**Практическое задание 04.** Решить поставленную задачу программирования по разделу «Многомерные массивы». Продумать организацию удобного ввода элементов многомерного массива. Продумать схему корректного вывода элементов многомерного массива.

**Учесть все возможные ограничения**, накладываемые не только на переменные, но и на участвующие в расчёте функциональные зависимости (если они указаны в задании по варианту).

**Ввести контроль исходных данных**. Реализовать схему «ввод до победного» без возможности прерывания. При некорректном вводе исходных данных заставлять пользователя вводить сведения до тех пор, пока он не введёт их корректно. Продумать побуждающие сообщения-подсказки, направляющие пользователя ко вводу корректных значений.

Использовать для контроля исходных данных различные состояния метода «*TryParse*», входящего в перечень доступных компонентов интересующего значащего (*valuable*) типа данных, например: *int, byte, float, double*.

Подобрать корректные тестовые примеры **в необходимом количестве\***.

Помнить о точности производимых вычислений при работе с вещественными значениями, например:

**«При заданной точности  , но ,**

**то есть , а ».**

Код составленного программного обеспечения сопроводить комментариями.

Отчёт по решённой задаче составить по схеме, изложенной в методических указаниях [Сафронов А.И., Зольникова Н.Н., Новиков В.Г. Составление отчётной документации по решённым задачам алгоритмизации и программирования: Учебно-методическое пособие для проведения аудиторных занятий по Учебной практике. – М.: РУТ (МИИТ), 2018. – 83 с.]. Скачать можно на сайте Национальной Электронной Библиотеки (НЭБ) *elibrary.ru* : <https://www.elibrary.ru/item.asp?id=46271697>.

Для удобства работы обучающихся далее представлена памятка основных разделов отчёта:

0. Титульный лист

1. Цель работы

2. Формулировка задачи

3. Блок-схема алгоритма

4. Подбор тестовых примеров

5. Листинг (код) программы

6. Расчёт тестовых примеров на ПК

7. Вывод по работе

Блок-схему алгоритма оформлять согласно правилам, изложенным в разделе 8 того же учебно-методического пособия.

Для ввода и вывода элементов одномерных массивов может быть применена упрощённая схема иллюстрирования: типовой фрагмент «циклический ввод» – заменён одним блоком «Данные», типовой фрагмент «циклический вывод» – заменён одним блоком «Документ».

\*Пример корректного подбора тестовых примеров в нужном количестве представлен в разделе 7 Сборника задач [Сафронов А.И., Зольникова Н.Н., Новиков В.Г. Получение первичных профессиональных умений и навыков научно-исследовательской деятельности: Сборник задач для проведения аудиторных занятий по Учебной практике. – М.: РУТ (МИИТ), 2019. – 91 с.]. Скачать можно на сайте Национальной Электронной Библиотеки (НЭБ) *elibrary.ru* : <https://www.elibrary.ru/item.asp?id=46307421>.

**Принять во внимание тот факт, что все необходимые для тестирования значения могут быть занесены сразу в один единственный многомерный массив.**

**Вывод массивов с размерностями, старшими второй, реализовывать постранично / послойно с остановками по Console.ReadKey(true);. Возврат к предыдущей странице не реализовывать – двигаться только вперёд.**

**После ввода с клавиатуры данных в массив – очищать окно консоли по Console.Clear();**

**Вывод исходных и модифицированных массивов вести строго друг под другом для обеспечения возможности их сопоставления.**

**Для привлечения внимания к наиболее существенным изменениям элементов рекомендуется не забывать про возможности изменения цветов фона и шрифта консоли.**

Материалы, подготавливаемые к отправке по электронной почте, оформляются согласно Приложению 4 того же Сборника задач.

Настоятельно рекомендовано перед решением задачи ознакомиться с рекомендациями, собранными в Приложении 3 того же Сборника задач.

Не забывать давать значимые имена переменным, проектам, решениям, файлам и другим разрабатываемым компонентам. При нежелании использовать значимые имена переменных в отчёте необходимо составить таблицу соответствия вида:

|  |  |  |
| --- | --- | --- |
| № п/п | Наименование переменной в программе | Смысловое содержание используемой переменной |
| 1 | *a* | Ускорение |
| 2 | *v* | Скорость |
| 3 | *S* | Пройдённый путь |
| 4 | *t* | Затраченное время |
| 5 | *x* | Абсцисса точки на Декартовой плоскости |
| 6 | *y* | Ордината точки на Декартовой плоскости |
| 7 | *z* | Аппликата точки в пространстве |

Проанализировать исходные данные. Определить, какие их переменных являются аргументами функции, а которые поступают в формате констант для расчёта зависимости. На константы наложить ограничения на этапе ввода исходных данных, переменные значения контролировать на каждой итерации\*\* цикла (циклов).

\*\*Итерация в программировании – в широком смысле – организация обработки данных, при которой действия повторяются многократно, не приводя при этом к вызовам самих себя (в отличие от рекурсии). В узком смысле – один шаг итерационного, циклического вычислительного процесса.

**Внимание! При выявлении противоречивых условий среди вариантов индивидуального задания не предпринимать самостоятельных решений по устранению противоречий, а обратиться за консультацией к преподавателю, читающему курс «Программирования и основ алгоритмизации».**

***Корректный ввод элементов двумерного массива в общем виде программного обеспечения имеет формат:***

*«Введите количество строк двумерного массива: [целое числовое значение]»*

*«Введите количество столбцов двумерного массива: [целое числовое значение]»*

*«Введите 1-й элемент 1-й строки массива: [числовое значение]»*

*«Введите 2-й элемент 1-й строки массива: [числовое значение]»*

*…*

*«Введите 1-й элемент 3-й строки массива: [числовое значение]»*

*«Введите 2-й элемент 3-й строки массива: [числовое значение]»*

*… … … …*

*«Введите [целое числовое значение]-й элемент [целое числовое значение]-й строки массива: [числовое значение]»*

***Корректный вывод элементов двумерного массива в общем виде программного обеспечения имеет формат:***

|*1.40 -0.10 3.33 ... -9.37*|

*[имя одномерного массива] =* |*6.38 5.13 6.78 ... -10.29*|

| *... ... ... ... ...*|

|*4.50 -7.02, 2.64, ..., -0.09*|

**Задание состоит только из Индивидуальной части. Общая часть в нём отсутствует.**

**Вариант №1**

Перемножить матрицы *E* и *P*, где *Е* – единичная матрица, *P* – пользовательская матрица. Перемножение реализовать поэлементно.

**Вариант №2**

Реализовать «выкройку» недиагональных элементов одной матрицы и «вклеивание» их поверх элементов другой матрицы. «Выкройка» подразумевает выставление на позициях элементов значений, равных «-1».

**Вариант №3**

Вычислить определитель любой матрицы размерностью большей, чем [3x3]. Учесть ограничение на размерность матрицы при указании соответствующего значения в процессе исполнения программы.

**Вариант №4**

Разложить заданную матрицу на миноры всех её элементов.

**Вариант №5**

Вывести трёхмерный массив целых чисел (со значениями из диапазона от 1 до 9 включительно) по спирали. Последний элемент расположить в центре окна консоли, остальные «раскручивать» вокруг него по часовой стрелке.

**Вариант №6**

Дана матрица размерностью [*NxM*], состоящая только из натуральных чисел. Выбрать в каждой строке матрицы свой наименьший элемент и переставить его в первый столбец той же строки.

**Вариант №7**

Реализовать «выкройку» диагональных элементов одной матрицы и «вклеивание» их поверх элементов другой матрицы. «Выкройка» подразумевает обнуление элементов.

**Вариант №8**

Выполнить возведение заданной матрицы в заданную степень при условии, что все операции умножения заменяются операциями сложения, а все операции сложения – операциями поиска минимального элемента между парой значений. Операции реализовать алгоритмически без переопределения операторов языка.

**Вариант №9**

Выполнить возведение заданной матрицы в заданную степень. Перемножение произвести поэлементно, а не по известным правилам перемножения из курса Высшей математики.

**Вариант №10**

Дана целочисленная матрица из *N* строк и *M* столбцов (*1 < N <= 100*, *1 < M <= 50*). Выполнить поиск количества различных (неповторяющихся, уникальных) элементов матрицы.

**Вариант №11**

Вывести трёхмерный массив по спирали. Первый элемент расположить в центре окна консоли.

**Вариант №12**

Выполнить транспонирование введённой матрицы.

**Вариант №13**

Реализовать перелистывание слоёв трёхмерного массива стрелочными курсорами «влево» и «вправо» с клавиатуры.

**Вариант №14**

Вывести трёхмерный массив «змейкой». Первый элемент расположить в левом верхнем углу окна консоли.

**Вариант №15**

Упорядочить двумерный массив по возрастанию. В полученном массиве заменить все элементы с чётными индексами элементами с нечётными индексами с сохранением порядка следования этих элементов.

**Вариант №16**

Выполнить возведение заданной матрицы размерностью [6x6] с нулевыми элементами на главной диагонали в седьмую степень при условии, что все операции умножения заменяются операциями сложения, а все операции сложения операциями поиска максимального элемента между парой значений. Полученную матрицу умножить на исходную, в которой элементы главной диагонали заменены на 1000.

**Вариант №17**

Выполнить возведение заданной матрицы в заданную степень при условии, что все операции умножения заменяются операциями вычитания, а все операции сложения операцией поиска максимального элемента между парой значений.

**Вариант №18**

Получить расчёт обратной матрицы для заданной.

**Вариант №19**

Заполнить трёхмерный массив с помощью датчика случайных чисел таким образом, чтобы все элементы в пределах слоя оказались различными (уникальными).

**Вариант №20**

В заданной матрице найти: сумму минимального и максимального элементов; количество отрицательных элементов, стоящих на чётных позициях. Обратить внимание на то, что индексация считается с нуля, а позиции – с единицы.

**Вариант №21**

Заполнить матрицу порядка *n* автоматически по шаблону:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | ... | n-2 | n-1 | n |
| 2 | 1 | 2 | ... | n-3 | n-2 | n-1 |
| 3 | 2 | 1 | ... | n-4 | n-3 | n-2 |
| ... | ... | ... | ... | ... | ... | ... |
| n-1 | n-2 | n-3 | ... | 2 | 1 | 2 |
| n | n-1 | n-2 | ... | 3 | 2 | 1 |

**Вариант №22**

Заполнить трёхмерный массив с помощью датчика случайных чисел таким образом, чтобы абсолютно все элементы в нём оказались различными (уникальными).

**Вариант №23**

Сложить матрицы *E* и *P*, где *E* – единичная матрица, *P* – пользовательская матрица. Вывести результат «змейкой».

**Вариант №24**

Вывести трёхмерный массив символов (со значениями из диапазона от «а» до «я» включительно) по спирали. Первый элемент расположить в центре окна консоли, остальные «раскручивать» вокруг него против часовой стрелки. Подобрать в качестве примера такие элементы массива, которые при наличии в нём 36 элементов выведут на экран прореженные 6 слов, состоящих из 6 букв, читаемых слева направо.

**Вариант №25**

Дана матрица размерностью [*NxM*], состоящая только из натуральных чисел. Выбрать в каждой строке матрицы свой наименьший элемент и переставить его в первый столбец строки, зеркальной к рассматриваемой относительно границ массива. Подобная зеркальность математически записывается как *x(i) = x(N-i)*.

**Вариант №26**

Заполнить матрицу порядка *n* автоматически по шаблону:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | ... | n-2 | n-1 | n |
| 0 | 1 | 2 | ... | n-3 | n-2 | n-1 |
| 0 | 0 | 1 | ... | n-4 | n-3 | n-2 |
| ... | ... | ... | ... | ... | ... | ... |
| 0 | 0 | 0 | ... | 0 | 1 | 2 |
| 0 | 0 | 0 | ... | 0 | 0 | 1 |

**Вариант №27**

Упорядочить двумерный массив по убыванию. В полученном массиве заменить все элементы с чётными индексами значениями 1 / *x*, где *x* – значение элемента.

**Вариант №28**

Заполнить трёхмерный массив с помощью датчика случайных чисел таким образом, чтобы все элементы в пределах только центрального слоя оказались различными (уникальными).

**Вариант №29**

Реализовать обмен элементами двух матриц по следующему правилу: «треугольник» над главной диагональю первой матрицы переносится в «треугольник» под главной диагональю второй матрицы и «треугольник» над главной диагональю второй матрицы переносится в «треугольник» под главной диагональю первой матрицы.

**Вариант №30**

Вывести трёхмерный массив целых чисел (со значениями из диапазона от -99 до 100 включительно) по спирали. Последний элемент расположить в центре окна консоли, остальные «раскручивать» вокруг него по часовой стрелке.