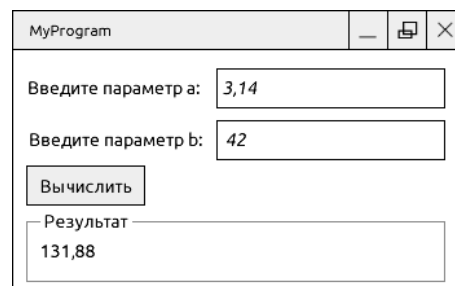


ДЕНЬ 3Е. Оконные приложения

3Е.1.Создать форму расчета площади участка

Разработать форму, которая содержит следующие элементы:

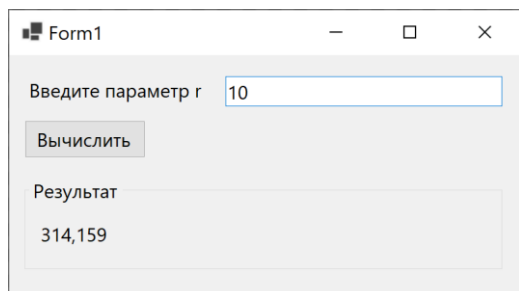
- Элементы ввода параметров (в зависимости от варианта, разное кол-во). Все параметры должны быть вещественными, принимать только положительные значения
- Кнопку «Вычислить»
- Блок вывода результата



При нажатии на кнопку «Вычислить» программа должна вычислить площадь участка согласно формуле, указанной в варианте, и вывести результат в блоке вывода результата (округлите до 3 чисел после запятой). Если пользователь ввел некорректные значения (например, текст в поля ввода или отрицательные значения) и нажал кнопку «Вычислить», текст ошибки должен отобразиться в поле вывода результата.

Пример

Формула площади: πr^2 , параметры: r



Варианты

№	Формула площади	Параметры
1	$a * a * \tan(p) / 2$	a, p
2	$a * b * \sin(c) / 2$	a, b, c
3	$a * h / 2$	a, h
4	$a * b$	a, b
5	$h * (a + b) / 2$	a, b, h
6	$r * (a + b + c) / 2$	a, b, c, r
7	$h * (a + b) / 2$	a, b, h
8	$a * h / 2$	a, h
9	$a * h / 2$	a, h
10	$a * b$	a, b
11	$a * h / 2$	a, h
12	$r * (a + b + c) / 2$	a, b, c, r
13	$a * b$	a, b
14	$a * b * \sin(c) / 2$	a, b, c
15	$a * b * \sin(c) / 2$	a, b, c
16	$a * a * \tan(p) / 2$	a, p

17	$a * b * \sin(c) / 2$	a, b, c
18	$a * a * \tan(p) / 2$	a, p
19	$h * (a + b) / 2$	a, b, h
20	$r * (a + b + c) / 2$	a, b, c, r

3Е.2. Создать форму генерации случайных строк

Создать форму содержащую:

- Радиокнопки выбора формата генерируемых строк (укажите в тексте кнопок содержимое формата)
- Поле ввода числа генерируемых строк (*NumericUpDown*, ввод только целых)
- Поле ввода имени файла
- Кнопку «Сохранить»

Пользователь выбирает один из трех форматов строк (по вариантам), указывает кол-во строк и имя файла, куда необходимо сохранить результат. При нажатии на кнопку «Сохранить» программа должна создать указанный файл и записать туда заданное кол-во случайно сгенерированных строк по выбранному формату.

Для случайного выбора используйте класс *Random*.

Формат описывается следующими спец. символами:

- A – любая гласная в верхнем регистре
- B – любая согласная в верхнем регистре
- X – любая буква в верхнем регистре
- a – любая гласная в нижнем регистре
- b – любая согласная в нижнем регистре
- x – любая буква в нижнем регистре
- 9 – любая цифра
- ? – предыдущий символ может быть опущен

Любые другие символы интерпретируются, как есть. Например, формат «Abc-999?», означает: сначала любая гласная в верхнем регистре, потом любая согласная в нижнем регистре, затем символ «с», затем символ «-», а далее две или три цифры: Arc-42

Пример

Форматы: 99, XX?X?, 009

99

XX?X?

