,10,24,16}} |
| 2 | Максимальное кол-во столбцов | {{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21}};  {{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}}; | {{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44}} |
| 3 | Минимальное кол-во столбцов | {{1},{2},{3}};  {{10},{4}}; | {{11},{6},{3}} |
| 4 | Максимальное кол-во строк | {{1},{25},{1},{16},{54},{25},{38},{1},{25},{1},{16},{54},{25},{38},{15},{17},{213},{2135},{23},{12}};  {{10},{5},{12},{1},{5},{225},{8},{15},{95},{145},{136},{54},{25},{38},{15},{17},{13},{5},{23},{12}}; | {{11},{30},{13},{17},{59},{250},{46},{16},{120},{146},{152},{108},{50},{76},{30},{34},{226},{2140},{46},{24}} |
| 5 | Минимальное кол-во строк | {{1,2,3}};  {{10,8,9}}; | {{11,10,12}} |
| 6 | Максимальное кол-во столбцов и строк | {{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21}};  {{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}}; | {{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44},{2,4,6,33,10,66,14,144,14,337,56,62,88,324,88,18,41,134,468,44}} |
| 7 | Минимальное кол-во столбцов и строк | {{1}};  {{10}}; | {{11}} |

Приложение 12

Задача функции:

Найти разницу двух матриц

void matrixSub(int matr1[][MAX\_SIZE], int matr2[][MAX\_SIZE], int result[][MAX\_SIZE], int size[])

matr1– уменьшаемая матрица

matr2– вычитаемая матрица

size – общий размер для матриц

Входные данные:

Разница двух матриц

Таблица №12 Тесты для функции matrixSub

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | {{1,2,3},{1,5,12}};  {{1,2,3,10},{1,5,12,16}}; | {{0,0,0,-10},{0,0,0,-16}} |
| 2 | Максимальное кол-во столбцов | {{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21}};  {{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}}; | {{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2}} |
| 3 | Минимальное кол-во столбцов | {{1},{2},{3}};  {{10},{4}}; | {{-9},{-2},{3}} |
| 4 | Максимальное кол-во строк | {{1},{25},{1},{16},{54},{25},{38},{1},{25},{1},{16},{54},{25},{38},{15},{17},{213},{2135},{23},{12}};  {{10},{5},{12},{1},{5},{225},{8},{15},{95},{145},{136},{54},{25},{38},{15},{17},{13},{5},{23},{12}}; | {{-9},{20},{-11},{15},{49},{-200},{30},{-14},{-70},{-144},{-120},{0},{0},{0},{0},{0},{200},{2130},{0},{0}} |
| 5 | Минимальное кол-во строк | {{1,2,3}};  {{10,8,9}}; | {{-9,-6,-6}} |
| 6 | Максимальное кол-во столбцов и строк | {{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21}};  {{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23},{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}}; | {{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2},{0,0,0,13,0,-54,0,12,-10,313,-10,-50,86,-144,-20,6,27,-44,444,-2}} |
| 7 | Минимальное кол-во столбцов и строк | {{1}};  {{10}}; | {{-9}} |

Приложение 13

Задача функции:

Найти произведение двух матриц

void matrixMul(Matrix\* mat1, Matrix\* mat2, int result[][MAX\_SIZE]);

matr1– множимое

matr2– множитель

Входные данные:

Произведение двух матриц

Таблица №13 Тесты для функции matrixMul

|  |  |  |  |
| --- | --- | --- | --- |
| № | НАЗВАНИЕ ТЕСТОВ | ТЕСТОВЫЕ ДАННЫЕ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
| 1 | Типовой тест | {{{1,2,3},{1,5,12}},3,2};  {{{1,2,3,10},{1,5,12,16},{1,2,3,10}},4,3}; | { {6,18,36,72}, {18,51,99,210} } |
| 2 | Максимальное кол-во столбцов | {{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21}, 20,1};  {{{10},{5},{12},{1},{5},{225},{8},{15},{95},{145},{136},{54},{25},{38},{15},{17},{13},{5},{23},{12}}, 1,20}; | { {71163} } |
| 3 | Минимальное кол-во столбцов и строк | { {1}, 1,1 };  { {{10}}, 1,1 }; | { {10} } |
| 4 | Максимальное кол-во строк | {{{10},{5},{12},{1},{5},{225},{8},{15},{95},{145},{136},{54},{25},{38},{15},{17},{13},{5},{23},{12}}, 1,20};  {{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21}, 20,1}; | { {71163} } |
| 6 | Максимальное кол-во столбцов и строк | { {{1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, {1,2,3,10,5,60,7,66,12,12,33,56,1,234,54,6,7,89,12,23}, 20,20 };  {{{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,1},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,45621},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21},{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21,{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21,{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21,{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21,{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21,{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21,{1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, {1,2,3,23,5,6,7,78,2,325,23,6,87,90,34,12,34,45,456,21, }, 20, 20}; | {{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980},{1260,2520,3780,12600,6300,75600,8820,83160,15120,15120,41580,70560,1260,294840,68040,7560,8820,112140,15120,28980}} |