## Лабораторная работа № 6.

При решении задач из этой лабораторной работы опирайтесь на материал лекций. Все программы должны содержать необходимые обработчики исключительных ситуаций.

## Массивы

В задачах необходимо реализовать отдельные методы для формирования массива, вывода его на экран, изменения и т. п. По возможности используйте цикл *foreach*.

**Часть *A*.** Реализуйте указанные задачи в консольных приложениях.

* 1. В одномерном массиве 10 целых чисел, элементы которого считываются с клавиатуры, поменяйте порядок элементов массива, расположенных между первым максимальным и последним минимальным элементами, на обратный.
  2. В консольном приложении за минимальное число действий реализуйте указанную задачу, используя одномерный массив 15 целых чисел. Массив инициализируется константами. Даны два неубывающих массива. Найдите их «пересечение», т. е. массив, содержащий общие элементы исходных массивов, причем кратность каждого элемента в итоговом массиве равняется минимуму из его кратностей в исходных массивах.
  3. В консольном приложении реализуйте указанную задачу, используя одномерный массив целых чисел. Размерность исходного массива считывается с клавиатуры с проверкой корректности ввода, элементы являются случайными числами из диапазона [–15…30]. Операции удаления и вставки элементов осуществить путем сдвига соответствующих элементов массива. Решение задачи путем вывода нескольких фрагментов исходного массива считается неверным. Программа должна содержать метод по выводу первых n элементов массива. Формулировка задачи: вставьте элемент, равный минимальному элементу всего массива, перед всеми элементами, равными максимальному элементу. Распечатайте получившийся массив. После этого удалите все повторяющиеся элементы, оставив только их первые вхождения, то есть получите массив различных элементов.
  4. Заполните массив следующим образом. Размерность массива определяется константами.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 3 | 4 | 10 | 11 | 21 |
| 2 | 5 | 9 | 12 | 20 | 22 |
| 6 | 8 | 13 | 19 | 23 | 30 |
| 7 | 14 | 18 | 24 | 29 | 31 |
| 15 | 17 | 25 | 28 | 32 | 35 |
| 16 | 26 | 27 | 33 | 34 | 36 |

После заполнения замените все четные элементы на сумму их соседей (соседними являются элементы, расположенные на одну строку выше или ниже в том же столбце или на один столбец левее или правее в той же строке; в зависимости от положения элемента количество его соседей может быть два, три или четыре).

**Часть *B*.** В консольном приложении реализуйте указанную задачу, используя двумерный массив целых чисел. Размерность массива считывается с клавиатуры, элементы – случайные числа из диапазона [–50…150].

* 1. Определите, является ли массив логическим квадратом (суммы по всем горизонталям, вертикалям и двум диагоналям равны). Если не является, то отсортируйте столбцы массива по неубыванию минимальных элементов в них (можно сформировать промежуточный одномерный массив минимумов).
  2. Определите, есть ли в данном массиве столбец, содержащий ровно два отрицательных элемента. Если есть, то переставьте этот столбец после последнего столбца (количество столбцов остается неизменным).
  3. Удалите строку и столбец, на пересечении которых находится минимальный элемент массива. Вставьте строку и столбец из нулей между средними строками и столбцами получившегося массива (если их четное количество). Операции удаления и вставки строк/столбцов осуществить путем сдвига соответствующих элементов двумерного массива. Решение задачи путем вывода нескольких фрагментов исходного массива считается неверным. Программа должна содержать метод по выводу элементов массива с индекса [0, 0] до индекса [*n*, *m*].

***Баллы***: Задачи 1, 2 и 6 оцениваются по одному баллу, задачи 3, 4, 5, 7 оцениваются по 2 балла.

## Строки.

При решении задач используйте типы *char*, *char*[], *string*, *StringBuilder*. При работе со строками руководствуйтесь правилом, что, если необходимо разбивать строку на слова, лучше использовать класс *String*, а если в строке должны производиться изменения, то предпочтительнее класс *StringBuilder*. При решении некоторых задач потребуется использовать оба названных класса.

**Часть *А*.** Разработайте консольные приложения, решающие поставленные задачи для введенных пользователем строк.

* 1. Определите, какой из двух символов, введенных с клавиатуры, встречается в строке чаще. Если они встречаются одинаковое число раз, то определите, сколько различных символов встречается в строке.
  2. Подсчитайте сумму чисел, встречающихся в строке. Символ ‘–’ перед числом считается знаком отрицательного числа. Все остальные символы (кроме цифр и минуса) считаются разделителями между числами.

**Часть *B*.** Разработайте приложения *Windows Forms*, решающие поставленные задачи для вводимых пользователем в компонент TextBox строк.

* 1. Сформируйте новую строку, отличающуюся от исходной тем, что группы идущих подряд одинаковых символов разделены символом ‘\*’. Пример: “aabecccdaaa” → “aa\*b\*e\*ccc\*d\*aaa”.
  2. Удалите из строки последовательности символов, расположенные между круглыми скобками вместе со скобками (считается, что скобки в строке расставлены без ошибок).

**Часть *C*.** Разработайте консольное приложение, решающее поставленную задачу для введенных пользователем строк, представляющих собой осмысленное текстовое сообщение, слова которого разделяются пробелами и знаками препинания.

* 1. Удалите из сообщения все слова, содержащие введенный пользователем символ без учета регистра. Выведите полученную строку. После этого выведите те слова предложения, которые встречаются в измененном сообщении не менее трех раз (выводить эти слова по убыванию их длин).

***Баллы***: Каждая задача из части «Строки» (8–12) оценивается в один балл.