009

Кол-во:

5

Файл:

result.txt

Сохранить

Выходной файл:

11
23
42
54
34

Варианты

№	Форматы		
1	A---B-?-?-?A---B	999?9?9?	Xaaaaaaaa
2	A9A-XX-ха?9	x?XB	9-9?-9?
3	M?aaxB	x9a9aX9a	+7 (123) 999-9999
4	AbB9	XXX AAb XBB	0?0?0?999
5	T?A9aBbA	b(BXX)Ab	bbb aaa BBB AAA
6	Xba	B9XABA?	123-999-567
7	(?3412)?999-999	AAAB?B?B?	xxxxxxx
8	9xxXxxB	x-B?A	999 999 999 999
9	R99999?9?	AxxxxxxxxB	999A?999
10	A9X	BX?bBAB	99?9?AA?A?BB?B?
11	aXB	A? AAb? A	ab?ab?ab?ab?
12	xABx9	baBB9(Xb)AAx	9-?9-?9
13	aB?bAbx?	99X9	b(a9a9a9a)b
14	9b?aB9aBAx	A9?b9bAaX	9A 9B 9x
15	Bab?aba?	999-999	XxxxxX
16	9X(AAB)XA	BxA	abab 999
17	8(999)999	XxXxXxX	9?9?9?AABB
18	AX[AA9X]Bx	B99-b?xb	XXXxxx999
19	Aba:bXX:aaAA	9ab? a? Xa	XXX-xxx-AAA-bbb
20	Ab?abab	XXX-999	B?B?B?AAA

ЗЕ.3. Создать программу, выполняющую заданное действие с набором чисел, вводимых в динамически создаваемые поля

Создать форму, содержащую следующие элементы:

- блок ввода чисел: *TableLayoutPanel*, в котором динамически добавляются элементы *NumericUpDown* со свойствами *Increment* = 0.01 и *DecimalPlaces* = 2
- кнопка «Вычислить»
- блок с результатом

Изначально в блоке ввода чисел должен быть только один элемент *NumericUpDown*. Если фокус находится на этом элементе и пользователь нажимает клавишу «Enter», ниже должно появиться следующее поле ввода и фокус должен переместиться на него. Если фокус находится на одном из элементов блока ввода и пользователь нажимает клавишу «Delete», этот элемент должен удалиться (если он не последний) и фокус должен переместиться на элемент выше (или ниже, если это первый элемент).

При нажатии на кнопку «Вычислить» программа должна выполнить требуемую операцию со всеми числами из блока ввода и вывести полученное значение в блоке результата

Пример

Просуммировать числа

Варианты

№	Операция	№	Операция
1	Перемножить все положительные числа	11	Умножить числа на их порядковые номера и сложить результаты
2	Разделить все числа на первое из чисел и вычислить сумму результатов	12	Извлечь квадратный корень из суммы квадратов чисел
3	Просуммировать все четные числа	13	Просуммировать все нечетные числа
4	Просуммировать все числа, которые меньше, чем предыдущее число	14	Умножить числа на их порядковые номера и вычислить среднее
5	Просуммировать все числа, которые больше первого из чисел	15	У каждого второго числа поменять знак и посчитать среднее
6	Просуммировать квадраты всех чисел	16	Округлить числа и сложить результаты

7	Просуммировать дробные части всех чисел	17	У каждого второго числа поменять знак и просуммировать все числа
8	Вычислить среднее значение	18	Просуммировать модули всех чисел
9	Перемножить все отрицательные числа	19	Вычислить среднее из квадратов чисел
10	Просуммировать все числа, которые больше, чем предыдущее число	20	Перемножить дробные части всех чисел

3Е.4. Создать программу для редактирования текста с диалогами открытия и сохранения файла

Создать форму, содержащую элементы (см. пример):

- меню (*MenuStrip*):
 - подменю «Файл»
 - кнопка «Открыть...»
 - кнопка «Сохранить»
 - кнопка «Сохранить как...»
 - кнопка «Выход»
 - кнопка «Преобразовать текст»
- многострочное поле ввода текста (*TextBox* со свойством *Multiline* = true)

При запуске программы кнопка «Сохранить» в меню должна быть неактивной

Кнопка в меню «Открыть» должна открывать стандартный диалог открытия файла, в котором доступны для выбора только файлы с расширением .txt. После выбора файла его содержимое помещается в поле ввода текста (старое значение очищается) и становится активной кнопка «Сохранить»

Кнопка «Сохранить» сохраняет текущий текст из поля ввода в последний из открытых файлов

Кнопка «Сохранить как...» должна открывать стандартный диалог сохранения файла с расширением по умолчанию .txt. После выбора файла текущий текст из поля ввода записывается в выбранный файл.

