Пермский филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский университет «Высшая школа экономики»

*Факультет экономики, менеджмента и бизнес-информатики*

Шульжик Кирилл Владимирович

**Отчёт по практической работе №5 (Вариант 16)**

студента образовательной программы бакалавриата «Программная инженерия» по направлению подготовки 09.03.04 Программная инженерия

Преподаватель

В.Д Марквирер

Оглавление

[Задание 3](#_Toc90648112)

[Анализ 4](#_Toc90648113)

[Проектирование 7](#_Toc90648114)

[Листинг 9](#_Toc90648115)

[Тесты 10](#_Toc90648116)

[ПРИЛОЖЕНИЕ А *Листинг* 37](#_Toc90648117)

Задание

Дано следующее задание:

1. Сформировать динамический одномерный массив, заполнить его случайными числами и вывести на печать.
2. Выполнить указанное в варианте задание и вывести полученный массив на печать.
3. Сформировать динамический двумерный массив, заполнить его случайными числами и вывести на печать.
4. Выполнить указанное в варианте задание и вывести полученный массив на печать.
5. Сформировать динамический рваный массив, заполнить его случайными числами и вывести на печать.
6. Выполнить указанное в варианте задание и вывести полученный массив на печать.

**При реализации функций необходимо продемонстрировать использование параметров разных типов и различные способы организации функций (параметры по умолчанию, перегрузку функций, и т.д.)**

Таблица 1 – Задание для 16 варианта

| **№ варианта** | **Одномерный массив** | **Двумерный массив** | **Рваный массив** |
| --- | --- | --- | --- |
| 16 | Добавить после каждого отрицательного элемента его модуль | Удалить все четные столбцы | Добавить К строк, начиная с номера N |

Задание представлено в таблице 1

Анализ

Таблица 2 – Анализ

| **№** | **Описание функции** | **Входные данные** | **Классы входных данных** | **Выходные данные** |
| --- | --- | --- | --- | --- |
| **Основные функции** | | | | |
| **1** | Формирование динамического одномерного массива с помощью датчика случайных чисел | Одномерный массив arr1 (ссылка на массив) | Целочисленный массив |  |
| **2** | Формирование динамического одномерного массива с клавиатуры | Одномерный массив arr1 (ссылка на массив) | Целочисленный массив |  |
| **3** | Добавление в массив после каждого отрицательного элемента его модуль | Одномерный массив arr1 (ссылка на массив) | Целочисленный массив |  |
| **4** | Формирование динамического двумерного массива с помощью датчика случайных чисел | Двумерный массив arr2 (ссылка на массив) | Целочисленный массив |  |
| **5** | Формирование динамического двумерного массива с клавиатуры | Двумерный массив arr2 (ссылка на массив) | Целочисленный массив |  |
| **6** | Удаление всех четных столбцов массива | Двумерный массив arr2 (ссылка на массив) | Целочисленный массив |  |
| **7** | Формирование динамического рваного массива с помощью датчика случайных чисел | Рваный массив arr3 (ссылка на массив) | Целочисленный массив |  |
| **8** | Формирование динамического рваного массива с клавиатуры | Рваный массив arr3 (ссылка на массив) | Целочисленный массив |  |
| **9** | Добавление K строк, начиная с номера N с помощью датчика случайных чисел | Рваный массив arr3 (ссылка на массив) | Целочисленный массив | Целочисленный одномерный массив (как способ передачи целочисленных переменных из тела функции) |
| **10** | Добавление K строк, начиная с номера N с клавиатуры | Рваный массив arr3 (ссылка на массив) | Целочисленный массив | Целочисленный одномерный массив (как способ передачи целочисленных переменных из тела функции) |
| **11** | Печать одномерного массива | Одномерный массив | Целочисленный |  |
| Cтрока-сообщение msg для взаимодействия с пользователем (необязательный параметр) | Строковый |
| **12** | Печать двумерного массива | Двумерный массив arr2 | Целочисленный |  |
| Строка-сообщение msg для взаимодействия с пользователем (необязательный параметр) | Строковый |
| **13** | Печать рваного массива | Рваный массив arr3 | Целочисленный |  |
| Строка-сообщение msg для взаимодействия с пользователем (необязательный параметр) | Строковый |
| **Вспомогательные функции** | | | | |
| **1** | Печать стартового меню |  |  |  |
| **2** | Печать меню для работы с одномерными массивами |  |  |  |
| **3** | Печать меню для работы с двумерными массивами |  |  |  |
| **4** | Печать меню для работы с рваными массивами |  |  |  |
| **5** | Ввод целочисленной переменной | Строка-сообщение msg для взаимодействия с пользователем (необязательный параметр) | Строковый | Целочисленная переменная |
| **6** | Ввод переменных в строку одномерного массива через пробел | Строка-сообщение msg для взаимодействия с пользователем (необязательный параметр) | Строковый | Одномерный целочисленный массив |

Анализ представлен в таблице 2

Проектирование

Таблица 3 – Проектирование

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Функция** | **Входные данные** | **Выходные данные** |
| **Основные функции** | | | |
| **1** | MakeRandomMas1 (формирование динамического одномерного массива с помощью датчика случайных чисел) | ref int[] arr1 – одномерный целочисленный массив (ссылка на массив) |  |
| **2** | MakeManualMas1 (формирование динамического одномерного массива с клавиатуры | ref int[] arr1 – одномерный целочисленный массив (ссылка на массив) |  |
| **3** | AbsFunction1 (добавление в одномерный целочисленный массив после каждого отрицательного элемента его модуль) | ref int[] arr1 – одномерный целочисленный массив (ссылка на массив) |  |
| **4** | MakeRandomMas2 (формирование динамического двумерного массива с помощью датчика случайных чисел) | ref int[,] arr2 – двумерный целочисленный массив (ссылка на массив) |  |
| **5** | MakeManualMas2 (формирование динамического двумерного массива с клавиатуры) | ref int[,] arr2 – двумерный целочисленный массив (ссылка на массив) |  |
| **6** | DelNechetColumns2 (удаление всех четных столбцов двумерного целочисленного массива) | ref int[,] arr2 – двумерный целочисленный массив (ссылка на массив) |  |
| **7** | MakeRandomMas3 (формирование динамического рваного массива с помощью датчика случайных чисел) | ref int[][] arr3 – рваный целочисленный массив (ссылка на массив) |  |
| **8** | MakeManualMas3 (формирование динамического рваного массива с клавиатуры) | ref int[][] arr3 – рваный целочисленный массив (ссылка на массив) |  |
| **9** | AddStringsRandomly3 (Добавление K строк, начиная с номера N в рваный целочисленный массив с помощью датчика случайных чисел) | ref int[][] arr3 – рваный целочисленный массив (ссылка на массив) | int[] elem – целочисленный одномерный массив (как способ передачи целочисленных переменных из тела функции) |
| **10** | AddStringsManually3 (добавление K строк, начиная с номера N в рваный целочисленный массив с клавиатуры) | int[][] arr3 – рваный целочисленный массив (ссылка на массив) | int[] elem – целочисленный одномерный массив (как способ передачи целочисленных переменных из тела функции) |
| **11** | PrintMas (печать одномерного массива) | int[] arr – одномерный целочисленный массив |  |
| string msg – строка-сообщение для взаимодействия с пользователем (необязательный параметр) |
| **12** | PrintMas (печать двумерного массива) | int[,] arr – двумерный целочисленный массив |  |
| string msg – строка-сообщение для взаимодействия с пользователем (необязательный параметр) |
| **13** | PrintMas (печать рваного массива) | int[][] arr – рваный целочисленный массив |  |
| string msg – строка-сообщение для взаимодействия с пользователем (необязательный параметр) |
| **Вспомогательные функции** | | | |
| **1** | MenuStart (печать стартового меню) |  |  |
| **2** | MenuOdnMass1 (печать меню для работы с одномерными массивами) |  |  |
| **3** | MenuOdnMass2  (печать меню для работы с двумерными массивами) |  |  |
| **4** | MenuOdnMass3  (печать меню для работы с рваными массивами) |  |  |
| **5** | CheckInt (ввод целочисленной переменной) | out int x – выходной параметр | int x – целочисленная переменная |
| string msg – строка-сообщение для взаимодействия с пользователем (необязательный параметр) |
| int left (левая граница (необязательный параметр)) |
| int right (правая граница (необязательный параметр)) |
| **6** | AddWithSpace (ввод переменных в строку одномерного массива через пробел) | string msg – строка-сообщение для взаимодействия с пользователем (необязательный параметр) | int[] arr – целочисленный одномерный массив |

Проектирование представлено в таблице 3

**Листинг**

См. [ПРИЛОЖЕНИЕ А](#Приложение).

Тесты

ПРИМЕЧАНИЕ. Тестирование ввода нецелочисленных переменных в различных функциях, за исключением функции, предназначенной для ввода целочисленной переменной, не будет производится, так как в этих функция ввод переменной производится через функцию, предназначенную для ввода целочисленной переменной.

Таблица 4 – Набор тестов для функции, предназначенной для формирования динамического одномерного массива с помощью датчика случайных чисел

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | -5 | Ошибка | Введите целое число, большее или равное 0 |
| 2 | 0 | Пустой массив/Повторный ввод | Массив пустой, дальнейшие действия не имеют смысла, сформируйте массив заново. Введите длину массива |
| 3 | 4 | Массив сформирован | Массив сформирован. Нажмите любую клавишу, чтобы продолжить |

Набор тестов для функции, предназначенной для формирования динамического одномерного массива с помощью датчика случайных чисел представлен в таблице 4.

Таблица 5 – Тестирование функции, предназначенной для формирования динамического одномерного массива с помощью датчика случайных чисел, по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 | Т4 |
| --- | --- | --- | --- | --- |
| Длина массива меньше нуля |  | + |  |  |
| Длина массива больше нуля |  |  |  | + |
| Длина массива равна нулю |  |  | + |  |
| Длина массива - не число | + |  |  |  |

Тестирование функции, предназначенной для формирования динамического одномерного массива с помощью датчика случайных чисел, по критериям черного ящика представлено в таблице 5.

Таблица 6 – Набор тестов для функции, предназначенной для Добавления в одномерный целочисленный массив после каждого отрицательного элемента его модуль

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | -11 87 -85 -2 -31 | -11 11 87 -85 85 -2 2 -31 31 | -11 11 87 -85 85 -2 2 -31 31 |
| 2 | 1 5 5 | 1 5 5 | 1 5 5 |
| 3 | -1 | -1 1 | -1 1 |

Набор тестов для функции, предназначенной для Добавления в одномерный целочисленный массив после каждого отрицательного элемента его модуль представлен в таблице 6.

Таблица 7 – Тестирование функции, предназначенной для добавления в одномерный целочисленный массив после каждого отрицательного элемента его модуль, по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 |
| --- | --- | --- | --- |
| Исходный массив содержит несколько отрицательных чисел | + |  |  |
| Исходный массив содержит одно отрицательное число |  |  | + |
| Исходный массив не содержит отрицательных чисел |  | + |  |

Тестирование функции, предназначенной для добавления в одномерный целочисленный массив после каждого отрицательного элемента его модуль, по критериям черного ящика представлено в таблице 7.

Таблица 8 – Набор тестов для функции, предназначенной для Формирование динамического двумерного массива с помощью датчика случайных чисел

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 |  |  | Введите количество строк |
| -4 | Ошибка/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Ошибка – пустой массив/повторный ввод | Массив пустой, дальнейшие действия не имеют смысла, введите количество строк заново |
| 4 | Ввод кол-ва столбцов | Введите количество столбцов |
| -4 | Ошибка/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Ошибка – пустой массив | Массив пустой, дальнейшие действия не имеют смысла, введите количество столбцов заново |
| 4 | Массив сформирован | Массив сформирован, нажмите любую клавишу, чтобы продолжить |

Набор тестов для функции, предназначенной для Формирование динамического двумерного массива с помощью датчика случайных чисел представлен в таблице 8

Таблица 9 – Тестирование функции, предназначенной для формирования динамического двумерного массива с помощью датчика случайных чисел, по критериям черного ящика

| Критерии | Т1 |
| --- | --- |
| Характеристика количества строк | |
| Количество строк – отрицательное число | + |
| Количество строк равно 0 | + |
| Количество строк – положительное число | + |
| Характеристика количества столбцов |  |
| Количество столбцов – отрицательное число | + |
| Количество столбцов – равно 0 | + |
| Количество столбцов – положительное число | + |

Тестирование функции, предназначенной для формирования динамического двумерного массива с помощью датчика случайных чисел, по критериям черного ящика представлено в таблице 9

Таблица 10 – Набор тестов для функции, предназначенной для формирование динамического двумерного массива с клавиатуры

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 |  |  | Введите количество строк |
| -4 | Ошибка/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Ошибка – пустой массив/повторный ввод | Массив пустой, дальнейшие действия не имеют смысла, введите количество строк заново |
| 4 | Ввод кол-ва столбцов | Введите количество столбцов |
| -4 | Ошибка/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Ошибка – пустой массив | Массив пустой, дальнейшие действия не имеют смысла, введите количество столбцов заново |
| 4 | Массив сформирован | Массив сформирован, нажмите любую клавишу, чтобы продолжить |

Набор тестов для функции, предназначенной для формирование динамического двумерного массива с клавиатуры представлен в таблице 10

Таблица 11 – Тестирование функции, предназначенной для формирования динамического двумерного массива с клавиатуры, по критериям черного ящика

| Критерии | Т1 |
| --- | --- |
| Характеристика количества строк | |
| Количество строк – отрицательное число | + |
| Количество строк равно 0 | + |
| Количество строк – положительное число | + |
| Характеристика количества столбцов | |
| Количество столбцов – отрицательное число | + |
| Количество столбцов – равно 0 | + |
| Количество столбцов – положительное число | + |

Тестирование функции, предназначенной для формирования динамического двумерного массива с клавиатуры, по критериям черного ящика представлено в таблице 11

ПРИМЕЧАНИЕ. Набор тестов и тестирование по критериям ЧЯ для функции, предназначенной для формирования динамического двумерного массива с клавиатуры (см. таблицу 10,11), и для функции, предназначенной для формирования динамического двумерного массива с помощью ДСЧ (см. таблицу 8,9), одинаковы, поскольку за ввод элементов двумерного массива в функции, предназначенной для формирования динамического двумерного массива с клавиатуры, отвечает другая функция.

Таблица 11 – Набор тестов для функции, предназначенной для удаления всех нечетных столбцов двумерного целочисленного массива

|  |  |  |  |
| --- | --- | --- | --- |
| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| 1 | 98 -60 12 20 -6  -43 7 -62 -92 86  -32 -20 -18 43 -51  0 -96 -34 -30 -37  57 -74 -88 -6 46 | массив с удалёнными чётными столбцами | 98 12 -6  -43 -62 86  -32 -18 -51  0 -34 -37  57 -88 46 |
| 2 | 98 12 -6  -43 -62 86  -32 -18 -51  0 -34 -37  57 -88 46 | массив с удалёнными чётными столбцами | 98 -6  -43 86  -32 -51  0 -37  57 46 |
| 3 | 98 -6  -43 86  -32 -51  0 -37  57 46 | массив с удалёнными чётными столбцами | 98  -43  -32  0  57 |
| 4 | 98  -43  -32  0  57 | Исходный массив | 98  -43  -32  0  57 |

Набор тестов для функции, предназначенной для удаления всех нечетных столбцов двумерного целочисленного массива, представлен в таблице 11

Таблица 12 – Тестирование функции, предназначенной для удаление всех нечетных столбцов двумерного целочисленного массива

| Критерии | Т1 | Т2 | Т3 | Т4 |
| --- | --- | --- | --- | --- |
| Массив состоит более, чем из 1 столбца | + | + | + |  |
| Массив состоит из 1 столбца |  |  |  | + |

Тестирование функции, предназначенной для удаление всех нечетных столбцов двумерного целочисленного массива, представлено в таблице 12

Таблица 13 – Набор тестов для функции, предназначенной для формирования динамического рваного массива с помощью датчика случайных чисел

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 |  |  | Введите количество строк |
| -4 | Сообщение об ошибке/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Пустой массив/повторный ввод | Массив пустой, дальнейшие действия не имеют смысла, введите количество строк заново |
| 3 | Ввод кол-ва элементов первой строки | Введите количество элементов 1-ой строки |
| -4 | Сообщение об ошибке/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Ввод кол-ва элементов второй строки | Введите количество элементов 2-ой строки |
| 0 | Сообщение об ошибке – пустой массив/повторный ввод | Введите положительное количество элементов 3-ой строки  (иначе дальнейшие действия не будут иметь смысла - пустой массив) |
| 4 | Массив сформирован | Массив сформирован  Нажмите любую клавишу, чтобы продолжить |
| 2 |  |  | Введите количество строк |
| 3 | Ввод кол-ва элементов первой строки | Введите количество элементов 1-ой строки |
| 2 | Ввод кол-ва элементов второй строки | Введите количество элементов 2-ой строки |
| 2 | Ввод кол-ва элементов третьей строки | Введите положительное количество элементов 3-ой строки |
| 2 | Массив сформирован | Массив сформирован |

Набор тестов для функции, предназначенной для формирования динамического рваного массива с помощью датчика случайных чисел, представлен в таблице 13

Таблица 14 – Тестирование функции, предназначенной для формирования динамического рваного массива с помощью датчика случайных чисел

| Критерии | Т1 | Т2 |
| --- | --- | --- |
| Характеристика количества строк | | |
| Количество строк – отрицательное число | + |  |
| Количество строк равно 0 | + |  |
| Количество строк – положительное число | + | + |
| Характеристика количества элементов строк | | |
| Количество элементов всех строк, кроме последней, равно 0 | + |  |
| Количество элементов всех строк не равно 0 |  | + |

Тестирование функции, предназначенной для формирования динамического рваного массива с помощью датчика случайных чисел представлено в таблице 14

Таблица 15 – Набор тестов для функции, предназначенной для формирования динамического рваного массива с клавиатуры

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 |  | Ввод количества строк | Введите количество строк |
| -3 | Сообщение об ошибке/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Пустой массив/повторный ввод | Массив пустой, дальнейшие действия не имеют смысла, введите количество строк заново |
| 3 | Ввод элементов первой строки | Вводите элементы 1-ой строки через пробел |
|  | Ввод элементов второй строки | Вводите элементы 2-ой строки через пробел |
|  | Предупреждение о следующем вводе | Введите как минимум одно число |
| 4 | Массив сформирован | Массив сформирован |
| 2 | -3 | Сообщение об ошибке/повторный ввод | Введите целое число, большее или равное 0 |
| 0 | Пустой массив/повторный ввод | Массив пустой, дальнейшие действия не имеют смысла, введите количество строк заново |
| 3 | Ввод элементов первой строки | Вводите элементы 1-ой строки через пробел |
| 34 34 | Ввод элементов второй строки | Вводите элементы 2-ой строки через пробел |
| 234 324 | Ввод элементов третьей строки | Вводите элементы 3-ой строки через пробел |
| 23 -4 | Массив сформирован | Массив сформирован |

Набор тестов для функции, предназначенной для формирования динамического рваного массива с клавиатуры, представлен в таблице 15

Таблица 16 - Тестирование функции, предназначенной для формирования динамического рваного массива с клавиатуры

|  |  |  |
| --- | --- | --- |
| Критерии | Т1 | Т2 |
| Характеристика количества строк | | |
| Количество строк – отрицательное число | + |  |
| Количество строк равно 0 | + |  |
| Количество строк – положительное число | + | + |
| Характеристика количества элементов строк | | |
| Количество элементов всех строк, кроме последней, равно 0 | + |  |
| Количество элементов всех строк не равно 0 |  | + |

Тестирование функции, предназначенной для формирования динамического рваного массива с клавиатуры представлено в таблице 16

ПРИМЕЧАНИЕ. Набор тестов и тестирование по критериям ЧЯ для функции, предназначенной для формирования динамического рваного массива с клавиатуры, и для функции, предназначенной для формирования динамического рваного массива с помощью ДСЧ, одинаковы, поскольку за ввод элементов рваного массива в функции, предназначенной для формирования динамического рваного массива с клавиатуры, отвечает другая функция.

Таблица 17 – Набор тестов для функции, предназначенной для добавления K строк, начиная с номера N в рваный целочисленный массив с помощью датчика случайных чисел

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | Массив  75 0 -35 74 84  88 -31  -98 51 -1 | Массив с 1 добавленной строкой, начиная с номера 1, с заданной длиной строки | =42  75 0 -35 74 84  88 -31  -98 51 -1 |
| K = 1 |
| N = 1 |
| Кол-во элементов первой строки -1 |
| 2 | Массив  -42  75 0 -35 74 84  88 -31  -98 51 -1 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = 3 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| N = 5 | Ввод кол-ва элементов первой строки | Введите кол-во элементов первой строки |
| Кол-во элементов первой строки = 4 | Ввод кол-ва элементов второй строки | Введите кол-во элементов второй строки |
| Кол-во элементов второй строки = 2 | Ввод кол-ва элементов третьей строки | Введите кол-во элементов третьей строки |
| Кол-во элементов третьей строки = 3 | Массив с 3 добавленными строками, начиная с номера 5, с заданными длинами строк | -42  75 0 -35 74 84  88 -31  -98 51 -1  69 23 -63 -87  -91 66  -100 9 25 |
| 3 | Массив  64  -77 78 70 53 | Ввод K | Введите, сколько строк вы хотите добавить |
| 2 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| 2 | Ввод кол-ва элементов первой строки | Введите кол-во элементов первой строки |
| 2 | Ввод кол-ва элементов второй строки | Введите кол-во элементов второй строки |
| 3 | Массив с 2 добавленными строками, начиная с номера 2, с заданными длинами строк | 64  64 23  -76 1 98  -77 78 70 53 |
| 4 | Массив  -69  12 -35 -83  -40 23 -56 88 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = 0 | Исходный массив | 69  12 -35 -83  -40 23 -56 88 |
| 5 | Массив  22 -44  -51  55 -53 -82 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = -2 | Ошибка/повторный ввод | Введите число, большее или равное 0 |
| 2 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| -2 | Ошибка/повторный ввод | Введите число, большее или равное единице и меньшее или равное 4 |
| 2 | Ввод длины первой строки | Введите длину первой строки |
| -3 | Ошибка/повторный ввод | Введите число, большее или равное 0 |
| 5 | Ввод длины второй строки | Введите длину второй строки |
| 5 | Массив с K добавленными строками, начиная с номера N, с заданной длиной строки | 22 -44  -86 -45 -18 -29 12  -99 -5 49 -13 -70  -51  55 -53 -82 |

Набор тестов для функции, предназначенной для добавления K строк, начиная с номера N в рваный целочисленный массив с помощью датчика случайных чисел представлен в таблице 17

Таблица 18 - Тестирование функции, предназначенной для добавления K строк, начиная с номера N, в рваный целочисленный массив с помощью датчика случайных чисел по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 |
| --- | --- | --- | --- | --- | --- |
| Добавление K строк, начиная с 1 номера | + |  |  |  |  |
| Добавление K строк, начиная с номера, равного увеличенной на единицу длине массива |  | + |  |  |  |
| Добавление K строк, начиная с номера, не равного 1 и не равного длине массива, увеличенной на 1 |  |  | + |  |  |
| Добавление 0 строк |  |  |  | + |  |
| Ввод отрицательных значений K,N, отрицательных значений количества элементов строк |  |  |  |  | + |

Тестирование функции, предназначенной для добавления K строк, начиная с номера N, в рваный целочисленный массив с помощью датчика случайных чисел по критериям черного ящика представлено в таблице 18

Таблица 19 – Набор тестов для функции, предназначенной для добавления K строк, начиная с номера N в рваный целочисленный массив с клавиатуры

