

Лабораторная работа № 8

ПРИЛОЖЕНИЯ WINDOWS FORMS. ДИНАМИЧЕСКАЯ ГЕНЕРАЦИЯ ОКНА И ЕГО КОМПОНЕНТОВ. РАБОТА С ОДНОМЕРНЫМИ МАССИВАМИ.

Цель работы:

- ознакомиться с принципами динамической генерации компонентов Windows Forms;
- научиться применять основные принципы пользовательского интерфейса;
- получить навыки в организации различных типов табличных элементов.

Краткие теоретические сведения.

Генерация случайного числа

Для генерации случайных чисел C# имеет 3 перегруженных метода:

1. Возвращает значение большее нуля. Максимальное число больше 10 миллионов:

```
Random x = new Random(); // объявление переменной для генерации чисел
```

```
int n = x.Next();
```

2. Возвращает значение имеющееся в промежутке (min,max). При этом min значение входит в случайное число, а max не входит:

```
int n = x.Next(-100,100);
```

3. Возвращает целое положительное число не больше максимального:

```
int n = x.Next(10);
```

При этом максимальное число также не входит в генерацию случайных чисел, то есть в данном случае от 0 до 9.

Генерация дробных случайных чисел

В языке C# методов для генерации дробных чисел, но это не проблема, так как если нам надо получить дробные числа, мы можем сделать это следующим образом:

```
double r = Convert.ToDouble(x.Next(-100, 100)/10.0);
```

В данном случае мы имеем числа от -10 до 9.9.

Также само, к примеру иметь положительные дробные числа:

```
double r = Convert.ToDouble(x.Next(100)/10.0);
```

В данном случае мы имеем генерацию от 0 до 9.9.

Контейнеры в Windows Forms

Для организации элементов управления в связанные группы существуют специальные элементы - контейнеры. Например, Panel, FlowLayoutPanel, SplitContainer, GroupBox. Ту же форму также можно отнести к контейнерам. Использование контейнеров облегчает управление элементами, а также придает форме определенный визуальный стиль.

Все контейнеры имеют свойство Controls, которое содержит все элементы данного контейнера. Когда мы переносим какой-нибудь элемент с панели инструментов на контейнер, например, кнопку, она автоматически добавляется в данную коллекцию данного контейнера. Либо мы также можем добавить элемент управления динамически с помощью кода в эту же коллекцию.

Динамическое добавление элементов

Добавим на форму кнопку динамически. Для этого добавим событие загрузки формы, в котором будет создаваться новый элемент управления. Это можно сделать либо с помощью кода, либо визуальным образом.

С помощью перетаскивания элементов с Панели Инструментов мы можем легко добавить новые элементы на форму. Однако такой способ довольно ограничен, поскольку очень часто требуется динамически создавать (удалять) элементы на форме.

Для динамического добавления элементов создадим обработчик события загрузки формы в файле кода:

```
private void Form1_Load(object sender, EventArgs e)
{
}
```

Теперь добавим в него код добавления кнопки на форму:

```
private void Form1_Load(object sender, EventArgs e)
{
    Button helloButton = new Button();
    helloButton.BackColor = Color.LightGray;
    helloButton.ForeColor = Color.DarkGray;
    helloButton.Location = new Point(10, 10);
    helloButton.Text = "Привет";
    this.Controls.Add(helloButton);
}
```

Сначала мы создаем кнопку и устанавливаем ее свойства. Затем, используя метод `Controls.Add` мы добавляем ее в коллекцию элементов формы. Если бы мы это не сделали, мы бы кнопку не увидели, поскольку в этом случае для нашей формы ее просто не существовало бы.

С помощью метода `Controls.Remove()` можно удалить ранее добавленный элемент с формы:

```
this.Controls.Remove(helloButton);
```

Хотя в данном случае в качестве контейнера использовалась форма, но при добавлении и удалении элементов с любого другого контейнера, например, `GroupBox`, будет применяться все те же методы.

DataGrid. Некоторые полезные свойства.

RowBackground и AlternatingRowBackground	Устанавливают фон строки. Если установлены оба свойства, цветовой фон чередуется: RowBackground - для нечетных строк и AlternatingRowBackground - для четных
ColumnHeaderHeight	Устанавливает высоту строки названий столбцов.
ColumnWidth	Устанавливает ширину столбцов.
RowHeight	Устанавливает высоту строк.
GridLinesVisibility	Устанавливает видимость линий, разделяющих столбцы и строки. Имеет четыре значения - All - видны все линии, Horizontal - видны только горизонтальные линии, Vertical - видны только вертикальные линии, None - линии отсутствуют
HeadersVisibility	Задаёт видимость заголовков
HorizontalGridLinesBrush и VerticalGridLinesBrush	Задаёт цвет горизонтальных и вертикальных линий соответственно

Существует возможность определить свойства отображения столбцов `DataGrid`. Для этого надо воспользоваться свойством `DataGrid.Columns` и определить коллекцию столбцов для отображения в таблице.

Причем можно задать также и другой тип столбца, отличный от текстового. `DataGrid` поддерживает следующие варианты столбцов:

DataGridTextBoxColumn	Отображает элемент <code>TextBlock</code> или <code>TextBox</code> при редактировании
DataGridHyperlinkColumn	Представляет гиперссылку и позволяет переходить по указанному адресу
DataGridCheckBoxColumn	Отображает элемент <code>CheckBox</code>
DataGridComboBoxColumn	Отображает выпадающий список - элемент

	ComboBox
DataGridTemplateColumn	Позволяет задать специфичный шаблон для отображения столбца

Практическая часть.

Построить приложение Windows Forms. Первичный вид окна представляет собой лишь поле для ввода размерности одномерного массива (массивов) и кнопку построения шаблонов для их ввода. В качестве шаблонов использовать:

a) массив TextBox-ов

b) компонент DataGridView.

Предусмотреть заполнение массива случайными числами с возможностью изменения элементов вручную.

Поле для ввода размерности одномерного массива (массивов) и кнопка построения шаблонов должны быть заблокированы после создания шаблона. После нажатия кнопки выполнения задания в окне не должно быть возможности для редактирования каких-либо компонентов, но должна присутствовать кнопка сброса, приводящая окно к первоначальному виду.

Варианты	Задание 1	Задание 2
1.	1	1
2.	2	2
3.	3	3
4.	4	4
5.	5	5
6.	6	6
7.	7	7
8.	8	8
9.	1	9
10.	2	10
11.	3	1
12.	4	2
13.	5	3
14.	6	4
15.	7	5
16.	8	6

Задание 1. Работа с одномерными массивами

1. Пусть даны целые числа a_1, \dots, a_n . Получите новую последовательность из целых чисел, заменяя a_i нулями, если значение a_i не равно максимальному из a_1, \dots, a_n , заменяя a_i единицей - в противном случае ($i=1, \dots, n$).
2. Пусть дана последовательность из n различных целых чисел. Найдите среднее арифметическое чисел этой последовательности, расположенных между максимальным и минимальным числами (в сумму включить и оба этих числа).
3. Пусть даны две последовательности по n -целых чисел в каждой. Найдите наименьшее среди тех чисел первой последовательности, которые не входят во вторую последовательность, считая, что хотя бы одно такое число есть.
4. Замените каждый элемент массива средним арифметическим всех предшествующих ему элементов.
5. Пусть даны целые числа a_1, \dots, a_n . Если в данной последовательности ни одно четное число не расположено после нечетного, то получите все отрицательные члены последовательности, иначе - все положительные. Порядок следования чисел в обоих случаях замените обратным.
6. Пусть даны вещественные числа a_1, \dots, a_{2n} . Преобразуйте эту последовательность по правилу: большее из a_i и a_{n+i} ($i=1, \dots, n$) примите в качестве нового значения a_i , меньшее - в качестве нового значения a_{n+i} .
7. Пусть даны целые числа a_1, \dots, a_n каждое из которых отлично от нуля. Если в последовательности отрицательные и положительные члены чередуются, то ответом должна служить сама исходная последовательность. Иначе - получите все отрицательные члены последовательности, сохранив порядок их следования.
8. Пусть дана последовательность из k произвольных чисел. Составить новую последовательность, где каждой элемент есть остаток от деления на 5 соответствующего элемента предыдущей последовательности.

Задание 2

1. В одномерном массиве, состоящем из n вещественных элементов, вычислить: максимальный кратный 10 элемент массива и его номер; сумму элементов массива, расположенных до последнего положительного элемента. Сжать массив, удалив из него все элементы, модуль которых находится в интервале $[a, b]$. Новый массив не заводить.

2. В одномерном массиве, состоящем из n вещественных элементов, вычислить: второй (по величине) минимальный положительный нечетный элемент и его позицию; среднее арифметическое элементов массива, расположенных за последним максимальным по модулю элементом. Преобразовать массив таким образом, чтобы сначала располагались все элементы равные нулю, а потом – все остальные.
3. В одномерном массиве, состоящем из n вещественных элементов: определить каких элементов больше – положительных нечетных или отрицательных четных; определить, все ли элементы массива простые числа. Преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях.
4. В одномерном массиве, состоящем из n вещественных элементов: определить какие числа встречаются в нем более двух раз; вывести на экран элементы массива, большие среднего арифметического. Сжать массив, удалив из него все нули.
5. В одномерном массиве, состоящем из n вещественных элементов: определить есть ли в массиве хотя бы одно положительное меньшее минимального четного элемента; вычислить сумму элементов массива, расположенных между последним и предпоследним нечетными элементами. Преобразовать массив таким образом, чтобы в первой его половине располагались элементы по возрастанию, во второй – по убыванию.
6. В одномерном массиве, состоящем из n вещественных элементов: определить является ли массив палиндромом; вычислить сумму модулей элементов массива, расположенных после первого элемента, равного нулю. Преобразовать массив таким образом, чтобы сначала располагались элементы, кратные 5, затем 4, после 3. Остальные – в конце.
7. В одномерном массиве, состоящем из n вещественных элементов: положительные элементы уменьшить вдвое, а отрицательные заменить на значения их индексов; определить число соседств двух положительных чисел. Преобразовать массив следующим образом: удалить все нечетные элементы, остальные упорядочить массив по возрастанию.

8. В одномерном массиве, состоящем из n целых элементов: вычислить среднее геометрическое чисел кратных 5, но не кратных 4. Получите новый массив, в котором все элементы исходного массива встречаются только один раз. Осуществить циклический сдвиг элементов массива вправо на k позиций.
9. В одномерном массиве, состоящем из n целых элементов: определить, образует ли он возрастающую последовательность; вывести те элементы массива, индексы которых являются степенями двойки (1,2,4,8, 16,...) Преобразовать элементы массива следующим образом: i -й элемент после преобразования равен наибольшему среди первых i элементов.
10. В одномерном массиве, состоящем из n целых элементов: определить, расположены ли его четные элементы в порядке возрастания; определить число соседств простого и четного элементов массива. Преобразовать массив следующим образом: положительные кратные трем числа расположить в порядке возрастания в конце массива.

Контрольные вопросы

1. Как узнать размер массива в C#?
2. Как узнать количество элементов в массиве C#?
3. Как объявить массив в C#?
4. Как изменить размер массива C#?
5. Как создать массив?
6. Преимущества и недостатки массива?