# Практическое занятие № 7. Обработка массивов

**1 Цель занятия**

Получить практические навыки обработки информации в массивах.

**2 Перечень оборудования и программного обеспечения**

Персональный компьютер

Microsoft Office (Word)

Microsoft Visual Studio

**3 Краткие теоретические сведения**

**3.1** **Объявление массивов**

***Массив*** – множество однотипных элементов. Любой массив является производным от класса System.Array.

В отличие от других языков программирования, при объявлении массива в С# нельзя указать его размер, т.к. при объявлении не создается сам массив, а только ссылка на будущий массив. Поэтому после объявления необходима инициализация массива.

Объявление одномерного массива выглядит следующим образом:

<тип>[] <имя массива>;

Квадратные скобки приписаны не к имени переменной, а к типу. Они являются неотъемлемой частью определения класса, так что запись int[] следует понимать как класс ***одномерный массив с элементами типа***int.

В данном случае речь идет об отложенной инициализации. ***При объявлении с отложенной инициализацией сам массив не создается, а создается только ссылка на массив,*** имеющая неопределенное значение null. Поэтому ***пока массив не будет реально создан и его элементы инициализированы, использовать его в вычислениях нельзя***.

int[] a, b, c; //три массива с отложенной инициализацией

Часто при объявлении массива используется имя с инициализацией. В случае простых переменных, могут быть два варианта инициализации:

a) инициализация является явной и задается константным массивом:

double[] x= {5.5, 6.6, 7.7};

Синтаксически, элементы константного массива следует заключать в фигурные скобки.

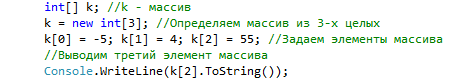
б) массив создается и инициализируется (выделяется место в памяти с указанным числом элементов массива). Массив из 5 элементов типа int:

int[] d= new int[5];

Доступ к отдельному элементу массива осуществляется посредством индекса. Индекс описывает позицию элемента внутри массива. В C# первый элемент имеет нулевой индекс.

Выход за границы массивов в С# расценивается как динамическая ошибка. Будет сгенерирована исключительная ситуация типа IndexOutOfRangeException, и программа прекратит выполнение.

**Пример 1:**



Здесь для создания массива используется строка Возможно объявление 

Элементы массива можно задавать сразу при объявлении:



Разумеется, приведенные конструкции применимы не только к типу int и не только к массиву размера 3.

В приведенном примере начальный элемент массива - это k[0], а последний - k[2].

При создании массива можно не указывать число элементов, которое хранит переменная, инициализируемая в результате работы программы.

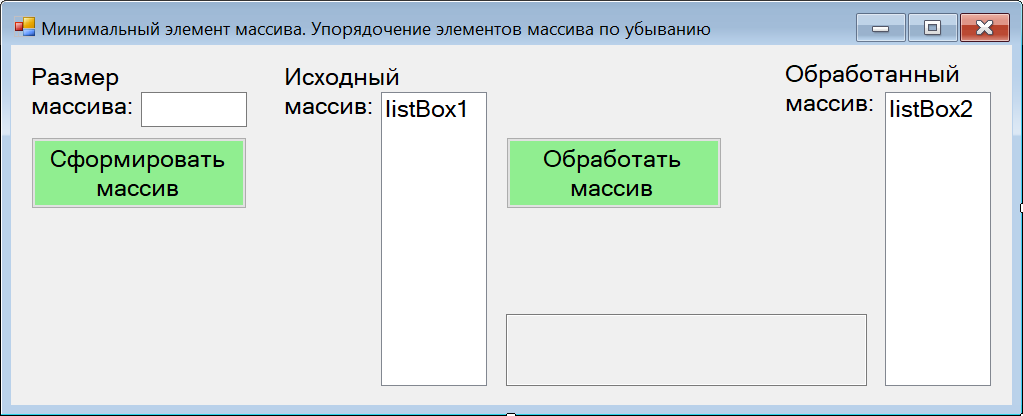
**Пример 2:**

|  |  |
| --- | --- |
| а) |  |
| б) |  |

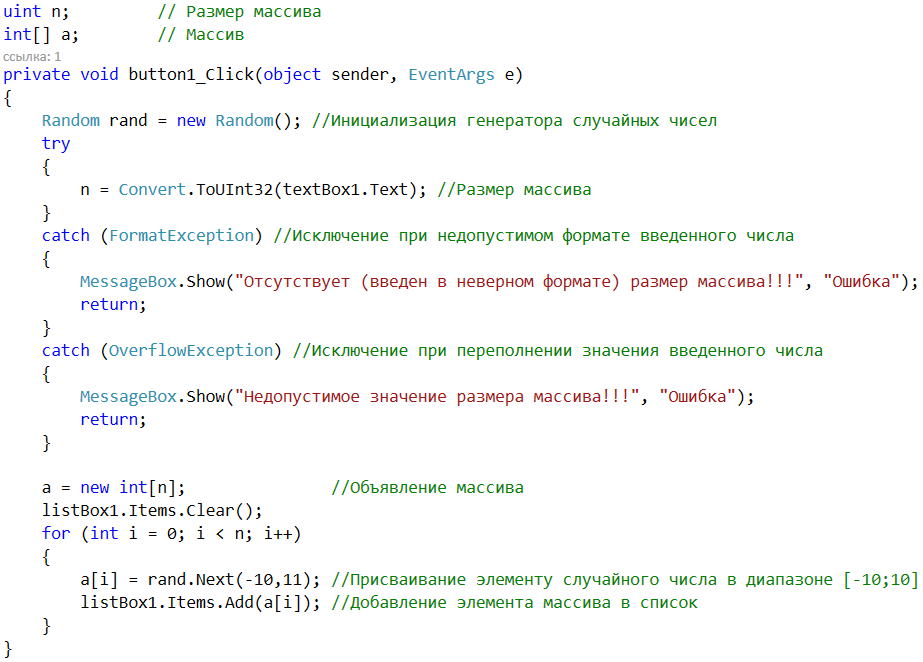
**Задание.** Дан массив размера *N*. Вывести минимальный элемент массива (значение и индекс), а затем упорядочить элементы массива в порядке убывания значений.

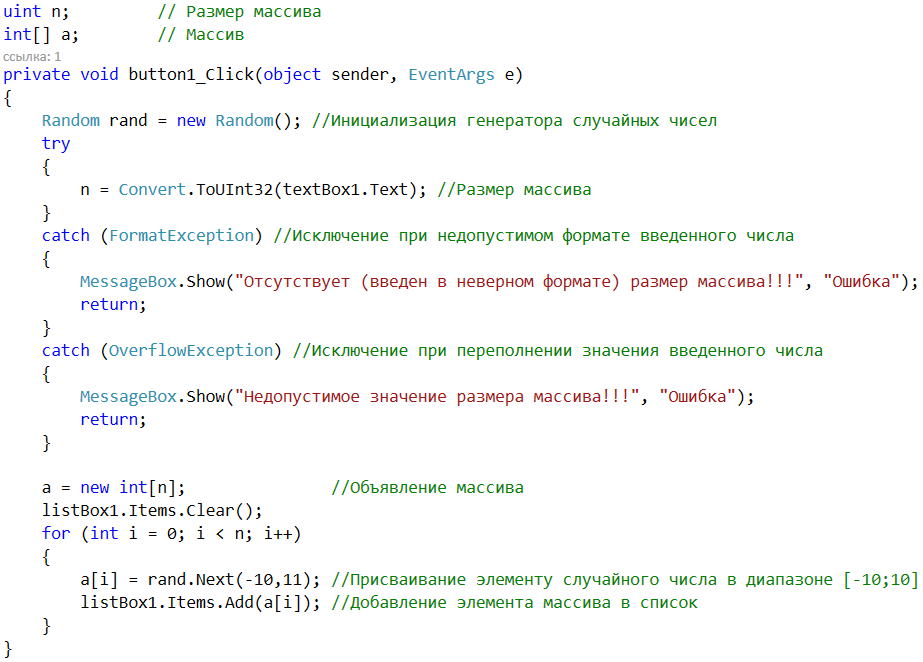
**Решение.**

а) Интерфейс приложения Windows Forms:

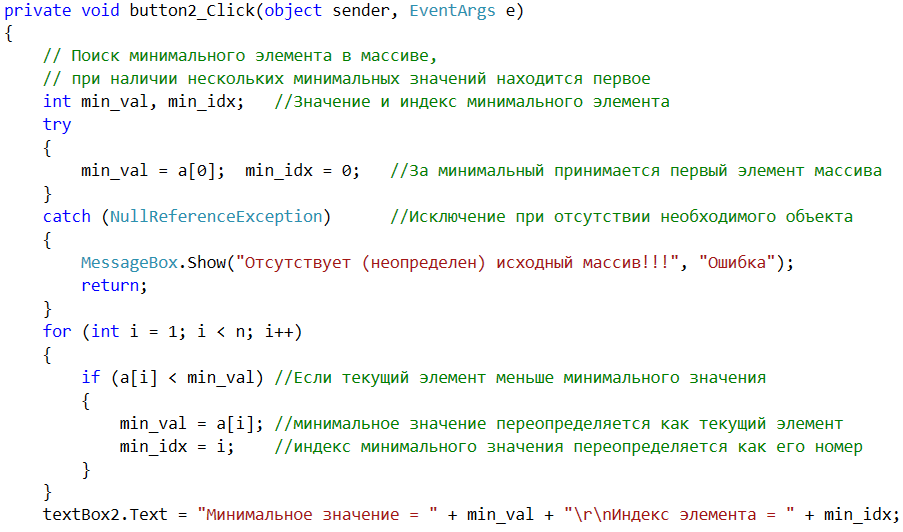


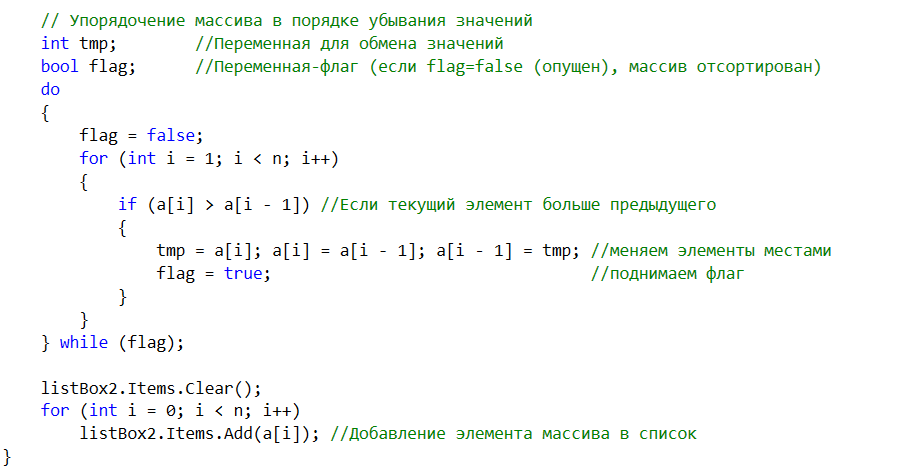
Код, выполняемый при щелчке на кнопку «Сформировать массив»:

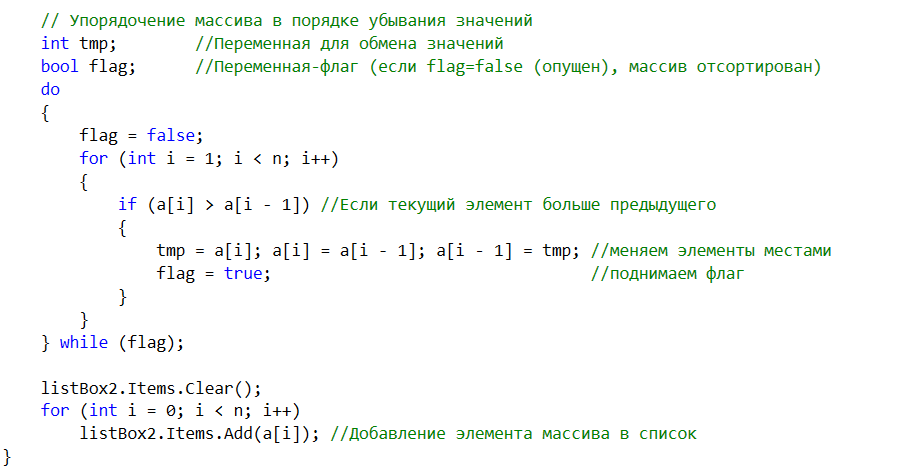




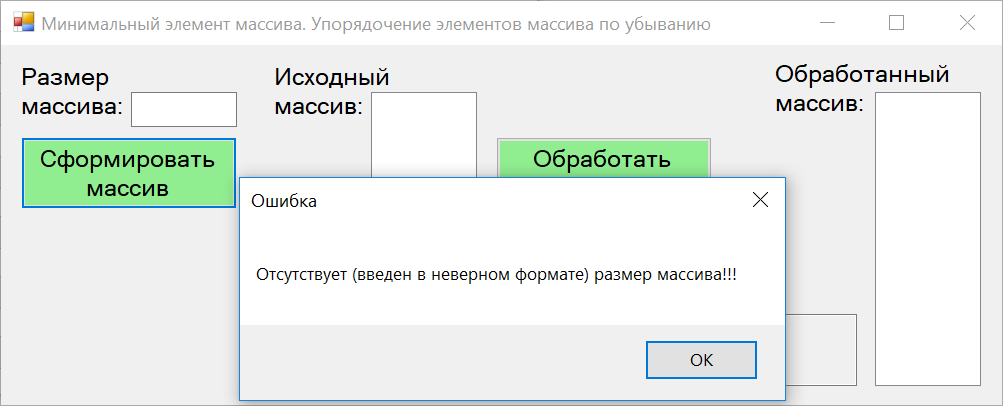
Код, выполняемый при щелчке на кнопку «Обработать массив»:

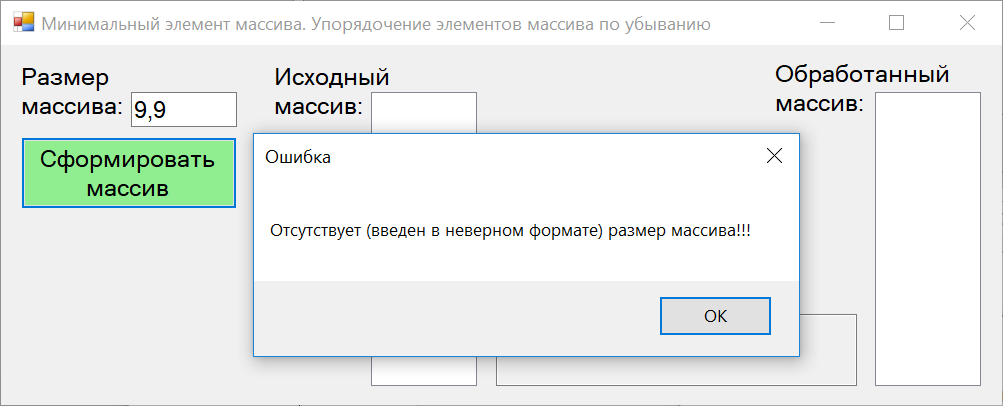


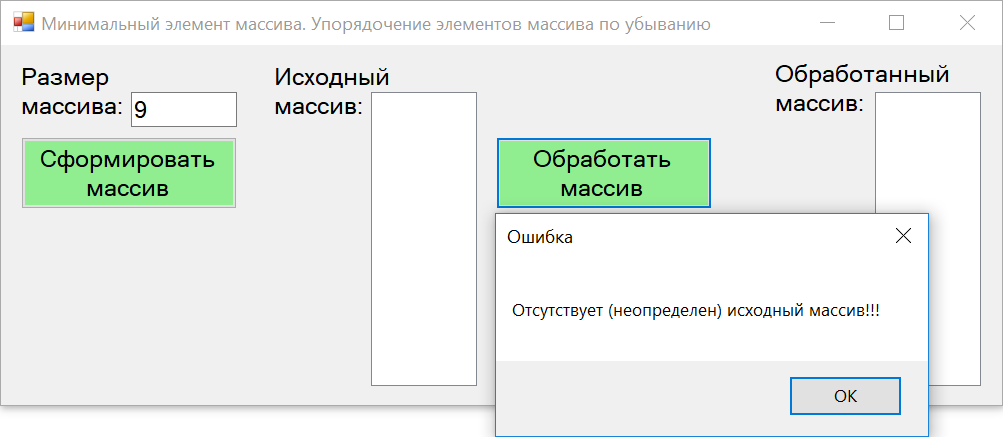


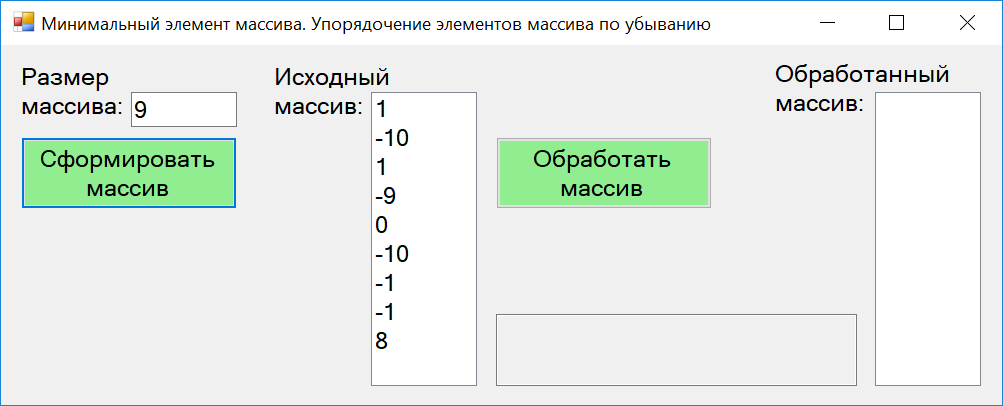


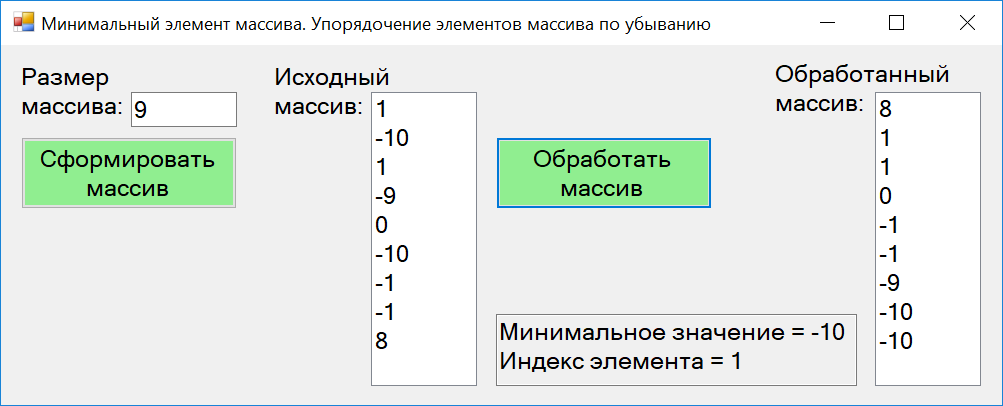
Скрины отлаженной программы:



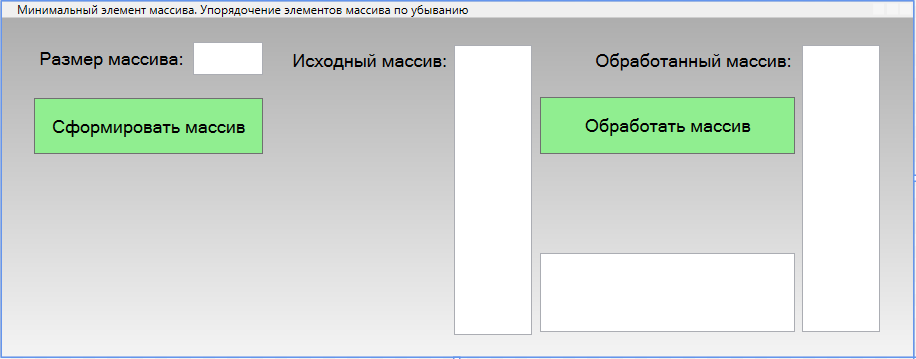




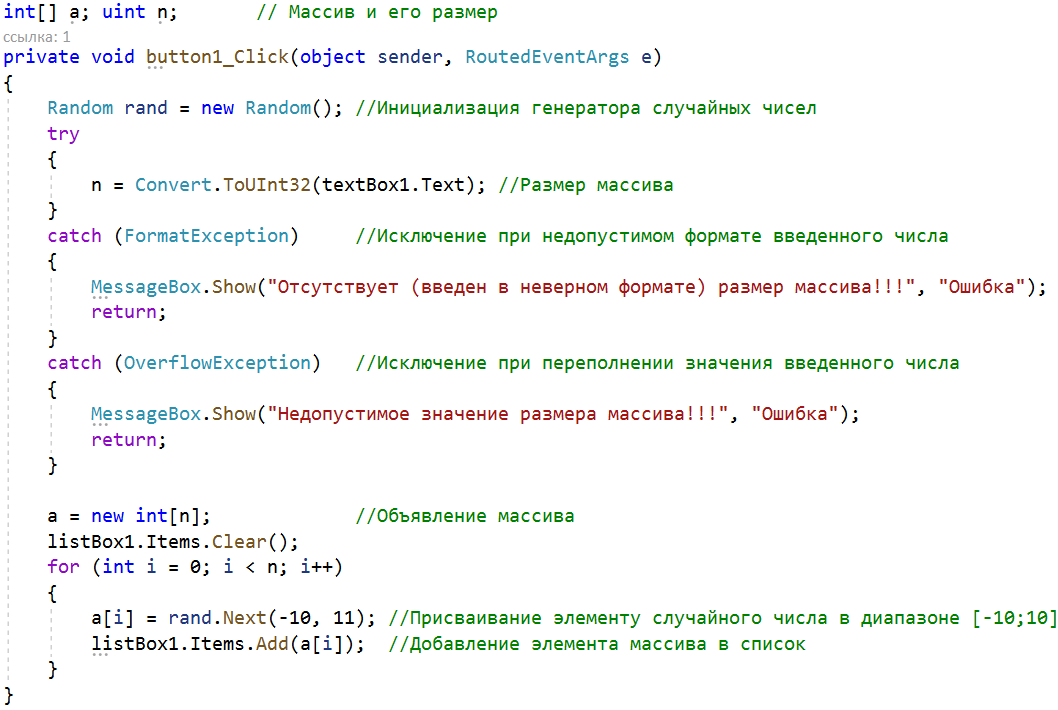




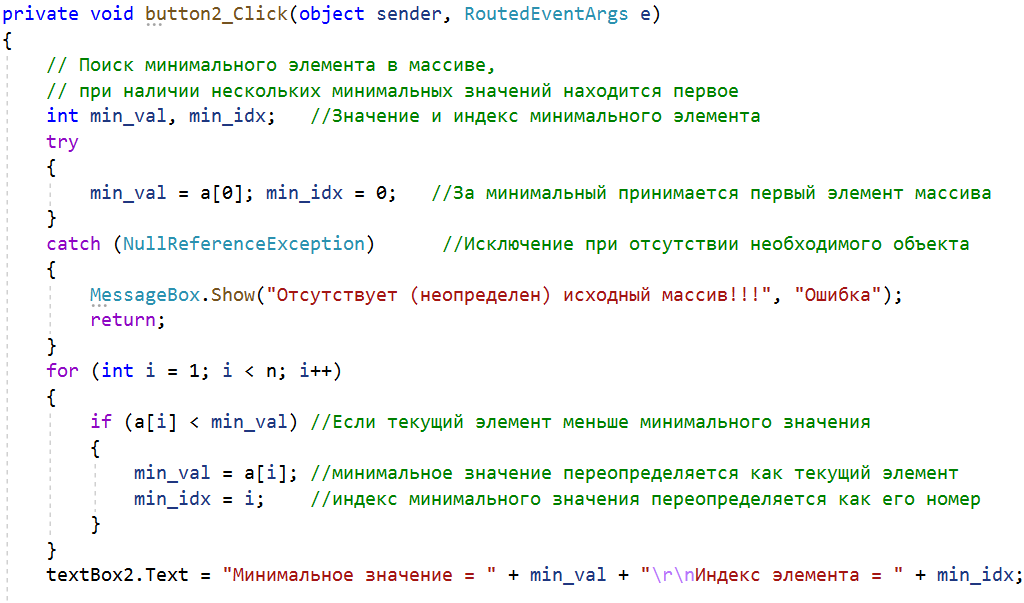
б) Интерфейс приложения WPF:

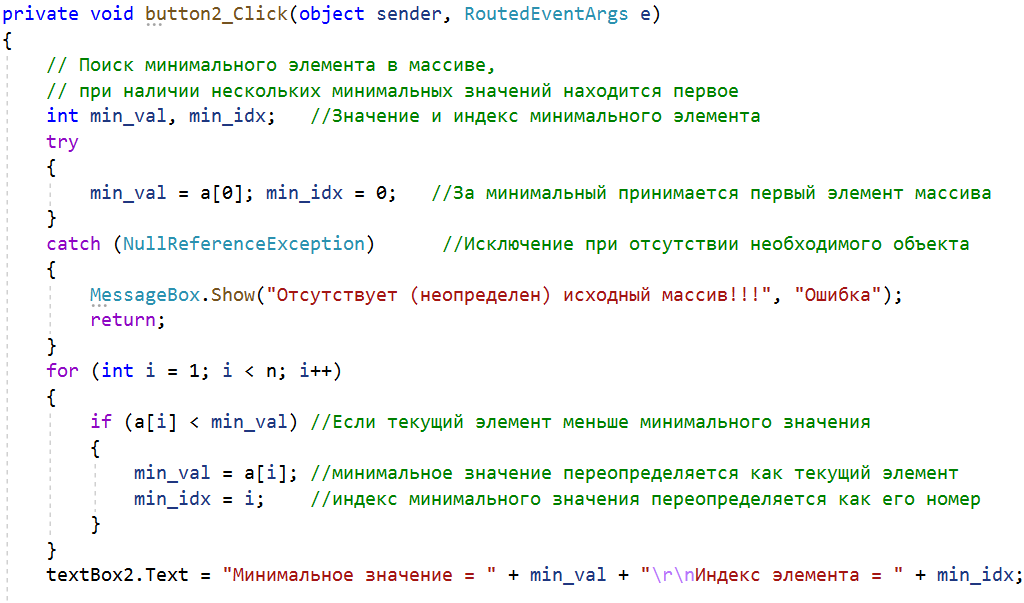


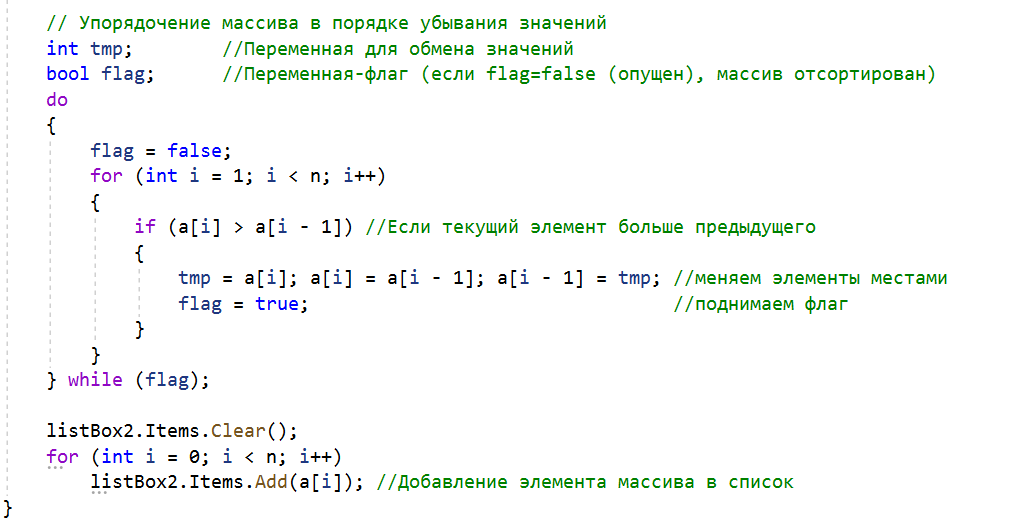
Код, выполняемый при щелчке на кнопку «Сформировать массив»:



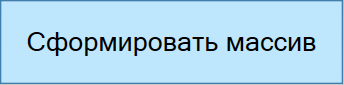
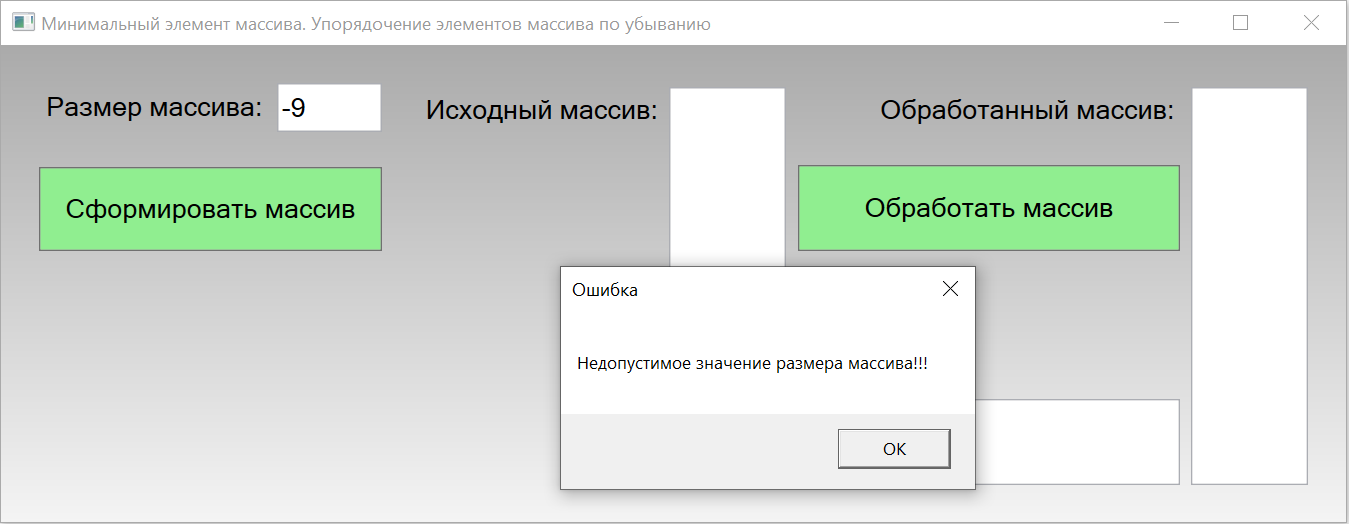
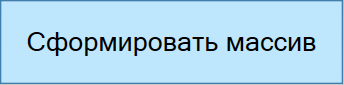
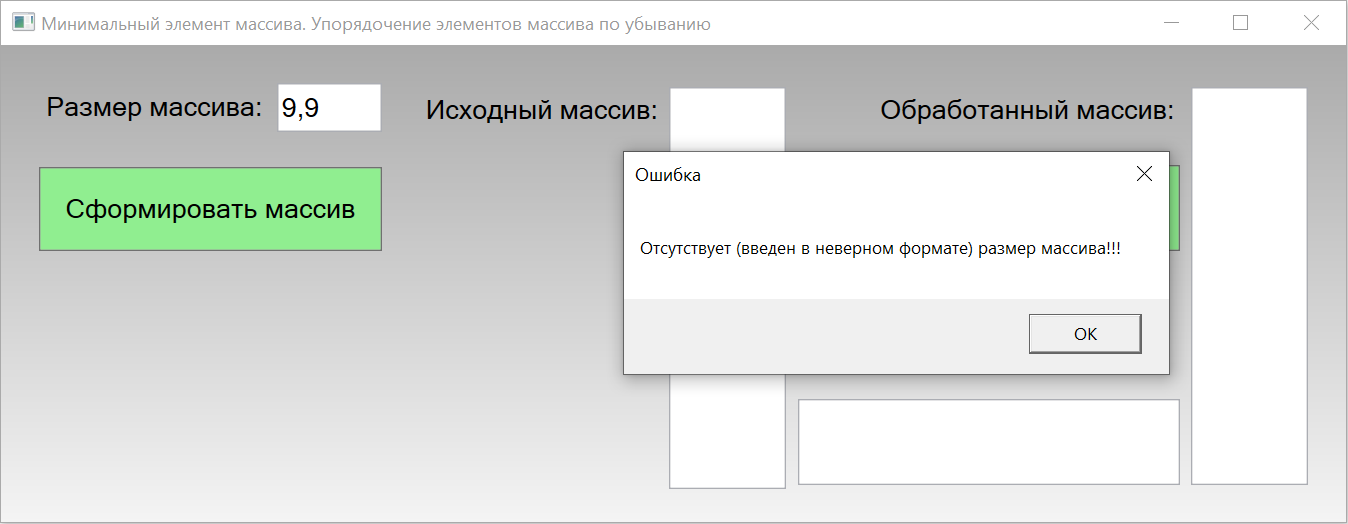
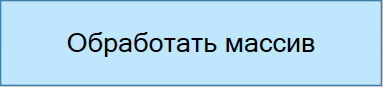
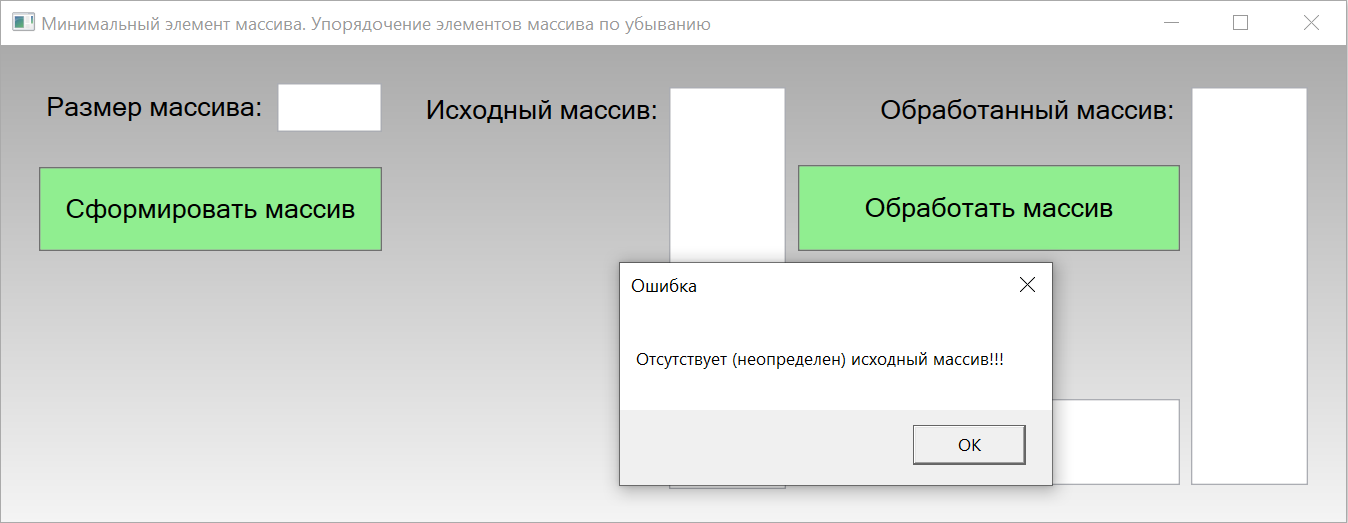
Код, выполняемый при щелчке на кнопку «Обработать массив»:

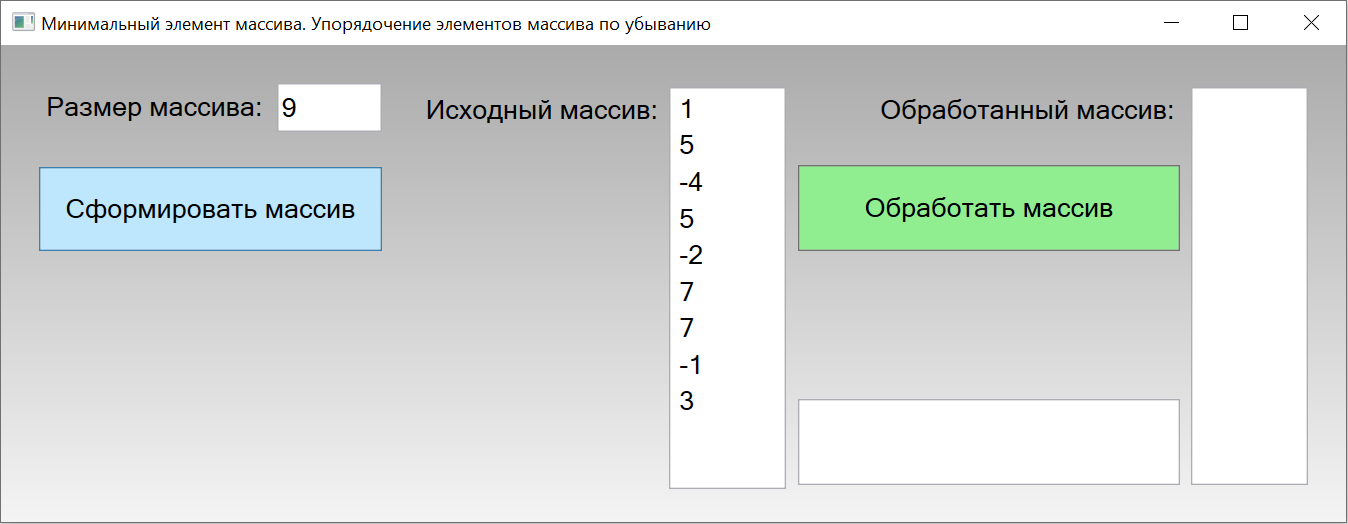


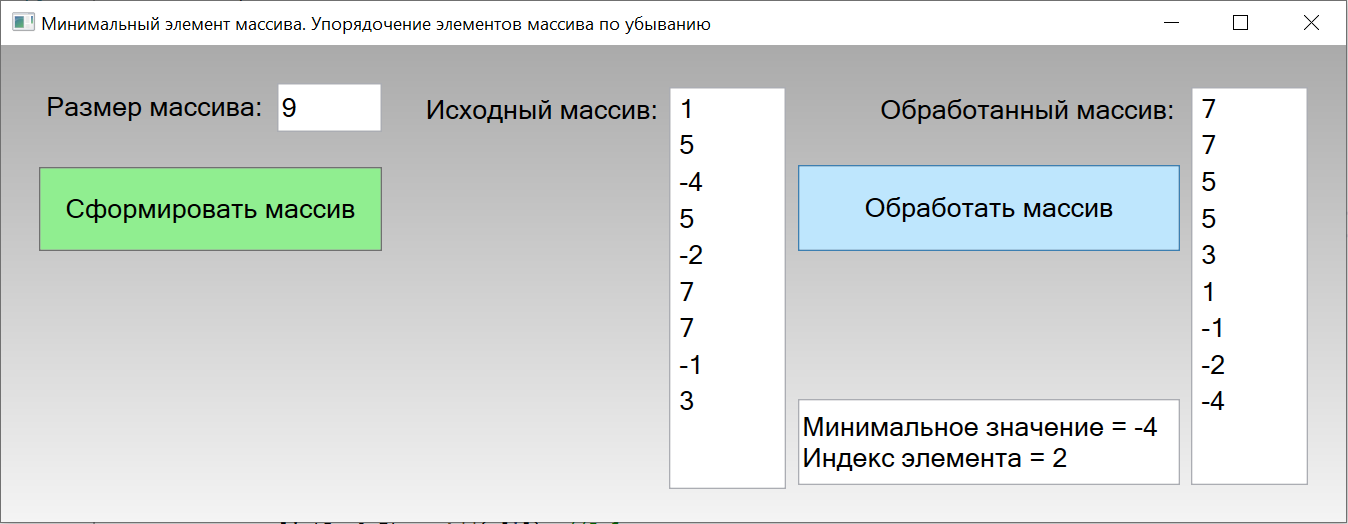




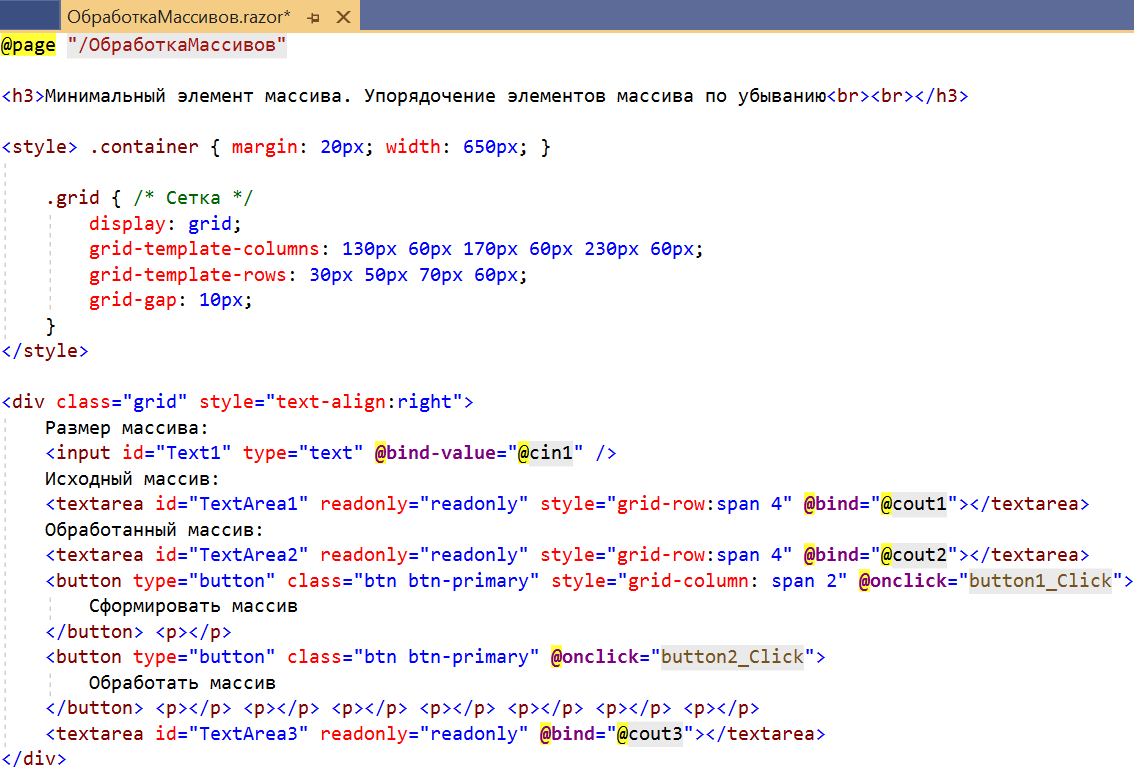
Скрины отлаженной программы:

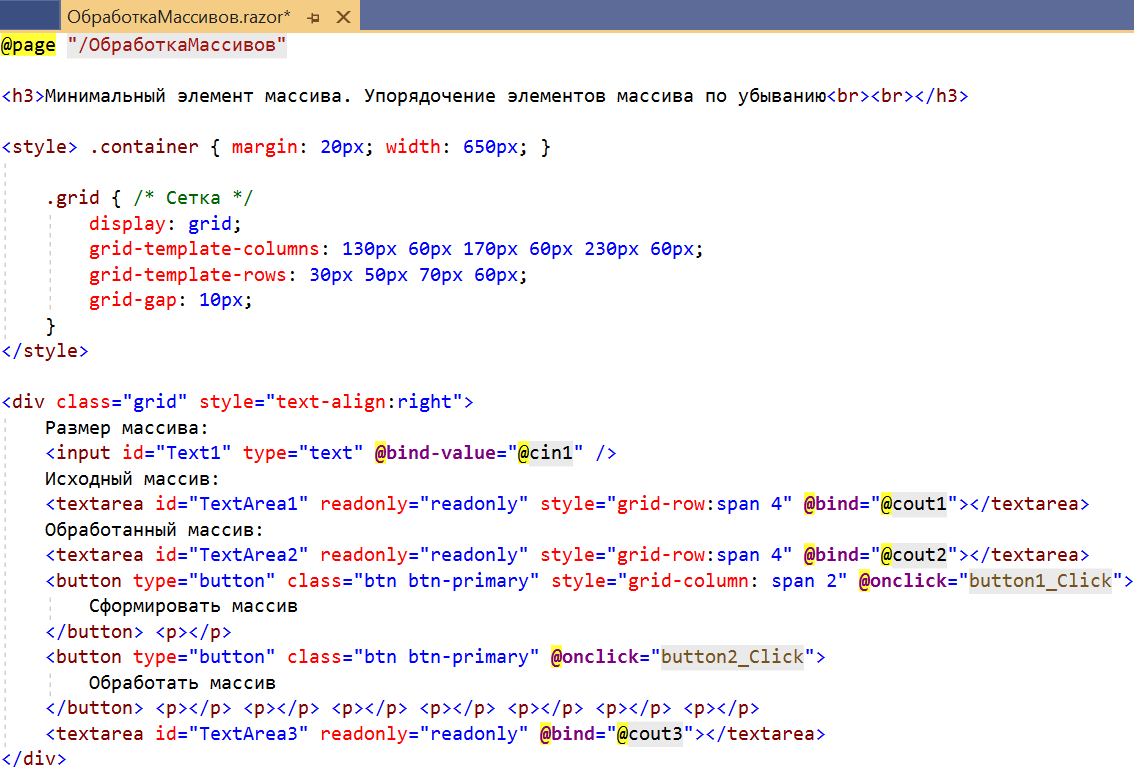


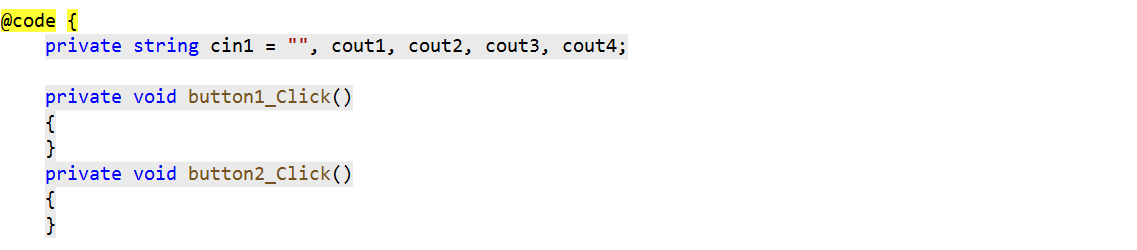


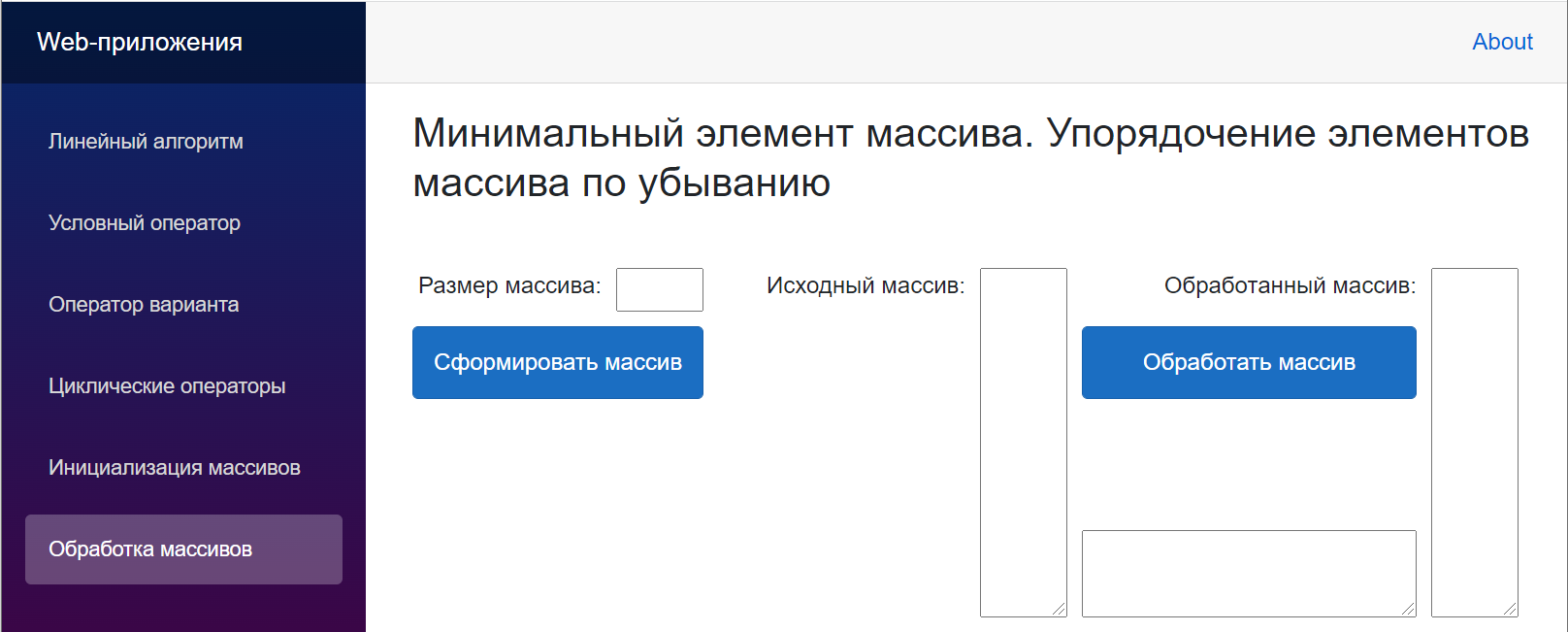


в) Интерфейс приложения WebAssembly Blazor:

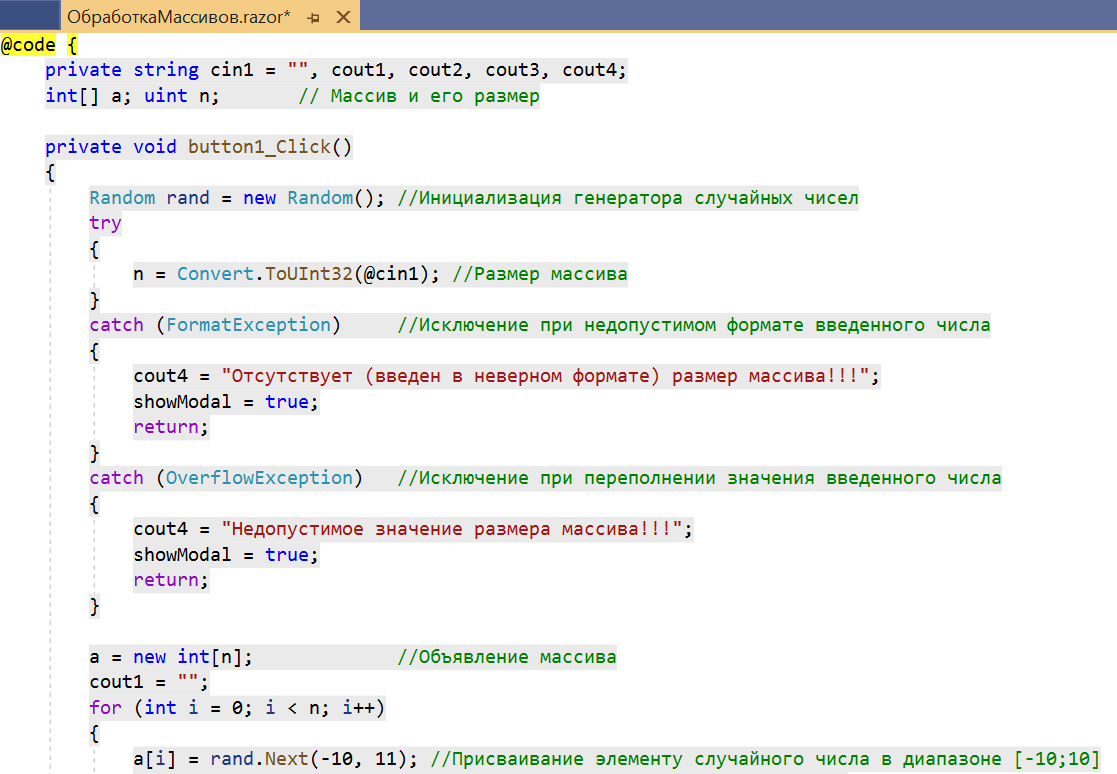


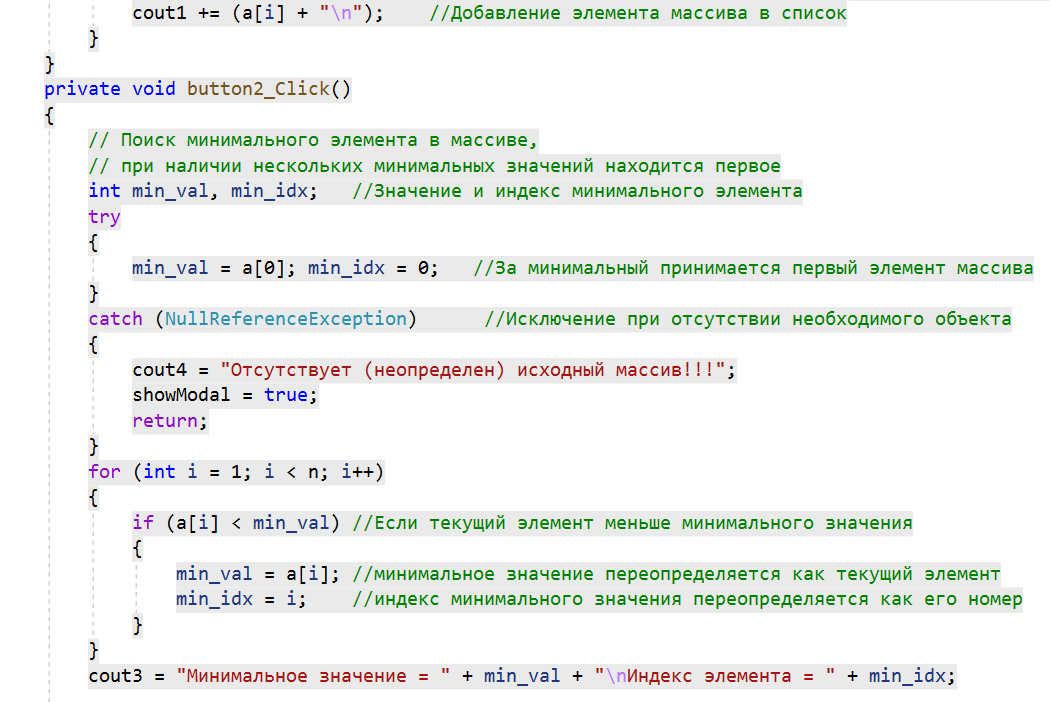


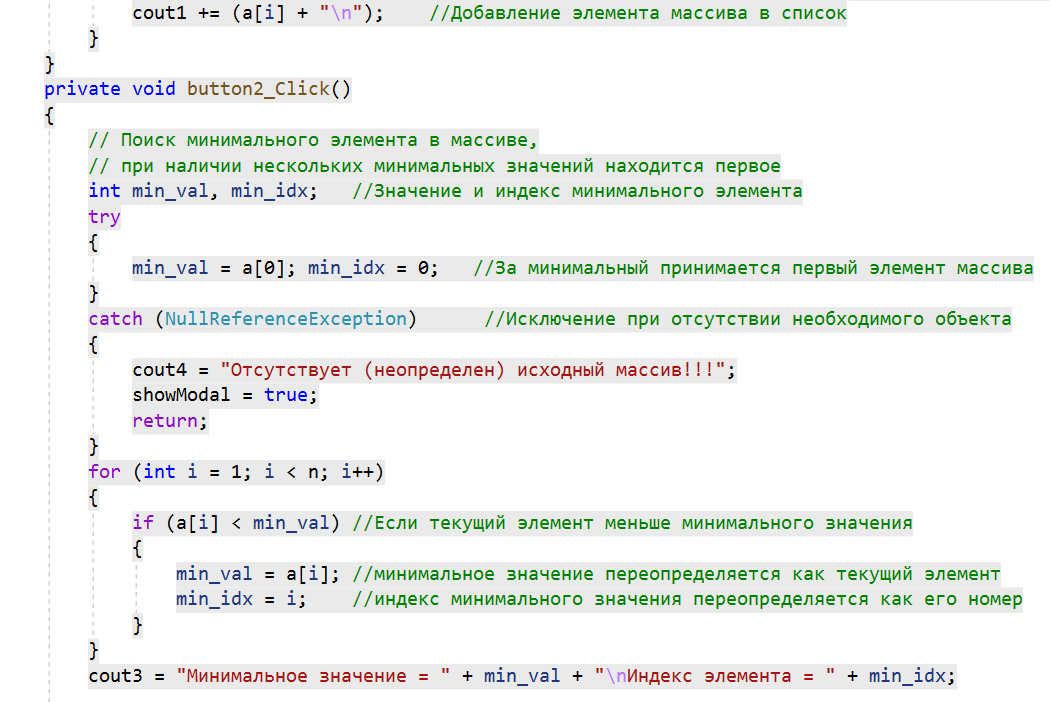


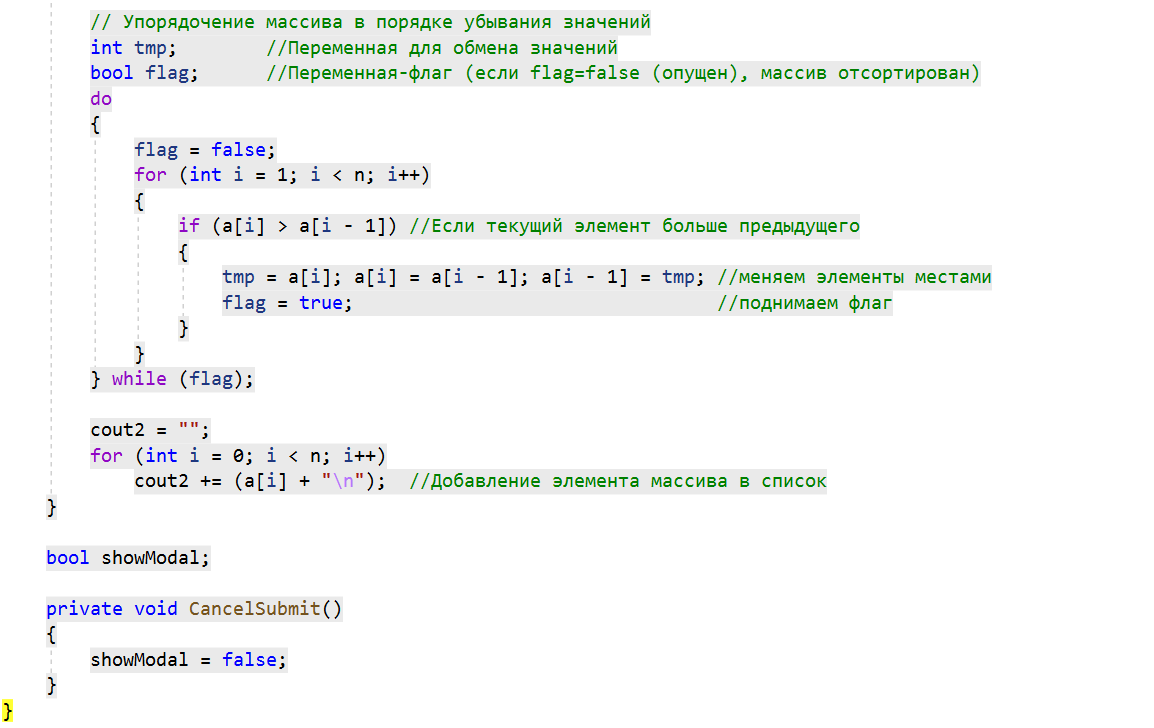


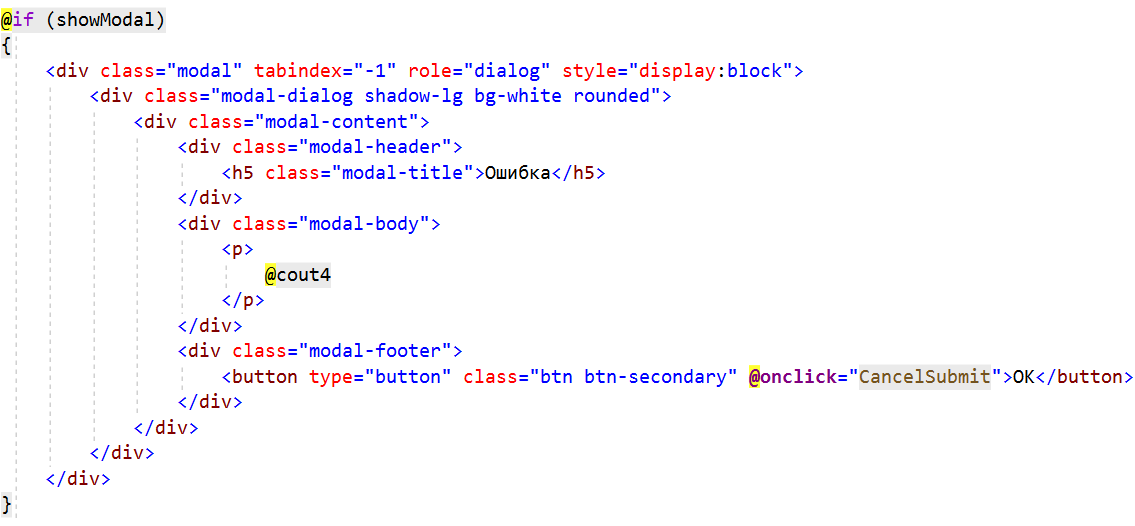
Коды приложения:



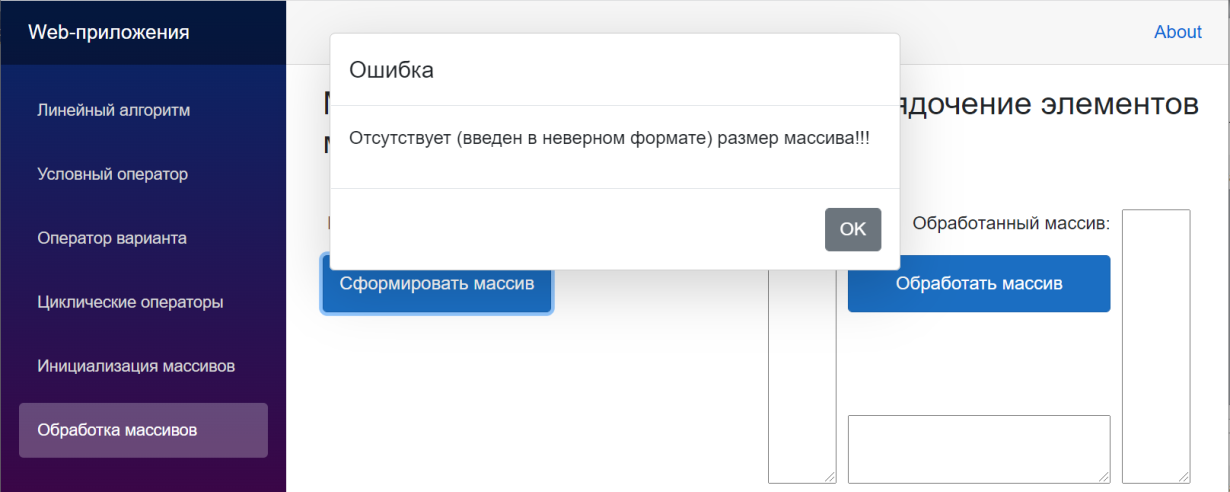


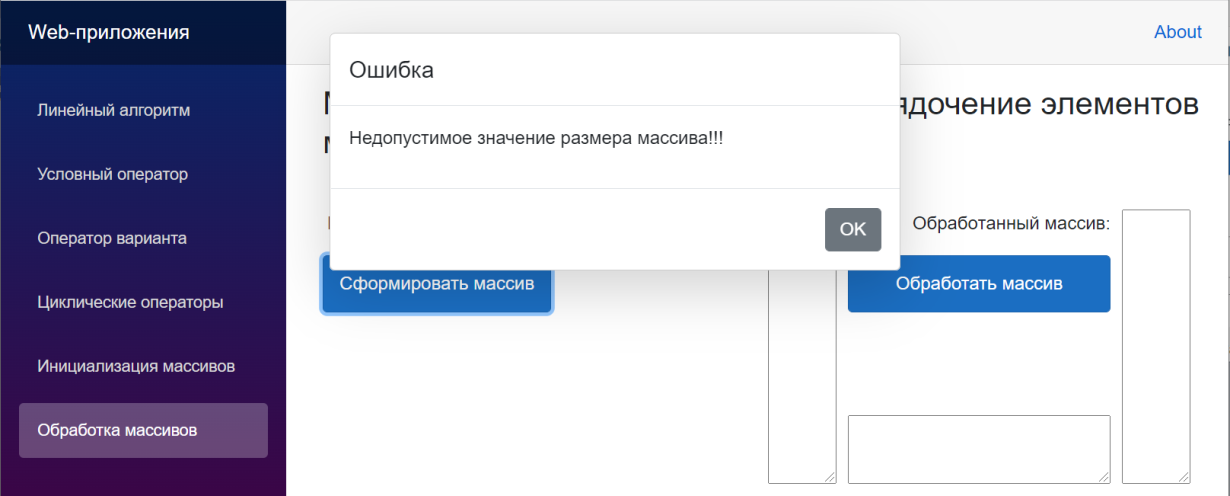


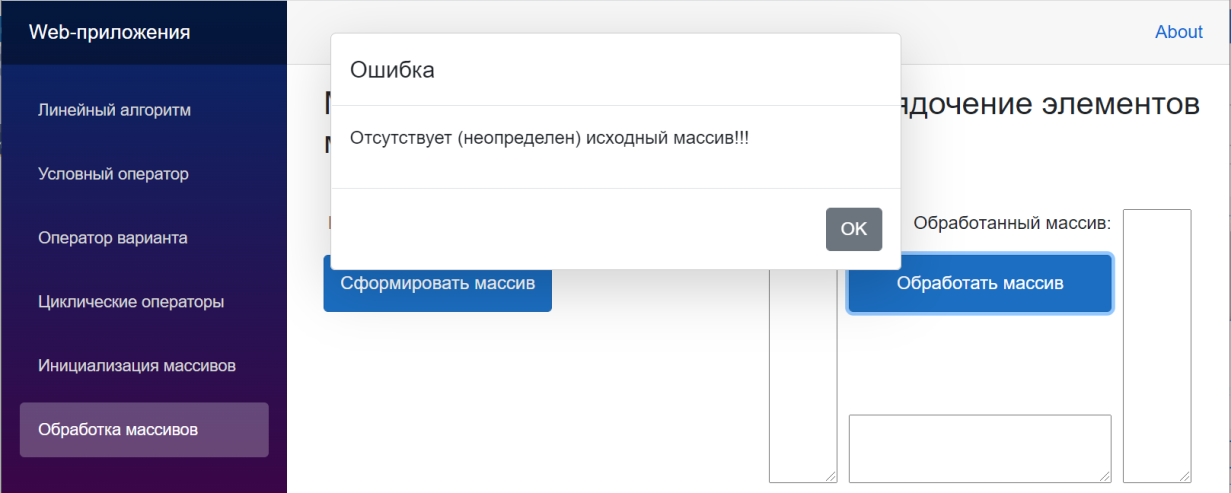


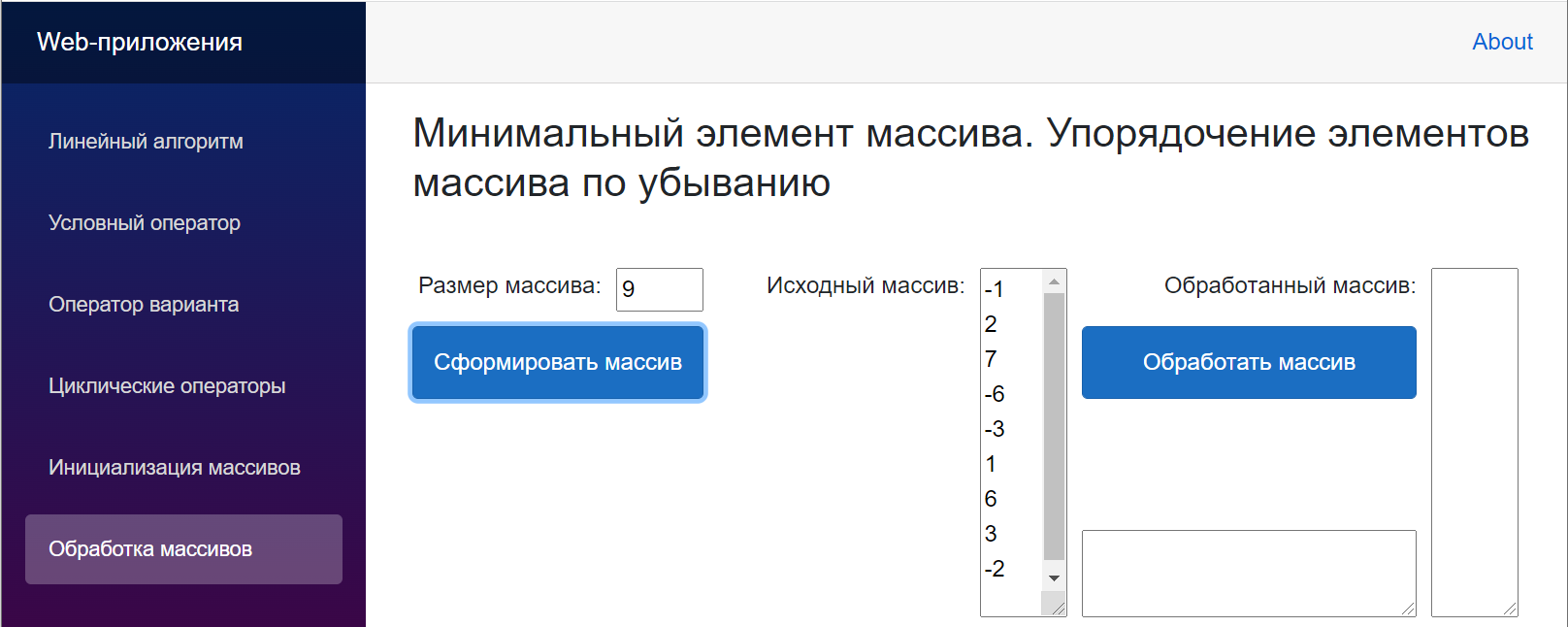


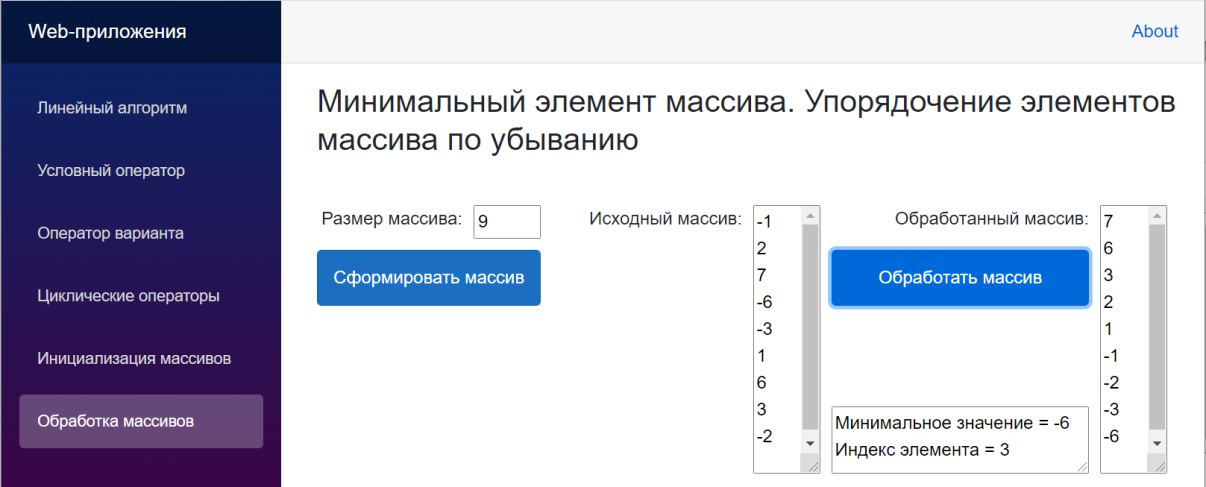
Скрины отлаженной программы:











**4 Порядок выполнения работы**

4.1 Изучить теоретические сведения и задание к работе.

4.2 В соответствии с вариантом задания создать форму приложения Windows Forms (WPF, WebAssembly Blazor).

4.3 В соответствии с вариантом задания разработать и отладить коды приложения Windows Forms (WPF, WebAssembly Blazor), используя инициализацию элементов массива случайными числами и исключения, выбрасываемые, если форматы или значения входных данных не соответствуют спецификациям параметров обрабатывающих их методов.

**5 Содержание отчета**

5.1 Название работы.

5.2 Цель работы.

5.3 Программные коды и скрины работы разработанного приложения Windows Forms в соответствии с вариантом задания.

**6 Варианты заданий**

6.1 Дан массив *A* размера *N*. Сформировать два новых массива *B* и *C*: в массив *B* записать все положительные элементы массива *A*, в массив *C* — все отрицательные (сохраняя исходный порядок следования элементов). Вывести вначале размер и содержимое массива *B*, а затем — размер и содержимое массива *C*.

6.2 Дан одномерный массив *A*, состоящий из *N* целых чисел. Переписать из него в одномерный массив *B* все нечетные числа. Если нечетных чисел в массиве *A* нет, то вывести на экран сообщение «Нет нечетных чисел», иначе вывести на экран массив *В*.

6.3 Дан целочисленный массив размера *N*. Увеличить все четные числа, содержащиеся в массиве, на исходное значение первого четного числа. Если четные числа в массиве отсутствуют, то оставить массив без изменений.

6.4 Пусть имеется два массива, заполненные целыми числами. Для двух массивов получите сумму попарных произведений их членов (скалярное произведение).

d(p,q)=\sqrt{(p_1-q_1)^2+(p_2-q_2)^2+\dots+(p_n-q_n)^2} = \sqrt{\sum_{k=1}^n (p_k-q_k)^2}6.5 Для двух массивов получите Евклидово расстояние между ними по формуле:

6.6 В массиве, элементами которого являются целые числа произвести следующие действия: сожмите массив, выбросив из него каждое второе значение. Оставшуюся половину массива заполните нулями.

6.7 Дан целочисленный массив размера *N*. Вывести все содержащиеся в данном массиве нечетные числа в порядке возрастания их индексов, а также их количество *K*.

6.8 Дан целочисленный массив размера *N*. Вывести все содержащиеся в данном массиве четные числа в порядке убывания их индексов, а также их количество *K*.

6.9 Дан целочисленный массив размера *N*. Вывести вначале все содержащиеся в данном массиве четные числа в порядке возрастания их индексов, а затем — все нечетные числа в порядке убывания их индексов.

6.10 Дан массив *A* размера *N* и целое число *K* (1 ≤ *K* ≤ *N*). Вывести элементы массива с порядковыми номерами, кратными *K*: *AK*, *A2·K*, *AN·K*, …

6.11 Дан массив размера *N*. Вывести его элементы в обратном порядке.

6.12 Дан массив *A* размера *N* (*N* — четное число). Вывести его элементы с четными номерами в порядке возрастания номеров: *A2*, *A4*, *A6*, …, *AN*.

6.13 Дан массив *A* размера *N* (*N* — нечетное число). Вывести его элементы с нечетными номерами в порядке убывания номеров: *AN*, *AN–2*, *AN–4*, …, *A1* .

6.14 Дан массив *A* размера *N*. Вывести вначале его элементы с четными номерами (в порядке возрастания номеров), а затем — элементы с нечетными номерами (также в порядке возрастания номеров): *A2*, *A4*, *A6*, …, *A1*, *A3*, *A5*, …

6.15 Дан массив *A* размера *N*. Вывести вначале его элементы с нечетными номерами в порядке возрастания номеров, а затем — элементы с четными номерами в порядке убывания номеров: *A1*, *A3*, *A5*, …, *A6*, *A4*, *A2*.

6.16 Дан массив *A* размера *N*. Вывести его элементы в следующем порядке: *A*1, *AN*, *A*2, *AN*−1, *A*3, *AN*−2, … .

6.17 Дан массив *A* размера *N*. Вывести его элементы в следующем порядке: *A*1, *A*2, *AN*, *AN*−1, *A*3, *A*4, *AN*−2, *AN*−3, … .

**7 Используемая литература**

7.1 Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул Технология разработки программного обеспечения. Форум, Инфра-М, 2009

7.2 Эндрю Троелсен Язык программирования С# 2010 и платформа .NET 4.0, М., ВИЛЬЯМС, 2011

7.3 А. Марченко C#. Введение в программирование, М, Вильямс, 2009

7.4 http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx.