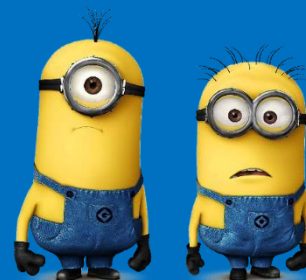




Task 49 (revision)

Task 49 (revision)

**C/C++ Basic Syntax.  
Pointers. Multidimensional  
Dynamic Arrays**



**LEARN. GROW. SUCCEED.**

© 2025. STEP Computer Academy - a leader in the field of professional computer education  
by Viktor Ivanchenko / [ivanvikvik@gmail.com](mailto:ivanvikvik@gmail.com) / Minsk

# Т

## Базовый синтаксис языка C/C++.

## Основы использования указателей.

## Динамические многомерные массивы.

## Адресная арифметика

### Цель работы

Проверить и практически закрепить работу с динамической памятью с использованием указателей и адресной арифметики в языке C/C++ на примере работы с динамическими многомерными массивами.

### Требования

- 1) Для каждого задания в начале рекомендуется разработать блок-схему алгоритма решения.
- 2) Проект обязательно должен быть сразу реализован и сохранён под управление системой контроля версий (VCS) **git** и в последующем залит в централизованный репозиторий на облачном хостинг-сервисе **GitHub**.  
се программы должны быть разбиты на отдельные функции. При выполнении задания необходимо по максимуму пытаться разрабатывать универсальный, масштабируемый, легко поддерживаемый и читаемый код.
- 4) В соответствующих компонентах бизнес-логики необходимо предусмотреть **«защиту от дурака»** (*fool-proof*), т.е. прежде чем выполнять действия с данными нужно проверить, являются ли данные адекватными (непротиворечивыми).
- 5) Одномерные и многомерные структуры данных рекомендуется реализовывать на базе **динамических C/C++ массивов**.
- 6) Также рекомендуется придерживаться **Single Responsibility Principle, SRP** (принципа единственной ответственности) – постарайтесь вынести основную бизнес-логику задания в отдельную функцию или функции (т.е. архитектура приложения должна минимум состоять из нескольких функций).

программа должна обязательно быть снабжена комментариями на английском языке, в которых необходимо указать краткое предназначение программы, её версию, ФИО разработчика, номер группы и дату разработки.

сходный текст основного кода и демонстрационной программы рекомендуется также снабжать поясняющими краткими комментариями.

если логически не подразумевается или в задании иного не указано, то входными и выходными данными являются целые числа.

программа должна быть снабжена дружелюбным и интуитивно понятным интерфейсом для взаимодействия с пользователем.

предусмотреть вывод на консоль удобочитаемого результата для пользователя. Рекомендуется для программного интерфейса использовать английский язык.

- 12) При проверке работоспособности приложения необходимо проверить все тестовые случаи.
- 13) Для предоставляемого решения задания также необходимо подсчитать его алгоритмическую сложность (**Big O Notation**) для всех типов измерений: худший, средний и лучший случаи.  
при разработке программ придерживайтесь соглашений по написанию кода на языке C/C++ (C++ *Code-Convention*).

## Основное задание

**среднее арифметическое ненулевых элементов** [The arithmetic mean of non-zero elements] на M. Необходимо разработать функцию (или программу), которая вычисляет среднее арифметическое ненулевых элементов матрицы. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным) и тестовые данные. Первоначальные данные матрицы могут быть автоматически сгенерированы с помощью встроенного генератора псевдослучайных чисел.

Тест 01

Введите размерность матрицы (N и M): **2 3**

Введите элементы матрицы:

**1 2 3**

**7 8 9**

Среднее арифметическое ненулевых элементов матрицы: 5.0

Test 02

Input the matrix dimension (N and M): **3 2**

Enter the matrix elements:

**1 0**

**1 1**

**0 1**

The Arithmetic mean of non-zero elements of the matrix: 1.0

Test 03

Input the matrix dimension (N and M): **3 4**

Enter the matrix elements:

**1 0 2 2**

**1 3 0 6**

**2 1 8 9**

The Arithmetic mean of non-zero elements of the matrix: 3.5

Test04

...