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | Массив  75 0 -35 74 84  88 -31  -98 51 -1 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = 1 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| N = 1 | Ввод элементов первой строки | Вводите элементы первой строки |
| -42 | Массив K добавленными строками, начиная с номера N, с заданными элементами | -42  75 0 -35 74 84  88 -31  -98 51 -1 |
| 2 | Массив  -42  75 0 -35 74 84  88 -31  -98 51 -1 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = 3 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| N = 5 | Ввод элементов первой строки | Вводите элементы первой строки через пробел |
| 69 23 -63 -87 | Ввод элементов второй строки | Вводите элементы второй строки через пробел |
| -91 66 | Ввод элементов третьей строки | Вводите элементы третьей строки через пробел |
| -100 9 25 | Массив K добавленными строками, начиная с номера N, с заданными элементами | -42  75 0 -35 74 84  88 -31  -98 51 -1  69 23 -63 -87  -91 66  -100 9 25 |
| 3 | Массив  64  -77 78 70 53 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = 2 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| N = 2 | Ввод элементов первой строки | Вводите элементы первой строки через пробел |
| 64 23 | Ввод элементов второй строки | Вводите элементы второй строки через пробел |
| -76 1 98 | Массив K добавленными строками, начиная с номера N, с заданными элементами | 64  64 23  -76 1 98  -77 78 70 53 |
| 4 | Массив  -69  12 -35 -83  -40 23 -56 88 | Ввод K | Введите, сколько строк вы хотите добавить |
| 0 | Исходный массив | -69  12 -35 -83  -40 23 -56 88 |
| 5 | Массив  22 -44  -51  55 -53 -82 | Ввод K | Введите, сколько строк вы хотите добавить |
| K = -2 | Ошибка/повторный ввод | Введите число, большее или равное 0 |
| 2 | Ввод N | Введите, начиная с какого номера вы хотите добавить строки |
| -2 | Ошибка/повторный ввод | Введите число, большее или равное единице и меньшее или равное 4 |
| 2 | Ввод элементов первой строки | Вводите элементы первой строки через пробел |
| -86 -45 -18 -29 12 | Ввод элементов второй строки | Вводите элементы второй строки через пробел |
| -99 -5 49 -13 -70 | Массив с K добавленными строками, начиная с номера N, с заданными элементами | 22 -44  -86 -45 -18 -29 12  -99 -5 49 -13 -70  -51  55 -53 -82 |

Набор тестов для функции, предназначенной для добавления K строк, начиная с номера N в рваный целочисленный массив с клавиатуры представлен в таблице 19

Таблица 20 - Тестирование функции, предназначенной для добавления K строк, начиная с номера N, в рваный целочисленный массив с клавиатуры по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 |
| --- | --- | --- | --- | --- | --- |
| Добавление K строк, начиная с 1 номера | + |  |  |  |  |
| Добавление K строк, начиная с номера, равного увеличенной на единицу длине массива |  | + |  |  |  |
| Добавление K строк, начиная с номера, не равного 1 и не равного длине массива, увеличенной на 1 |  |  | + |  |  |
| Добавление 0 строк |  |  |  | + |  |
| Ввод отрицательных значений K, N |  |  |  |  | + |

Тестирование функции, предназначенной для добавления K строк, начиная с номера N, в рваный целочисленный массив с клавиатуры по критериям черного ящика представлено в таблице 20

Таблица 21 – Набор тестов для функции, предназначенной для печати одномерного целочисленного массива

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | Массив 45 45 45  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  45 45 45 |
| 2 | Массив -4  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  -4 |
| 3 | Массив 45 -4 45  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  45 -4 45 |

Набор тестов для функции, предназначенной для печати одномерного целочисленного массива представлен в таблице 21

ПРИМЕЧАНИЕ. В функцию не будет входить массив нулевой длины, поэтому тесты для этого случая пропущены

Таблица 22 - Тестирование функции печати одномерного массива по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 |
| --- | --- | --- | --- |
| Длина массива = 1 | + |  |  |
| Длина массива больше 1 |  | + | + |
| Массив состоит из положительных элементов | + |  |  |
| Массив состоит из отрицательных элементов |  | + |  |
| Массив состоит и из отрицательных элементов, и из положительных элементов |  |  | + |

Тестирование функции печати одномерного массива по критериям черного ящика представлено в таблице 22

Таблица 23 – Набор тестов для функции, предназначенной для печати двумерного целочисленного массива

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | Массив  234234234 3  4 34  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  234234234 3  4 34 |
| 2 | Массив -4  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  -4 |
| 3 | Массив 45 -4  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  45 -4 |

Набор тестов для функции, предназначенной для печати двумерного целочисленного массива представлен в таблице 23

Таблица 24 – Тестирование функции печати двумерного массива по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 |
| --- | --- | --- | --- |
| Массив состоит из 1 строки и из 1 столбца |  | + |  |
| Массив состоит более чем из 1 строки и из 1 столбца | + |  | + |
| Массив состоит только из положительных элементов | + |  |  |
| Массив состоит только из отрицательных элементов |  | + |  |
| Массив состоит и из отрицательных элементов, и из положительных элементов |  |  | + |
| Длина максимального по модулю числа последовательности значительно превышает длину минимального | + |  |  |

Тестирование функции печати двумерного массива по критериям черного ящика представлено в таблице 24.

Таблица 25 – Набор тестов для функции, предназначенной для печати двумерного целочисленного массива

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | Массив  1 324234 3  1 3  1 2 2323  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  1 324234 3  1 3  1 2 2323 |
| 2 | Массив -4  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  -4 |
| 3 | Массив  45 45  4  Строка “Массив представлен ниже:” | Вывод строки, после – массива | Массив представлен ниже  45 45  4 |

Набор тестов для функции, предназначенной для печати двумерного целочисленного массива представлен в таблице 25.

Таблица 26 – Тестирование функции печати двумерного массива по критериям черного ящика

|  |  |  |  |
| --- | --- | --- | --- |
| Критерии | Т1 | Т2 | Т3 |
| Массив состоит из 1 строки и из 1 столбца |  | + |  |
| Массив состоит более чем из 1 строки | + |  | + |
| Массив состоит только из положительных элементов |  |  | + |
| Массив состоит только из отрицательных элементов |  | + |  |
| Массив состоит и из отрицательных элементов, и из положительных элементов | + |  |  |
| Длина максимального по модулю числа последовательности значительно превышает длину минимального | + |  |  |

Тестирование функции печати двумерного массива по критериям черного ящика представлено в таблице 26.

Таблица 27 – Набор тестов для функции, предназначенной для печати стартового меню

|  |  |  |  |
| --- | --- | --- | --- |
| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| 1 | 5 | Ошибка/повторный ввод | Введите целое число от 0 до 3! |
| 2 | 1 | Печать меню для работы с одномерными массивами | 1. Сформировать одномерный массив с помощью датчика случайных чисел  2. Сформировать одномерный массив вручную  3. Печать массива  4. Добавить после каждого отрицательного элемента его модуль  0. Назад |
| 3 | 2 | Печать меню для работы с двумерными массивами | 1. Сформировать двумерный массив с помощью датчика случайных чисел  2. Сформировать двумерный массив вручную  3. Печать массива  4. Удалить все четные столбцы  0. Назад |
| 4 | 3 | Печать меню для работы с рваными массивами | 1. Сформировать рваный массив с помощью датчика случайных чисел  2. Сформировать рваный массив вручную  3. Печать массива  4. Добавить K строк, начиная с номера N  0. Назад |
| 5 | 0 | Выход из программы | Благодарим за использование программы! |

Набор тестов для функции, предназначенной для печати стартового меню, представлен в таблице 27

Таблица 28 – Тестирование функции печати стартового меню

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 |
| --- | --- | --- | --- | --- | --- |
| Выбор 1 пункта меню |  | + |  |  |  |
| Выбор 2 пункта меню |  |  | + |  |  |
| Выбор 3 пункта меню |  |  |  | + |  |
| Выбор 4 пункта меню |  |  |  |  | + |
| Ввод символа, не соответствующего ни одному пункту меню | + |  |  |  |  |

Тестирование функции печати стартового меню представлено в таблице 28.

Таблица 29 – Набор тестов для функции, предназначенной для печати меню, предназначенного для работы с одномерными массивами

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | 5 | Ошибка/повторный ввод | Введите целое число от 0 до 4! |
| 2 | 1 | Переход в функцию, предназначенную для формирования одномерного целочисленного массива с помощью датчика случайных чисел | Введите длину массива |
| 3 | 2 | Переход в функцию, предназначенную для формирования динамического одномерного массива с клавиатуры | Вводите числа через пробел, после нажмите энтер |
| 4 | 3 | Переход в функцию, предназначенную для печати одномерного массива | Массив представлен ниже:  4 |
| 5 | 4 | Переход в функцию, предназначенную для добавления в массив после каждого отрицательного элемента его модуль | Массив представлен ниже:  4  Исходный массив:  4 |
| 6 | 0 | Возвращение в функцию, предназначенную для печати стартового меню | 1. Работа с одномерными массивами  2. Работа с двумерными массивами  3. Работа с рваными массивами  0. Выход из программы |

Набор тестов для функции, предназначенной для печати меню, предназначенного для работы с одномерными массивами, представлен в таблице 29

Таблица 30 – Тестирование функции печати меню, предназначенного для работы с одномерными массивами

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 |
| --- | --- | --- | --- | --- | --- | --- |
| Выбор 1 пункта меню |  | + |  |  |  |  |
| Выбор 2 пункта меню |  |  | + |  |  |  |
| Выбор 3 пункта меню |  |  |  | + |  |  |
| Выбор 4 пункта меню |  |  |  |  | + |  |
| Выбор 5 пункта меню |  |  |  |  |  | + |
| Ввод символа, не соответствующего ни одному пункту меню | + |  |  |  |  |  |

Тестирование функции, предназначенной для печати меню, предназначенного для работы с одномерными массивами представлено в таблице 30.

Таблица 31 – Набор тестов для функции, предназначенной для печати меню, предназначенного для работы с двумерными массивами

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | 5 | Ошибка/повторный ввод | Введите целое число от 0 до 4! |
| 2 | 1 | Переход в функцию, предназначенную для формирования динамического двумерного массива с помощью датчика случайных чисел | Введите длину массива |
| 3 | 2 | Переход в функцию, предназначенную для формирования динамического двумерного массива с клавиатуры | Вводите числа через пробел, после нажмите enter |
| 4 | 3 | Переход в функцию, предназначенную для печати двумерного массива | Массив представлен ниже:  4 8 5  4 |
| 5 | 4 | Переход в функцию, предназначенную для удаления всех четных столбцов двумерного целочисленного массива | Массив представлен ниже:  4 5  4 |
| 6 | 0 | Возвращение в функцию, предназначенную для печати стартового меню | 1. Работа с одномерными массивами  2. Работа с двумерными массивами  3. Работа с рваными массивами  0. Выход из программы |

Набор тестов для функции, предназначенной для печати меню, предназначенного для работы с двумерными массивами представлен в таблице 31

Таблица 32 – Тестирование функции печати меню, предназначенного для работы с двумерными массивами

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 |
| --- | --- | --- | --- | --- | --- | --- |
| Выбор 1 пункта меню |  | + |  |  |  |  |
| Выбор 2 пункта меню |  |  | + |  |  |  |
| Выбор 3 пункта меню |  |  |  | + |  |  |
| Выбор 4 пункта меню |  |  |  |  | + |  |
| Выбор 5 пункта меню |  |  |  |  |  | + |
| Ввод символа, не соответствующего ни одному пункту меню | + |  |  |  |  |  |

Тестирование функции печати меню, предназначенного для работы с двумерными массивами представлено в таблице 32.

Таблица 33 – Набор тестов для функции, предназначенной для печати меню, предназначенного для работы с рваными массивами

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | 5 | Ошибка/повторный ввод | Введите целое число от 0 до 4! |
| 2 | 1 | Переход в функцию, предназначенную для формирования динамического рваного массива с помощью датчика случайных чисел | Введите количество строк |
| 3 | 2 | Переход в функцию, предназначенную для формирования динамического рваного массива с клавиатуры | Введите количество строк |
| 4 | 3 | Переход в функцию, предназначенную для печати рваного массива | Массив представлен ниже:  234 234  234 324  4 |
| 5 | 4 | Переход в функцию, предназначенную для добавления K строк, начиная с номера N в рваный целочисленный массив с клавиатуры | 1. Добавить К строк, начиная с номера N с помощью датчика случайных чисел  2. Добавить K строк, начиная с номера N вручную |
| 6 | 0 | Возвращение в функцию, предназначенную для печати стартового меню | 1. Работа с одномерными массивами  2. Работа с двумерными массивами  3. Работа с рваными массивами  0. Выход из программы |

Набор тестов для функции, предназначенной для печати меню, предназначенного для работы с рваными массивами представлен в таблице 33

Таблица 34 – Тестирование функции печати меню, предназначенного для работы с рваными массивами

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 |
| --- | --- | --- | --- | --- | --- | --- |
| Выбор 1 пункта меню |  | + |  |  |  |  |
| Выбор 2 пункта меню |  |  | + |  |  |  |
| Выбор 3 пункта меню |  |  |  | + |  |  |
| Выбор 4 пункта меню |  |  |  |  | + |  |
| Выбор 5 пункта меню |  |  |  |  |  | + |
| Ввод символа, не соответствующего ни одному пункту меню | + |  |  |  |  |  |

Тестирование функции печати меню, предназначенного для работы с рваными массивами представлено в таблице 34.

Таблица 35 – Набор тестов для функции, предназначенной для ввода целочисленной переменной

| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| --- | --- | --- | --- |
| 1 | left = 2  right = 5  msg = “Ввод числа x” | Вывод сообщения | “Ввод числа x” |
| “фыва” | Ошибка/повторный ввод | Введите целое число, большее или равное 2 и меньшее или равное 5 |
| 0 | Ошибка/повторный ввод | Введите целое число, большее или равное 2 и меньшее или равное 5 |
| 7 | Ошибка/повторный ввод | Введите целое число, большее или равное 2 и меньшее или равное 5 |
| 3 | Присвоение переменной значение 3 | int x = 3 |
| 2 | left = 2  right = 3  msg: “Ввод числа х” | Вывод сообщения | “Ввод числа x” |
| 1 | Ошибка/повторный ввод | Введите либо 2, либо 3 |
| 2 | Присвоение переменной значение 2 | int x = 2 |
| 3 | left = 2  msg: “Ввод числа x” | Вывод сообщения | “Ввод числа x” |
| 1 | Ошибка/повторный ввод | Введите целое число, большее или равное 2 |
| 3 | Присвоение переменной значение 3 | int x = 3 |
| 4 | right = 5  msg: “Ввод числа x” | Вывод сообщения | “Ввод числа x” |
| 8 | Ошибка/повторный ввод | Введите целое число, меньшее или равное 5 |
| 3 | Присвоение переменной значение 3 | int x = 3 |
| 5 | msg: “Ввод числа x” | Вывод сообщения | “Ввод числа x” |
| dsg | Ошибка/повторный ввод | Введите целое число |
| 5 | Присвоение переменной значение 5 | int x = 5 |
| 6 | msg: “Ввод x”  left = 5  right = 5 | Вывод сообщения | “Ввод числа x” |
| 4 | Ошибка/повторный ввод | Введите число 5 |
| 5 | Присвоение переменной значение 5 | int x = 5 |
| 7 | left = 10  rigt = 5 | Сообщение об ошибке | Неверно заданы границы диапазона |

Набор тестов для функции, предназначенной для ввода целочисленной переменной представлен в таблице 35

Таблица 36 – Тестирование функции, предназначенной для ввода целочисленной переменной, по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 | Т7 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Ввод не целого числа | + |  |  |  | + |  |  |
| Ввод числа, не входящего в заданный диапазон | + | + | + | + |  | + |  |
| Заданный диапазон состоит из 1 числа |  |  |  |  |  | + |  |
| Левая граница превышает правую границу |  |  |  |  |  |  | + |

Тестирование функции, предназначенной для ввода целочисленной переменной, по критериям черного ящика представлено в таблице 36.

Таблица 37 – Набор тестов для функции, предназначенной для ввода переменных в строку одномерного массива через пробел

|  |  |  |  |
| --- | --- | --- | --- |
| №Теста | Входные данные | Ожидаемый результат | Выходные данные |
| 1 | 345 ап | Сообщение об ошибке/повторный ввод | Вводите только целые числа через пробел! |
| 2 | 345 345 -4 | Массив сформирован | Массив сформирован |
| 3 | 4 5 3 | Массив сформирован | Массив сформирован |
| 4 | 43534543543534 | Сообщение об ошибке/повторный ввод | Вводите только целые числа через пробел! |
| 5 | --3 | Сообщение об ошибке/повторный ввод | Вводите только целые числа через пробел! |
| 6 | 4.5 7.34 | Сообщение об ошибке/повторный ввод | Вводите только целые числа через пробел! |
| 7 | 2423ып 435 | Сообщение об ошибке/повторный ввод | Вводите только целые числа через пробел! |
| 8 | 3 34 5 | Массив сформирован | Массив сформирован |
| 9 | -34543543534543535 | Сообщение об ошибке/повторный ввод | Вводите только целые числа через пробел! |

Набор тестов для функции, предназначенной для ввода переменных в строку одномерного массива через пробел, представлен в таблице 37

Таблица 38 – Тестирование функции, предназначенной для ввода переменных в строку одномерного массива через пробел, по критериям черного ящика

| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 | Т7 | Т8 | Т9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ввод не числа через пробел | + |  |  |  |  |  |  |  |  |
| Ввод целых чисел через несколько пробелов |  | + | + |  |  |  |  |  |  |
| Ввод не числа слитно с другим числом |  |  |  |  |  |  | + |  |  |
| Ввод целых чисел через пробел |  |  |  |  |  |  |  | + |  |
| Ввод не целых чисел |  |  |  |  |  | + |  |  |  |
| Ввод числа с несколькими минусами |  |  |  |  | + |  |  |  |  |
| Ввод числа, превышающего максимальное значения типа int32 |  |  |  | + |  |  |  |  |  |
| Ввод числа, меньшего, чем минимальное значение типа int32 |  |  |  |  |  |  |  |  | + |

Тестирование функции, предназначенной для ввода переменных в строку одномерного массива через пробел, по критериям черного ящика представлено в таблице 38.

ПРИЛОЖЕНИЕ А

**Листинг**

using System;

namespace \_5\_лаба

{

class Program

{

static void MakeRandomMas1(ref int[] arr1)

{

CheckInt(out int size, msg: "Введите длину массива", left: 0);

while (size == 0)

{

Console.WriteLine("\nМассив пустой, дальнейшие действия не имеют смысла, сформируйте массив заново");

CheckInt(out size, msg: "\nВведите длину массива", left: 0);

}

Random rnd = new();

arr1 = new int[size];

for (int i = 0; i < size; i++)

{

arr1[i] = rnd.Next(-100, 100);

}

Console.WriteLine("\nМассив сформирован");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

static void PrintMas(int[] arr, string msg = "")

{

if (msg != "")

Console.WriteLine(msg + "\n");

foreach (int x in arr)

{

Console.Write(x + " ");

}

Console.WriteLine("\n");

}

static void CheckInt(out int x, string msg = "", int left = Int32.MinValue, int right = Int32.MaxValue)

{

x = 0;

int kok = 0;

if (right < left && right != Int32.MaxValue && left != Int32.MinValue)

{

Console.WriteLine("Неверно задана левая и правая граница");

kok = 4;

}

else if (right < left && right == Int32.MaxValue)

{

Console.WriteLine($"левая граница диапазона не может превышать {Int32.MaxValue}");

kok = 4;

}

else if (right < left && left == Int32.MinValue)

{

Console.WriteLine($"левая граница диапазона не может превышать {Int32.MaxValue}");

kok = 4;

}

else

{

if (msg != "")

Console.WriteLine(msg);

string bug = Console.ReadLine();

while ((!int.TryParse(bug, out x) || int.Parse(bug) < left || int.Parse(bug) > right) && kok != 4)

{

Console.WriteLine();

kok = 2;

if (left != Int32.MinValue && right != Int32.MaxValue && right - left == 1)

{

Console.WriteLine($"Введите либо {left}, либо {right}");

bug = Console.ReadLine();

}

else if (left != Int32.MinValue && right != Int32.MaxValue)

{

Console.WriteLine($"Введите целое число, большее или равное {left} и меньшее или равное {right}");

bug = Console.ReadLine();

}

else if (left != Int32.MinValue)

{

Console.WriteLine($"Введите целое число, большее или равное {left}");

bug = Console.ReadLine();

}

else if (right != Int32.MaxValue)

{

Console.WriteLine($"Введите целое число, меньшее или равное {right}");

bug = Console.ReadLine();

}

else

{

Console.WriteLine($"Введите целое число");

bug = Console.ReadLine();

}

}

if (kok == 2)

x = int.Parse(bug);

}

}

static void AbsFunction1(ref int[] arr)

{

int kolvo = 0;

foreach (int x in arr)

{

if (x < 0)

kolvo += 1;

}

int[] arr1 = new int[arr.Length + kolvo];

int k = 0;

for (int i = 0; i < arr.Length; i++)

{

arr1[k] = arr[i];

if (arr[i] < 0)

{

arr1[k + 1] = Math.Abs(arr[i]);

k += 1;

}

k += 1;

}

arr = arr1;

}

static void MakeRandomMas2(ref int[,] arr2)

{

CheckInt(out int strings, "Введите количество строк", left: 0);

while (strings == 0)

CheckInt(out strings, msg: "\nМассив пустой, дальнейшие действия не имеют смысла, введите количество строк заново", left: 0);

CheckInt(out int columns, "\nВведите количество столбцов", left: 0);

while (columns == 0)

CheckInt(out columns, msg: "\nМассив пустой, дальнейшие действия не имеют смысла, введите количество столбцов заново", left: 0);

arr2 = new int[strings, columns];

Random rand = new();

for (int i = 0; i < strings; i++)

{

for (int j = 0; j < columns; j++)

arr2[i, j] = rand.Next(-100, 100);

}

Console.WriteLine("\nМассиф сформирован");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

static void MakeManualMas2(ref int[,] arr2)

{

CheckInt(out int strings, "Введите количество строк", left: 0);

while (strings == 0)

CheckInt(out strings, msg: "\nМассив пустой, дальнейшие действия не имеют смысла, введите количество строк заново", left: 0);

CheckInt(out int columns, "\nВведите количество столбцов", left: 0);

while (columns == 0)

CheckInt(out columns, msg: "\nМассив пустой, дальнейшие действия не имеют смысла, введите количество столбцов заново", left: 0);

arr2 = new int[strings, columns];

int k = 1;

int[] arr = null;

for (int i = 0; i < strings; i++)

{

int j = 0;

if (k%10 == 3 && k%100 != 13)

arr = AddWithSpace($"\nВводите символы {k}-ей строки через пробел, после нажмите энтер");

else

arr = AddWithSpace($"\nВводите символы {k}-ой строки через пробел, после нажмите энтер");

k++;

while (arr.Length != columns)

{

if (arr.Length == 0)

arr = AddWithSpace($"Количество элементов строки должно быть равно {columns}");

else

arr = AddWithSpace($"\nКоличество элементов строки должно быть равно {columns}");

}

foreach (int x in arr)

{

arr2[i, j] = x;

j += 1;

}

}

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

static void DelChetColumns2(ref int[,] arr2)

{

int columns = 0;

for (int i = 0; i < arr2.GetLength(1); i++)

{

if (i % 2 == 0)

columns += 1;

}

int[,] arr\_copy = new int[arr2.GetLength(0), columns];

for (int i = 0; i < arr2.GetLength(0); i++)

{

int k = 0;

for (int j = 0; j < arr2.GetLength(1); j++)

{

if (j % 2 == 0)

{

arr\_copy[i, k] = arr2[i, j];

k += 1;

}

}

}

arr2 = arr\_copy;

}

static void MakeRandomMas3(ref int[][] arr3)

{

CheckInt(out int strings, "Введите количество строк", left: 0);

while (strings == 0)

CheckInt(out strings, msg: "\nМассив пустой, дальнейшие действия не имеют смысла," +

" введите количество строк заново", left: 0);

arr3 = new int[strings][];

Random rand = new();

int k = 1;

int z = 0;

for (int i = 0; i < strings; i++)

{

int x = 0;

if (i == strings - 1 && z == strings - 1)

{

if (k % 10 == 3 && k%100 != 13)

CheckInt(out x, msg: $"\nВведите положительное количество элементов {k}-ей строки\n(иначе " +

$"дальнейшие действия не будут иметь смысла – пустой массив)", left: 1);

else

CheckInt(out x, msg: $"\nВведите положительное количество элементов {k}-ой строки\n(иначе " +

$"дальнейшие действия не будут иметь смысла – пустой массив)", left: 1);

}

else

{

if (k % 10 == 3 && k % 100 != 13)

CheckInt(out x, msg: $"\nВведите количество элементов {k}-ей строки", left: 0);

else

CheckInt(out x, msg: $"\nВведите количество элементов {k}-ой строки", left: 0);

}

if (x == 0)

z += 1;

arr3[i] = new int[x];

for (int j = 0; j < x; j++)

arr3[i][j] = rand.Next(-100, 100);

k += 1;

}

Console.WriteLine("\nМассив сформирован");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

static void MakeManualMas3(ref int[][] arr3)

{

CheckInt(out int strings, "Введите количество строк", left: 0);

while (strings == 0)

CheckInt(out strings, msg: "\nМассив пустой, дальнейшие действия не имеют смысла," +

" введите количество строк заново", left: 0);

arr3 = new int[strings][];

int k = 0;

int[] arr1 = null;

for (int i = 0; i < strings; i++)

{

if (k != strings - 1)

{

if ((i + 1) % 10 == 3)

arr1 = AddWithSpace($"\nВводите элементы {i + 1}-ей строки через пробел");

else

arr1 = AddWithSpace($"\nВводите элементы {i + 1}-ой строки через пробел");

if (arr1.Length == 0)

k += 1;

arr3[i] = new int[arr1.Length];

arr1.CopyTo(arr3[i], 0);

}

else

{

Console.WriteLine("\nВведите как минимум одно число");

arr1 = AddWithSpace();

while (arr1.Length == 0)

{

Console.WriteLine("\nВведите как минимум одно число");

arr1 = AddWithSpace();

}

arr3[i] = new int[arr1.Length];

arr1.CopyTo(arr3[i], 0);

}

}

Console.WriteLine("\nМассив сформирован");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

static int[] AddWithSpace(string msg = "")

{

string str = "";

int check = 0;

Console.WriteLine(msg);

do

{

check = 0;

str = Console.ReadLine() + " ";

if (str.Length == 1)

break;

while (str.IndexOf(" ") != -1)

str = str.Replace(" ", " ");

int x; int k = 0; int gi;

if (str[str.Length - 2].ToString() != "-")

{

for (int i = 0; i < str.Length - 1; i++)

{

if (k == -2)

k = 0;

if (k == -4)

k = -2;

if (str[i] != ' ')

{

string ch = str[i].ToString();

if (ch == "-")

{

if(int.TryParse(str[i-1].ToString(), out gi))

{

Console.WriteLine("\nВводите только целые числа через пробел!");

check = -4;

break;

}

k = -4;

continue;

}

if (k == -2)

ch = str[i - 1].ToString() + str[i].ToString();

if (!int.TryParse(ch, out x) && k != -4)

{

Console.WriteLine("\nВводите только целые числа через пробел!");

check = -4;

break;

}

}

}

}

else

{

Console.WriteLine("\nВводите только целые числа через пробел!");

check = -4;

}

}

while (check == -4);

if (str == " ")

{

int[] arr1 = new int[0];

return arr1;

}

int arrSize = 0;

for (int i = 0; i < str.Length; i++)

{

if (str[i] == ' ' && i != 0)

{

arrSize++;

}

}

int[] arr = new int[arrSize];

int previousWhiteSpaceIndex = 0;

for (int i = 0, j = 0; i < str.Length; i++)

{

if (str[i] == ' ')

{

var word = str.Substring(previousWhiteSpaceIndex, i - previousWhiteSpaceIndex);

if (word.Length != 0)

{

if (!Int32.TryParse(word, out arr[j]))

AddWithSpace("\nВводите только целые числа через пробел!");

else

{

arr[j] = int.Parse(word);

j++;

}

}

previousWhiteSpaceIndex = i + 1;

}

}

return arr;

}

static void PrintMas(int[][] arr, string msg = "")

{

int znakomest = 0;

int min = Int32.MaxValue; int max = Int32.MinValue;

Console.WriteLine(msg + '\n');

for (int i = 0; i < arr.Length; i++)

{

foreach (int item in arr[i])

{

if (item > max)

max = item;

if (item < min)

min = item;

}

}

if (max.ToString().Length > min.ToString().Length)

znakomest = max.ToString().Length;

else

znakomest = min.ToString().Length;

switch (znakomest)

{

case 1:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],2}");

Console.WriteLine();

}

}

break;

case 2:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],3}");

Console.WriteLine();

}

}

break;

case 3:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],4}");

Console.WriteLine();

}

}

break;

case 4:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],5}");

Console.WriteLine();

}

}

break;

case 5:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],6}");

Console.WriteLine();

}

}

break;

case 6:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],7}");

Console.WriteLine();

}

}

break;

case 7:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],8}");

Console.WriteLine();

}

}

break;

case 8:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],9}");

Console.WriteLine();

}

}

break;

case 9:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],10}");

Console.WriteLine();

}

}

break;

case 10:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],11}");

Console.WriteLine();

}

}

break;

case 11:

for (int i = 0; i < arr.GetLength(0); i++)

{

if (arr[i].Length == 0)

continue;

else

{

for (int j = 0; j < arr[i].Length; j++)

Console.Write($"{arr[i][j],12}");

Console.WriteLine();

}

}

break;

}

}

static void PrintMas(int[,] arr, string msg = "")

{

if (arr.Length == 0)

Console.WriteLine("Пустой массив");

else

{

int znakomest = 0;

int min = Int32.MaxValue; int max = Int32.MinValue;

Console.WriteLine(msg + "\n");

foreach (int item in arr)

{

if (item > max)

max = item;

if (item < min)

min = item;

}

if (max.ToString().Length > min.ToString().Length)

znakomest = max.ToString().Length;

else

znakomest = min.ToString().Length;

switch (znakomest)

{

case 1:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],2}");

Console.WriteLine();

}

break;

case 2:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],3}");

Console.WriteLine();

}

break;

case 3:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],4}");

Console.WriteLine();

}

break;

case 4:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],5}");

Console.WriteLine();

}

break;

case 5:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],6}");

Console.WriteLine();

}

break;

case 6:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],7}");

Console.WriteLine();

}

break;

case 7:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],8}");

Console.WriteLine();

}

break;

case 8:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],9}");

Console.WriteLine();

}

break;

case 9:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],10}");

Console.WriteLine();

}

break;

case 10:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],11}");

Console.WriteLine();

}

break;

case 11:

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

Console.Write($"{arr[i, j],12}");

Console.WriteLine();

}

break;

}

}

}

static int[] AddStringsRandomly3(ref int[][] arr3)

{

CheckInt(out int K, left: 0, msg: "Введите, сколько строк вы хотите добавить");

if (K == 0)

{

int[] arr = { 0 };

return arr;

}

CheckInt(out int N, left: 1, right: arr3.Length+1, msg: "\nВведите, начиная с какого номера вы хотите добавить строки");

int[] elem = new int[K + 2];

elem[0] = K; elem[1] = N; int z = 2;

int[][] arr1 = new int[arr3.Length + K][];

int k = 0;

int skik = 0;

Random rand = new();

for (int i = 0; i < N-1; i++)

{

skik += 1;

arr1[i] = new int[arr3[i].Length];

arr3[i].CopyTo(arr1[i], 0);

}

int elemstrings = 0;

for (int i = N - 1; i < N-1+K; i++)

{

k++;

if (k%10 == 3 && k%100 != 13)

CheckInt(out elemstrings, left: 0, msg: $"\nВведите количество элементов {k}-ей строки");

else

CheckInt(out elemstrings, left: 0, msg: $"\nВведите количество элементов {k}-ой строки");

elem[z] = elemstrings;

z += 1;

arr1[i] = new int[elemstrings];

for (int j = 0; j < elemstrings; j++)

arr1[i][j] = rand.Next(-100, 100);

}

for (int i = N-1+k; i < arr1.Length; i ++)

{

arr1[i] = new int[arr3[skik].Length];

arr3[skik].CopyTo(arr1[i], 0);

skik += 1;

}

arr3 = arr1;

return elem;

}

static int[] AddStringsManually3(ref int[][] arr3)

{

CheckInt(out int K, left: 0, msg: "Введите, сколько строк вы хотите добавить");

if (K == 0)

{

int[] arr = { 0 };

return arr;

}

CheckInt(out int N, left: 1, right: arr3.Length+1, msg: "\nВведите, начиная с какого номера вы хотите добавить строки");

int[] elem = new int[K + 2];

elem[0] = K; elem[1] = N; int z = 2;

int[][] arr1 = new int[arr3.Length + K][];

int k = 0;

int skik = 0;

for (int i = 0; i < N - 1; i++)

{

skik += 1;

arr1[i] = new int[arr3[i].Length];

arr3[i].CopyTo(arr1[i], 0);

}

int[] arr2 = null;

for (int i = N - 1; i < N - 1 + K; i++)

{

k++;

if (k%10 == 3 && k % 100 != 13)

arr2 = AddWithSpace($"\nВводите элементы {k}-ей строки через пробел");

else

arr2 = AddWithSpace($"\nВводите элементы {k}-ой строки через пробел");

elem[z] = arr2.Length;

z += 1;

arr1[i] = new int[arr2.Length];

arr2.CopyTo(arr1[i], 0);