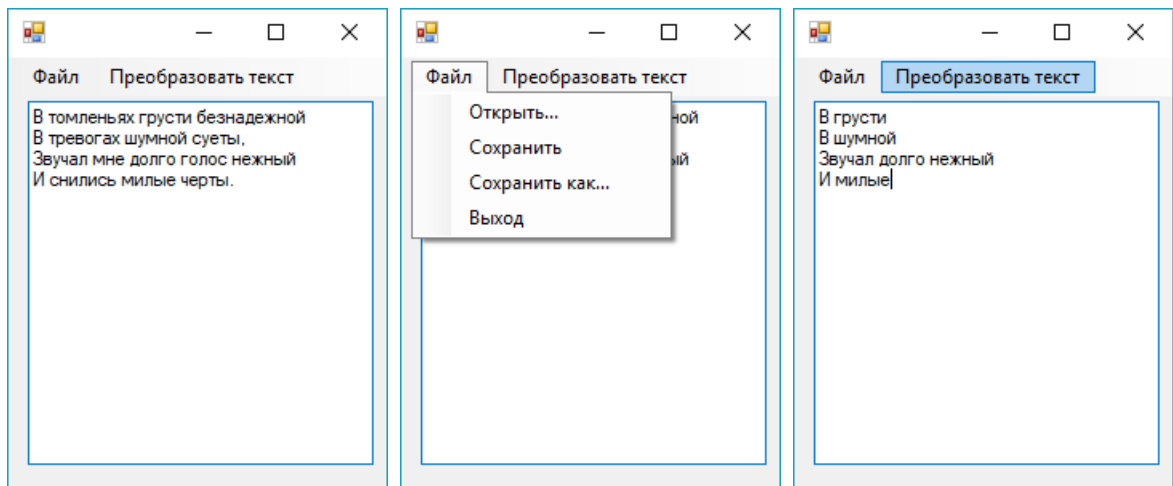
Кнопка «Выход» должна закрывать программу

Кнопка «Преобразовать текст» должна изменить текст в поле ввода согласно варианту

При закрытии формы, если были несохраненные изменения, программа должна попросить подтверждение выхода во всплывающем окне.

Пример

Удаление каждого второго слова



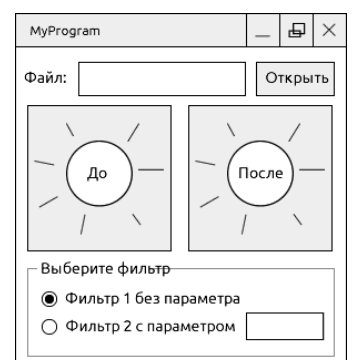
Варианты

№	Преобразование	№	Преобразование
1	Перевести первую букву каждого слова в верхний регистр	11	Перевести в верхний регистр все гласные
2	Перевернуть строки (абг деж -> жед гба)	12	Удалить все слова, в которых больше 5 согласных
3	Перевести все буквы первого слова каждой строке в верхний регистр	13	Перевернуть каждое слово в строке (абг деж -> гба жед)
4	Перевести в верхний регистр каждую вторую букву слов: абв гдеж -> аБв гДеЖ	14	Удалить все слова, в которых меньше 5 согласных
5	Поменять местами слова в каждой паре слов: аб вг де жз -> вг аб жз де	15	Удалить первое и последнее слова в строках
6	Удалить все слова размером больше, чем из 5 букв	16	Удалить все согласные
7	Удалить все слова без согласных букв	17	Удалить все гласные
8	Удалить знаки препинания	18	Перемешать в случайном порядке все слова в строке
9	Удалить все слова без гласных букв	19	Удалить все слова размером меньше, чем из 5 букв
10	Удаление каждого второго слова	20	Перевести в верхний регистр все согласные

3Е.5.Создать форму применения фильтра к ВМР-изображению

Создать форму, содержащую следующие элементы:

- поле ввода имени файла
- кнопка «Открыть»
- блок отображения входного изображения
- блок выбора фильтра (группа с радиокнопками)
- блок отображения результирующего изображения

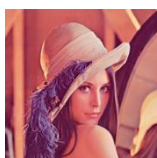


Блок выбора фильтра должен предлагать на выбор один из двух фильтров (согласно варианту). Некоторые фильтры требуют ввода дополнительного параметра. Изначально выбран первый фильтр. Одновременно активным может быть только один фильтр.

При нажатии на кнопку «Открыть» программа должна открыть указанный BMP-файл, отобразить его в блоке входного изображения, применить к нему текущий активный фильтр и вывести измененное изображение в блоке результирующего изображения. Если файл не найден или его невозможно открыть, вывести сообщение об ошибке во всплывающем окне.






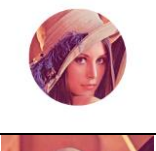


При смене активного фильтра, если входное изображение уже загружено, результирующее изображение должно обновиться автоматически.

Пример входного изображения:



Список фильтров:

Код	Описание	Пример работы
Ф1	Перевод в градации серого (усреднение всех каналов цвета)	
Ф2	Бинаризация по порогу (задается параметр Т, пиксели, яркость которых меньше Т, становятся черными, больше Т - белыми)	
Ф3	Отражение изображения по горизонтали	
Ф4	Отражение изображения по вертикали	
Ф5	Отражение изображения по горизонтали и по вертикали	
Ф6	Добавление черной рамки вокруг изображения (задается параметр W, определяющий размер рамки)	
Ф7	Зануление всех каналов, кроме красного	

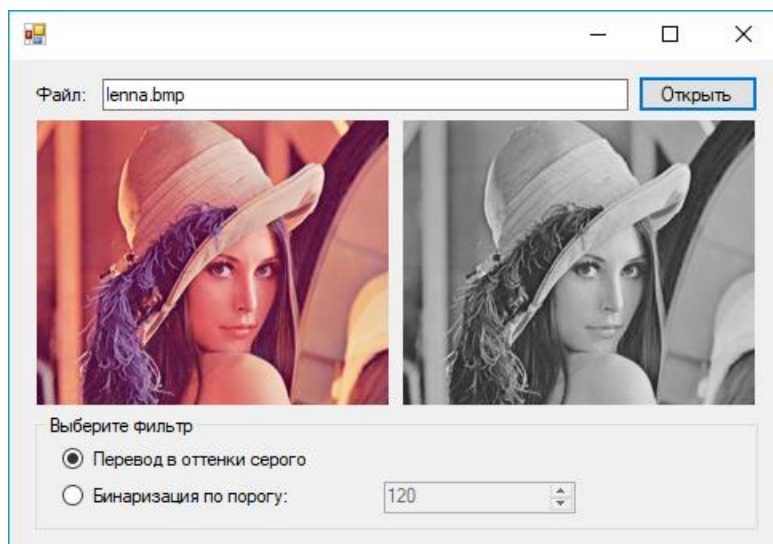
Ф8	Зануление всех каналов, кроме синего	
Ф9	Зануление всех каналов, кроме зеленого	
Ф10	Усиление черного (задается параметр Т, все пиксели меньше этого значения, становятся черными по каждому из каналов)	
Ф11	Усиление белого (задается параметр Т, все пиксели больше этого значения, становятся белыми по каждому из каналов)	
Ф12	Инвертирование цветов (негатив)	
Ф13	Выделить в круг (задается параметр R - радиус круга. Все пиксели. находящиеся дальше от центра, чем R, становятся белыми)	
Ф14	Вырезать круг (задается параметр R - радиус круга. Все пиксели. находящиеся ближе от центра, чем R, становятся белыми)	
Ф15	Удвоение значения яркости	

Для открытия и работы с изображением используйте класс *System.Drawing.Bitmap*. Подключите в проекте ссылку на стандартную библиотеку *System.Drawing.dll*, если класс недоступен.

Для получения цвета пикселя используйте *Bitmap.GetPixel(Int32, Int32)*. Для установки значения пикселя – *Bitmap.SetPixel(Int32, Int32, System.Drawing.Color)*

Пример

Фильтры: перевод в оттенки серого (Ф1), бинаризация по порогу (Ф2)



Варианты

№	Фильтр		№	Фильтр		№	Фильтр		№	Фильтр	
	1	2		1	2		1	2		1	2
1	Ф1	Ф8	6	Ф15	Ф8	11	Ф13	Ф9	16	Ф6	Ф12
2	Ф13	Ф9	7	Ф14	Ф14	12	Ф10	Ф3	17	Ф11	Ф4
3	Ф2	Ф11	8	Ф10	Ф9	13	Ф7	Ф4	18	Ф13	Ф5
4	Ф2	Ф7	9	Ф1	Ф5	14	Ф12	Ф5	19	Ф6	Ф11
5	Ф3	Ф9	10	Ф10	Ф15	15	Ф4	Ф15	20	Ф14	Ф12