**умма элементов** [*The sum of elements*]. Дана математическая квадратная матрица размером N. Необходимо разработать функцию (или программу), которая вычисляет сумму элементов матрицы, расположенных на главной и побочной диагоналях. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным) и тестовые данные. Первоначальные данные матрицы могут быть автоматически сгенерированы с помощью встроенного генератора псевдослучайных чисел.

*Тест 01*

*Введите размерность матрицы:* **3**

*Введите элементы матрицы:*

**2 3 4**

**3 4 5**

**4 5 6**

*Сумма элементов, стоящих на главной и побочной диагоналях:* **24**

*Test 02*

*Input the matrix dimension:* **2**

*Enter the matrix elements:*

**-1 2**

**-2 3**

*The Sum of the elements on the main and secondary diagonals:* **2**

*Test03*

...

**оличество строк с большим количество положительных элементов**  
ская прямоугольная матрица размером N на M. Необходимо разработать функцию (или программу), которая находит количество строк, в которых положительных элементов больше чем отрицательных. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным) и тестовые данные. Первоначальные данные матрицы могут быть автоматически сгенерированы с помощью встроенного генератора псевдослучайных чисел.

*Тест 01*

*Введите размерность матрицы (N и M):* **3 4**

*Введите элементы матрицы:*

*Количество строк с большим количество положительных элементов:* **2**

*Test 02*

*Input the matrix dimension (N and M):* **3 2**

*Enter the matrix elements:*

**1 0**

**1 1**

**0 1**

*The Number of rows with a large number of positive elements:* **1**

*Test 03*

*Input the matrix dimension (N and M):* **2 2**

*Enter the matrix elements:*

**1**

**2 -1**

*The Number of rows with a large number of positive elements:* **0**

*Test04*

...

**сумма элементов столбцов с экстремальными элементами** [The Sum of elements of columns with extreme elements]. Дана математическая прямоугольная матрица размером N на M. Необходимо разработать функцию (или программу), которая высчитывает сумму элементов в тех столбцах заданной матрицы, которые содержат хотя бы один экстремальный элемент. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным) и тестовые данные. Первоначальные данные матрицы могут быть автоматически сгенерированы с помощью встроенного генератора псевдослучайных чисел.

Тест 01

Введите размерность матрицы (N и M): **2 2**

Введите элементы матрицы:

Сумма элементов столбцов с экстремальными элементами: 6

Test 02

Input the matrix dimension (N and M): **2 3**

Enter the matrix elements:

**1 2 2**

**3 1 2**

The Sum of elements of columns with extreme elements: 7

Test 03

Input the matrix dimension (N and M): **3 4**

Enter the matrix elements:

**-1 -2 2 2**

**2 -1 1 7**

**1 2 3 1**

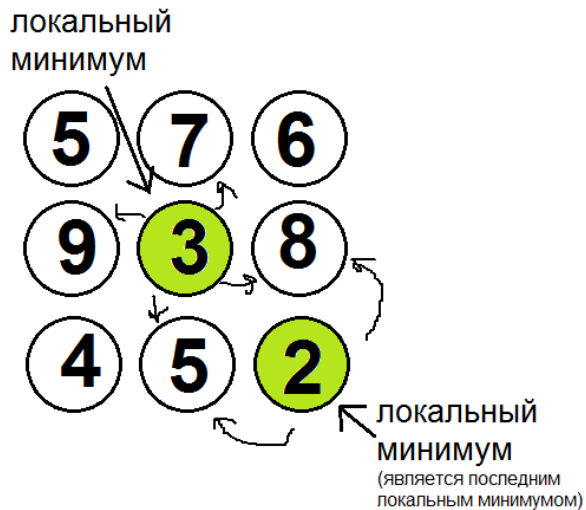
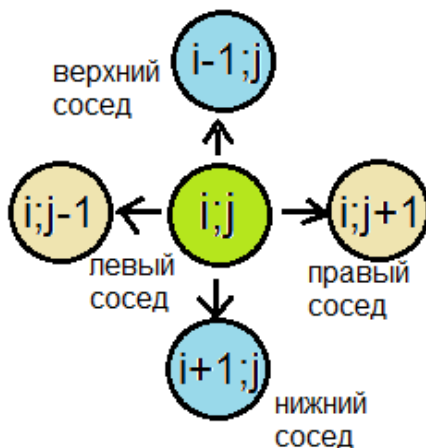
The Sum of elements of columns with extreme elements: 9

Test04

...

**оследний локальный минимум** [The last local minimum]. Дана математическая прямоугольная матрица размером N на M. Необходимо разработать функцию (или программу), которая находит и возвращает местоположение последнего локального минимума. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным) и тестовые данные. Первоначальные данные матрицы могут быть автоматически сгенерированы с помощью встроенного генератора псевдослучайных чисел.

В математике **локальным минимумом** называется элемент матрицы, значение которого строго меньше его непосредственного окружения, т.е. соседних элементов: левого, правого, верхнего и нижнего.



Тест 01

Введите размерность матрицы (N и M): **3 3**

Введите элементы матрицы:

Последний локальный минимум имеет координаты [3;3] и имеет значение 2.



Test 02

Input the matrix dimension (N and M): **2 3**

Enter the matrix elements:

**5 2 4**

**1 3 7**

The last local minimum has coordinates [2;1] and has a value of 1.

Test 03

Input the matrix dimension (N and M): **2 2**

Enter the matrix elements:

**1 1**

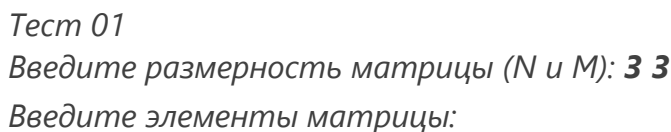
**1 1**

The matrix has no local minima.

Test04

...

В математике **локальным максимумом** называется элемент матрицы, значение которого строго больше его непосредственного окружения, т.е. соседних элементов: левого, правого, верхнего и нижнего.



Test 02

Input the matrix dimension (N and M): **2 3**

Enter the matrix elements:

**2 4**

**1 3 7**

The minimum maximum among all local n

Test 03

Input the matrix dimension (N and M): **2 2**

Enter the matrix elements:

**1 1**

**1 1**

The matrix has no local maxima.

Test04

...

**Сжатие матрицы** [*The matrix compression*]. Дана математическая прямоугольная матрица размером N на M. Необходимо разработать функцию (или программу), которая уплотняет заданную матрицу, удаляя из нее строки и столбцы, заполненные нулями. Ниже приведен рекомендуемый вид экрана программы (данные, введенные пользователем, выделены полужирным) и тестовые данные. Первоначальные данные матрицы могут быть автоматически сгенерированы с помощью встроенного генератора псевдослучайных чисел.

Тест 01

Введите размерность матрицы (N и M): **3 4**

Введите элементы матрицы:

Результирующая матрица после сжатия:

**1 2 3**

**2 1 1**

**2 7 1**

Test 02

Input the matrix dimension (N and M): **2 3**

Enter the matrix elements:

**1 0 2**

**3 1 0**

The resulting matrix after compression:

**1 0 2**

**3 1 0**

Test 03

Input the matrix dimension (N and M): **4 3**

Enter the matrix elements:

**0 0 0**

**2 0 1**

**0 0 0**

**2 0 -1**

The resulting matrix after compression:

**2**

**2 -1**

**1**

*Test04*

...

*Best of LUCK with it, and remember to HAVE FUN while you're learning :)*  
*Victor Ivanchenko*