}

for (int i = N - 1 + k; i < arr1.Length; i++)

{

arr1[i] = new int[arr3[skik].Length];

arr3[skik].CopyTo(arr1[i], 0);

skik += 1;

}

arr3 = arr1;

return elem;

}

static string MenuStart()

{

Console.WriteLine("1. Работа с одномерными массивами");

Console.WriteLine("2. Работа с двумерными массивами");

Console.WriteLine("3. Работа с рваными массивами");

Console.WriteLine("0. Выход из программы");

string bug = Console.ReadLine();

while (bug != "0" && bug != "1" && bug != "2" && bug != "3")

{

Console.WriteLine("\nВведите целое число от 0 до 3!");

bug = Console.ReadLine();

}

return bug;

}

static string MenuOdnMass1()

{

Console.WriteLine("1. Сформировать одномерный массив с помощью датчика случайных чисел");

Console.WriteLine("2. Сформировать одномерный массив вручную");

Console.WriteLine("3. Печать массива");

Console.WriteLine("4. Добавить после каждого отрицательного элемента его модуль");

Console.WriteLine("0. Назад");

string bug = Console.ReadLine();

while (bug != "0" && bug != "1" && bug != "2" && bug != "3" && bug != "4")

{

Console.WriteLine("\nВведите целое число от 0 до 4!");

bug = Console.ReadLine();

}

return bug;

}

static string MenuOdnMass2()

{

Console.WriteLine("1. Сформировать двумерный массив с помощью датчика случайных чисел");

Console.WriteLine("2. Сформировать двумерный массив вручную");

Console.WriteLine("3. Печать массива");

Console.WriteLine("4. Удалить все четные столбцы");

Console.WriteLine("0. Назад");

string bug = Console.ReadLine();

while (bug != "0" && bug != "1" && bug != "2" && bug != "3" && bug != "4")

{

Console.WriteLine("\nВведите целое число от 0 до 4!");

bug = Console.ReadLine();

}

return bug;

}

static string MenuOdnMass3()

{

Console.WriteLine("1. Сформировать рваный массив с помощью датчика случайных чисел");

Console.WriteLine("2. Сформировать рваный массив вручную");

Console.WriteLine("3. Печать массива");

Console.WriteLine("4. Добавить K строк, начиная с номера N");

Console.WriteLine("0. Назад");

string bug = Console.ReadLine();

while (bug != "0" && bug != "1" && bug != "2" && bug != "3" && bug != "4")

{

Console.WriteLine("\nВведите целое число от 0 до 4!");

bug = Console.ReadLine();

}

return bug;

}

static void Main(string[] args)

{

bool massive1 = false; bool massive2 = false; bool massive3 = false;

int[] arr1 = null; int[,] arr2 = null; int[][] arr3 = null; int ll = 0;

do

{

string Menu = MenuStart();

switch (Menu)

{

case "0":

{

ll = -4;

Console.Clear();

Console.WriteLine("Благодарим за использование программы!\n");

break;

}

case "1":

{

int ll1 = 0;

do

{

Console.Clear();

string Menu1 = MenuOdnMass1();

switch (Menu1)

{

case "0":

{

ll1 = -4;

Console.Clear();

break;

}

case "1":

{

Console.Clear();

MakeRandomMas1(ref arr1);

massive1 = true;

break;

}

case "2":

{

Console.Clear();

arr1 = AddWithSpace(msg: "Вводите числа через пробел, после нажмите энтер");

while (arr1.Length == 0)

arr1 = AddWithSpace(msg: "Массив пустой. " +

"Дальнейшие действия не имеют смысла, сформируйте массив заново");

Console.WriteLine("\nМассив сформирован");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

massive1 = true;

break;

}

case "3":

{

Console.Clear();

if (massive1)

PrintMas(arr1, msg: "Массив представлен ниже:");

else

{

Console.WriteLine("Сначало нужно сформировать массив!");

}

Console.WriteLine("Нажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

break;

}

case "4":

{

Console.Clear();

if (massive1)

{

int[] kok = arr1;

AbsFunction1(ref arr1);

PrintMas(arr1, msg: "Полученный массив представлен ниже:");

PrintMas(kok, msg: "Исходный массив:");

}

else

Console.WriteLine("Сначало нужно сформировать массив");

Console.WriteLine("Нажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

break;

}

}

} while (ll1 != -4);

break;

}

case "2":

{

int ll1 = 0;

do

{

Console.Clear();

string Menu1 = MenuOdnMass2();

switch (Menu1)

{

case "0":

{

ll1 = -4;

Console.Clear();

break;

}

case "1":

{

Console.Clear();

MakeRandomMas2(ref arr2);

massive2 = true;

break;

}

case "2":

{

Console.Clear();

MakeManualMas2(ref arr2);

massive2 = true;

break;

}

case "3":

{

Console.Clear();

if (massive2)

PrintMas(arr2, msg: "Массив представлен ниже:");

else

{

Console.WriteLine("Сначало нужно сформировать массив!");

}

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

break;

}

case "4":

{

Console.Clear();

if (massive2)

{

int[,] kok = arr2;

DelChetColumns2(ref arr2);

PrintMas(arr2, msg: "Полученный массив представлен ниже:");

PrintMas(kok, msg: "\nИсходный массив:");

}

else

Console.WriteLine("Сначало нужно сформировать массив");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

break;

}

}

} while (ll1 != -4);

break;

}

case "3":

{

int ll1 = 0;

do

{

Console.Clear();

string Menu1 = MenuOdnMass3();

switch (Menu1)

{

case "0":

{

ll1 = -4;

Console.Clear();

break;

}

case "1":

{

Console.Clear();

MakeRandomMas3(ref arr3);

massive3 = true;

break;

}

case "2":

{

Console.Clear();

MakeManualMas3(ref arr3);

massive3 = true;

break;

}

case "3":

{

Console.Clear();

if (massive3)

PrintMas(arr3, msg: "Массив представлен ниже:");

else

{

Console.WriteLine("Сначало нужно сформировать массив!");

}

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

break;

}

case "4":

{

Console.Clear();

if (massive3)

{

Console.WriteLine("1. Добавить К строк, начиная с номера N с помощью датчика случайных чисел");

Console.WriteLine("2. Добавить K строк, начиная с номера N вручную");

string bug = Console.ReadLine();

if (bug == "1")

{

int[][] kok\_rv = arr3;

Console.Clear();

int[] elem = AddStringsRandomly3(ref arr3);

Console.Clear();

if (elem[0] == 0)

{

PrintMas(arr3, msg: "Полученный массив представлен ниже:");

PrintMas(kok\_rv, msg: "\nИсходный массив:");

}

else

{

Console.WriteLine($"Условие: добавить {elem[0]} строк, начиная с номера {elem[1]}");

for (int i = 2; i < elem.Length; i++)

{

Console.WriteLine($"Количество элементов первой строки – {elem[i]}");

}

PrintMas(arr3, msg: "\nПолученный массив представлен ниже:");

PrintMas(kok\_rv, msg: "\nИсходный массив:");

}

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

else if (bug == "2")

{

int[][] kok\_rv = arr3;

Console.Clear();

int[] elem = AddStringsManually3(ref arr3);

Console.Clear();

if (elem[0] == 0)

{

PrintMas(arr3, msg: "Полученный массив представлен ниже:");

PrintMas(kok\_rv, msg: "\nИсходный массив:");

}

else

{

Console.WriteLine($"Условие: добавить {elem[0]} строк, начиная с номера {elem[1]}");

for (int i = 2; i < elem.Length; i++)

{

Console.WriteLine($"Количество элементов первой строки – {elem[i]}");

}

PrintMas(arr3, msg: "\nПолученный массив представлен ниже:");

PrintMas(kok\_rv, msg: "\nИсходный массив:");

}

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

else

{

while (bug != "1" && bug != "2")

{

Console.WriteLine("\nВведите либо 1, либо 2!");

bug = Console.ReadLine();

}

}

}

else

{

Console.WriteLine("Сначало нужно сформировать массив");

Console.WriteLine("\nНажмите любую клавишу, чтобы продолжить");

Console.ReadKey();

Console.Clear();

}

break;

}

}

} while (ll1 != -4);

break;

}

}

} while (ll != -4);

}

}

}