

Министерство образования Московской области

ГБПОУ МО «XXXX»

09.02.07

Дисциплина: Основы алгоритмизации и программирования

ЛАБОРАТОРНЫЕ РАБОТЫ

ОТЧЁТ

ККОО.ОАXXXX.000

Студент:

Преподаватель:

Дата:

Оценка:

2021

Содержание

1	Лабораторная работа 1	3
2	Лабораторная работа 2	8
3	Лабораторная работа 3	28
4	Лабораторная работа 4	34
5	Лабораторная работа 5-6	55
6	Лабораторная работа 7	94
7	Лабораторная работа 8	151
8	Лабораторная работа 9	158

					ККОО.ОАXXXX.000			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		XXXX			Отчет по лабораторным работам.	Лит.	Лист	Листов
Провер.		XXXX						
Реценз.						Гр. XXXX		
Н. Контр.								
Утверд.								

Лабораторная работа №1

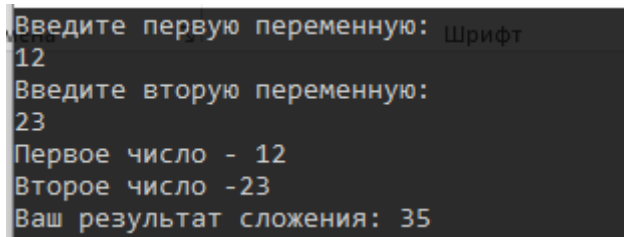
Тема: Разработка простейших консольных приложений.

Цель: Овладение возможностью использования команд ввода-вывода.

Часть 2.

1. Написать программу, которая запрашивает с клавиатуры два целых числа и выводит на экран сумму данных чисел:

```
string first;  
string second;  
Console.WriteLine("Введите первую переменную: ");  
first = Console.ReadLine();  
int first_int = Convert.ToInt32(first);  
Console.WriteLine("Введите вторую переменную: ");  
second = Console.ReadLine();  
int second_int = Convert.ToInt32(second);  
int result = first_int + second_int;  
Console.WriteLine($"Первое число - {first_int}");  
Console.WriteLine($"Второе число - {second_int}");  
Console.WriteLine($"Ваш результат сложения: {result}");  
Console.ReadKey();
```



```
Введите первую переменную: Шрифт  
12  
Введите вторую переменную:  
23  
Первое число - 12  
Второе число - 23  
Ваш результат сложения: 35
```

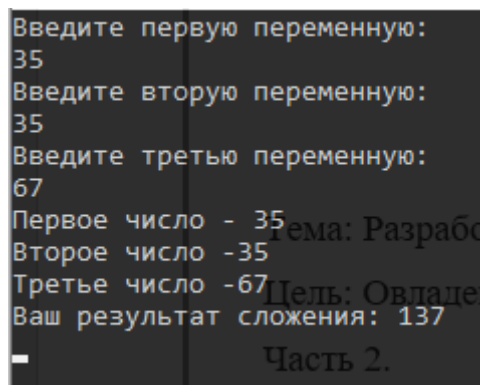
2. Написать программу, которая запрашивает с клавиатуры три целых числа и выводит на экран сумму данных чисел:

```
string first;  
string second;  
string third;  
Console.WriteLine("Введите первую переменную: ");
```

```

first = Console.ReadLine();
int first_int = Convert.ToInt32(first);
Console.WriteLine("Введите вторую переменную: ");
second = Console.ReadLine();
int second_int = Convert.ToInt32(second);
Console.WriteLine("Введите третью переменную: ");
third = Console.ReadLine();
int third_int = Convert.ToInt32(third);
int result = first_int + second_int + third_int;
Console.WriteLine($"Первое число - {first_int}");
Console.WriteLine($"Второе число - {second_int}");
Console.WriteLine($"Третье число - {third_int}");
Console.WriteLine($"Ваш результат сложения: {result}");
Console.ReadKey();

```



```

Введите первую переменную:
35
Введите вторую переменную:
35
Введите третью переменную:
67
Первое число - 35
Второе число - 35
Третье число - 67
Ваш результат сложения: 137

```

3. Написать программу, которая запрашивает с клавиатуры два вещественных числа, и выводит на экран результат деления первого числа на второе (вещественные числа выводятся с точностью до 3 знаков после запятой):

```

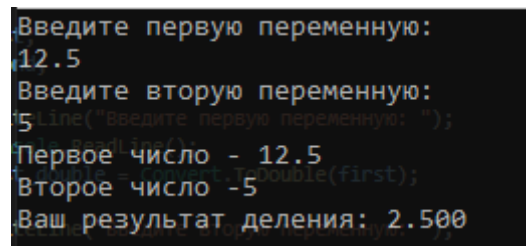
string first;
string second;
Console.WriteLine("Введите первую переменную: ");
first = Console.ReadLine();
double first_double = Convert.ToDouble(first);

```

```

Console.WriteLine("Введите вторую переменную: ");
second = Console.ReadLine();
double second_double = Convert.ToDouble(second);
double result = first_double / second_double;
Console.WriteLine($"Первое число - {first_double:f3}");
Console.WriteLine($"Второе число - {second_double:f3}");
Console.WriteLine($"Ваш результат деления: {result:f3}");
Console.ReadKey();

```



```

Введите первую переменную:
12.5
Введите вторую переменную:
5
Первое число - 12.5
Второе число - 5
Ваш результат деления: 2.500

```

4. Написать программу, которая запрашивает с клавиатуры три вещественных числа и выводит на экран следующее сообщение (вещественные числа выводятся с точностью до 2 знаков после запятой):

```

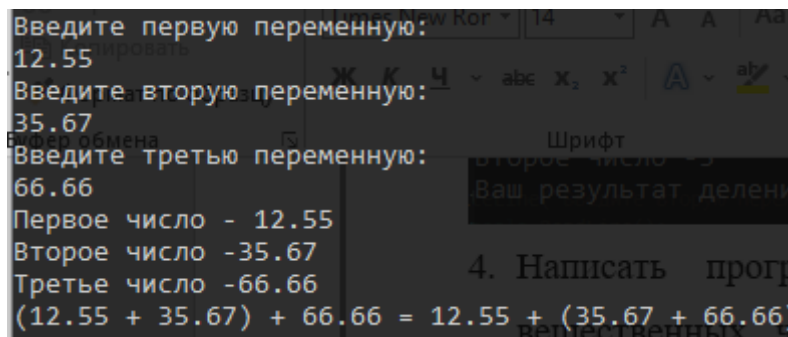
string first;
string second;
string third;
Console.WriteLine("Введите первую переменную: ");
first = Console.ReadLine();
Double first_f = Convert.ToDouble(first);
Console.WriteLine("Введите вторую переменную: ");
second = Console.ReadLine();
Double second_f = Convert.ToDouble(second);
Console.WriteLine("Введите третью переменную: ");
third = Console.ReadLine();
Double third_f = Convert.ToDouble(third);
Console.WriteLine($"Первое число - {first_f}");
Console.WriteLine($"Второе число - {second_f}");

```

```

Console.WriteLine($"Третье число -{third_f}");
Console.WriteLine($"({first_f:f2} + {second_f:f2}) + {third_f:f2} =
{first_f:f2} + ({second_f:f2} + {third_f:f2})");
Console.ReadKey();

```



```

Введите первую переменную:
12.55
Введите вторую переменную:
35.67
Введите третью переменную:
66.66
Первое число - 12.55
Второе число -35.67
Третье число -66.66
(12.55 + 35.67) + 66.66 = 12.55 + (35.67 + 66.66)

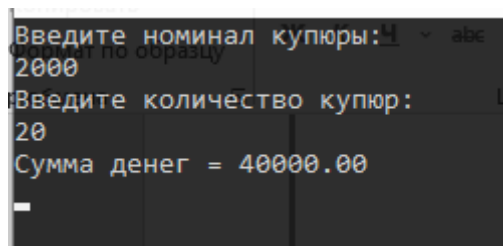
```

5. Написать программу, которая запрашивает с клавиатуры номинал купюры и количество купюр, и выводит экран следующее сообщение:

```

Console.WriteLine("Введите номинал купюры:");
Double currency = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("Введите количество купюр:");
int count = Convert.ToInt32(Console.ReadLine());
Double result = count * currency;
Console.WriteLine($"Сумма денег = {result:f2}");
Console.ReadKey();

```



```

Введите номинал купюры:
2000
Введите количество купюр:
20
Сумма денег = 40000.00

```

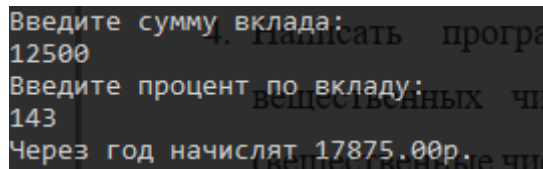
6. Написать программу, которая запрашивает с клавиатуры сумму вклада и процент по вкладу, и выводит на экран начисленную в конце года сумму:

```

Console.WriteLine("Введите сумму вклада:");
double sum = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("Введите процент по вкладу:");
double percent = Convert.ToDouble(Console.ReadLine());

```

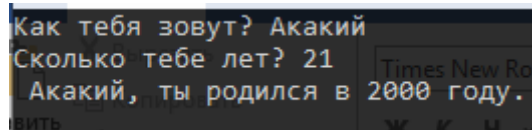
```
double result = sum * (percent / 100);
Console.WriteLine($"Через год начислят {result:f2}" + "р.");
Console.ReadKey();
```



```
Введите сумму вклада:
12500
Введите процент по вкладу:
143
Через год начислят 17875.00р.
```

7. Написать программу, которая запрашивает с клавиатуры имя человека и его возраст, и выводит на экран следующее сообщение (в примере текущим годом считается 2009):

```
Console.Write("Как тебя зовут?");
string name = Console.ReadLine();
Console.Write("Сколько тебе лет?");
int age = Convert.ToInt32(Console.ReadLine());
int dob = 2021 - age;
Console.WriteLine($"{name}, ты родился в {dob} году.");
Console.ReadKey();
```



```
Как тебя зовут? Акакий
Сколько тебе лет? 21
Акакий, ты родился в 2000 году.
```

Лабораторная работа №2.

Часть 2.1

Тема: Целый тип данных

Существует восемь целых типов. Они отличаются диапазоном значений, а значит, и размером памяти, отводимой для их представления.

Тип	Диапазон значений	Размер (байт)
sbyte	-128 ... 127	1
byte	0 ... 255	1
short	-32 768 ... 32 767	2
ushort	0 ... 65 535	2
int	-2 147 483 648 ... 2 147 483 647	4
uint	0 ... 4 294 967 295	4
long	-9 223 372 036 854 775 808 ... 9 223 372 036 854 775 807	8
ulong	0 ... 18 446 744 073 709 551 615	8

Операции с величинами целого типа: сложение (+), вычитание (-), умножение (*), нахождение целой части деления (/), нахождение остатка от деления (%).

В следующей программе определяются цифры трехзначного числа. Можно ее использовать и для определения цифр двузначного числа, просто цифра сотен в этом случае равна нулю, и это делается проще.

```
int a, one, dec, hun, rez;
```

```
Console.WriteLine("Введите трехзначное число");
```

```
a = int.Parse(Console.ReadLine());
```

```
one = a % 10;
```

```
Console.WriteLine("Цифра единиц числа - {0}", one);
```

```
dec = (a / 10) % 10;
```

```
Console.WriteLine("Цифра десятков числа - {0}", dec);
```

```
hun = a / 100;
```

```
Console.WriteLine("Цифра сотен числа - {0}", hun);
```



```
rez = hun * 100 + dec * 10 + one;
Console.WriteLine("это тоже число - {0}", rez);
Console.ReadKey();
```

$$\begin{array}{r} 137 \overline{)10} \\ 130 \overline{)13} \overline{)10} \\ 7 \overline{)10} \overline{)1} \\ 3 \end{array}$$

Например, если Вы введете число 137, то значение переменной one будет равно 7, dec — 3 и hun — 1. Вспомните деление чисел столбиком.

Экспериментальный раздел работы:

1. Измените программу для нахождения цифр двузначного числа. Сохраните ее.

```
int a, one, dec, hun, rez;
Console.WriteLine("Введите двузначное число");
a = Convert.ToInt32(Console.ReadLine());
one = a % 10;
Console.WriteLine("Цифра единиц числа - {0}", one);
dec = (a / 10) % 10;
Console.WriteLine("Цифра десятков числа - {0}", dec);
hun = a / 100;
rez = dec * 10 + one;
Console.WriteLine("Результат - {0}", rez);
Console.ReadKey();
```

```
Введите двузначное число
27
Цифра единиц числа - 7
Цифра десятков числа - 2
Результат - 27
_
```

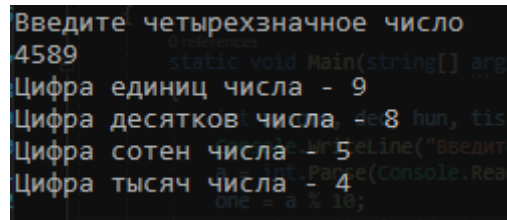
2. Измените программу для нахождения цифр четырехзначного числа. Сохраните ее.

```
int a, one, dec, hun, tis, rez;
Console.WriteLine("Введите четырехзначное число");
```

```

a = int.Parse(Console.ReadLine());
one = a % 10;
Console.WriteLine("Цифра единиц числа - {0}", one);
dec = (a / 10) % 10;
Console.WriteLine("Цифра десятков числа - {0}", dec);
hun = (a / 100) % 10;
Console.WriteLine("Цифра сотен числа - {0}", hun);
tis = a / 1000;
Console.WriteLine("Цифра тысяч числа - {0}", tis);
Console.ReadKey();

```



```

Введите четырехзначное число
4589
Цифра единиц числа - 9
Цифра десятков числа - 8
Цифра сотен числа - 5
Цифра тысяч числа - 4

```

Задания для самостоятельной работы

- Дано двузначное число. Определить:
 - сумму и произведение цифр числа;
 - число, образованное перестановкой цифр исходного числа.

```
int chislo, a, b, result;
```

```
Console.WriteLine("Введите двузначное число:");
```

```
chislo = Convert.ToInt32(Console.ReadLine());
```

```
a = chislo % 10;
```

```
b = chislo / 10;
```

```
Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}
\nДесятки - {b}");
```

```
int summa = a + b;
```

```
int proizv = a * b;
```

```
Console.WriteLine($"Сумма цифр - {summa} \nПроизведение -
{proizv}");
```

```
Console.WriteLine($"Число, образованное перестановкой числа  
{chislo} - {a}{b}");
```

```
Console.ReadKey();
```

```
Введите двузначное число:
27
Ваше число - 27
Единицы - 7
Десятки - 2
Число, образованное перестановкой числа {chislo} - {a}{b}
Сумма цифр - 9
Произведение - 14
Число, образованное перестановкой числа 27 - 72
```

2. Дано трехзначное число. Определить:

- сумму и произведение цифр числа;
- число, образованное перестановкой цифр исходного числа;
- число, полученное перестановкой цифр десятков и единиц;
- число, полученное перестановкой цифр сотен и десятков;
- четырехзначное число, полученное приписыванием цифры единиц в качестве цифры тысяч (например, из числа 137 необходимо получить число 7137).

Примечание

Сколько различных чисел можно получить из трехзначного числа путем перестановки цифр? Ответ: 6

```
int chislo, a, b, c;
```

```
Console.WriteLine("Введите трехзначное число:");
```

```
chislo = Convert.ToInt32(Console.ReadLine());
```

```
a = chislo % 10;
```

```
b = chislo / 10 % 10;
```

```
c = chislo / 100;
```

```
Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}\nДесятки - {b} \nСотни - {c} ");
```

```
int summa = a + b + c;
```

```
int proizv = a * b * c;
```

```

Console.WriteLine($"Сумма цифр - {summa} \nПроизведение - {proizv}");

Console.WriteLine($"Число, образованное перестановкой числа {chislo} - {a}{b}{c}");

Console.WriteLine($"Число, полученное перестановкой цифр десятков и единиц - {c}{a}{b}");

Console.WriteLine($"Число, полученное перестановкой цифр сотен и десятков - {b}{c}{a}");

Console.WriteLine($"Полученное четырехзначное число - {a}{c}{b}{a}");

Console.ReadKey();

```

```

Введите трехзначное число:
359
Ваше число - 359
Единицы - 9
Десятки - 5
Сотни - 3
Сумма цифр - 17
Произведение - 135
Число, образованное перестановкой числа 359 - 953
Число, полученное перестановкой цифр десятков и единиц - 395
Число, полученное перестановкой цифр сотен и десятков - 539
Полученное четырехзначное число - 9359

```

3. Решить задачу 2 (кроме последнего пункта) для четырехзначных чисел.

```

int chislo, a, b, c, d;

Console.WriteLine("Введите четырехзначное число:");

chislo = Convert.ToInt32(Console.ReadLine());

a = chislo % 10;
b = (chislo / 10) % 10;
c = (chislo / 100) % 10;
d = chislo / 1000;

Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a} \nДесятки - {b} \nСотни - {c} \nТысячи - {d} ");

int summa = a + b + c + d;

```

```

int proizv = a * b * c * d;

Console.WriteLine($"Сумма цифр - {summa} \nПроизведение - {proizv}");

Console.WriteLine($"Число, образованное перестановкой числа {chislo} - {a}{b}{c}{d}");

Console.WriteLine($"Число, полученное перестановкой цифр десятков и единиц - {d}{c}{a}{b}");

Console.WriteLine($"Число, полученное перестановкой цифр сотен и десятков - {d}{b}{c}{a}");

Console.ReadKey();

```

```

Введите четырехзначное число:
3509
Ваше число - 3509
Единицы - 9
Десятки - 0
Сотни - 5
Тысячи - 3
Сумма цифр - 17
Произведение - 0
Число, образованное перестановкой числа 3509 - 9053
Число, полученное перестановкой цифр десятков и единиц - 3590
Число, полученное перестановкой цифр сотен и десятков - 3059

```

4. Арифметическая прогрессия — это последовательность чисел, в которой разность между последующим и предыдущим элементами остается неизменной. Последовательность 12, 15, 18, 21, 24, ... является арифметической прогрессией, 12 — первый член прогрессии (a_1), разность прогрессии равна 3. Любой член прогрессии вычисляется по формуле $a_n = a_1 + d \cdot (n - 1)$, где d — разность прогрессии, n — номер взятого члена. Даны a_1 и d . Найти n , при котором значение a_n выходит за диапазон типа short (экспериментальным путем).

```

int a_1 = 12;

int d = 3;

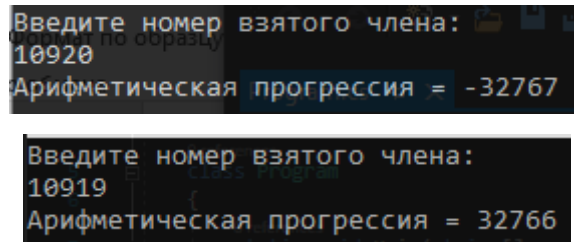
Console.WriteLine("Введите номер взятого члена:");

int n = Convert.ToInt32(Console.ReadLine());

short a_n = (short)(a_1 + d * (n - 1));

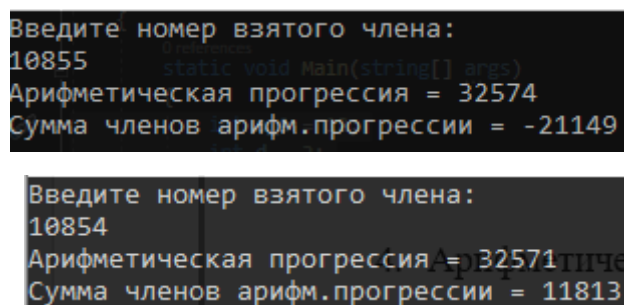
```

```
Console.WriteLine($"Арифметическая прогрессия = {a_n}");
//10920
```



5. Сумма первых n членов арифметической прогрессии вычисляется по формуле $S_n = (a_1 + a_n) * n / 2$. Даны a_1 и d . Найти n , при котором значение S_n выходит за диапазон типа short (экспериментальным путем).

```
int a_1 = 12;
int d = 3;
Console.WriteLine("Введите номер взятого члена:");
int n = Convert.ToInt32(Console.ReadLine());
short a_n = (short)(a_1 + d * (n - 1));
Console.WriteLine($"Арифметическая прогрессия = {a_n}");
short s_n = (short)((a_1 + a_n) * n / 2);
Console.WriteLine($"Сумма членов арифм.прогрессии = {s_n}");
//10855
```



Часть 2.2

Разработать консольное приложение, с помощью которого можно выполнить математические вычисления (см. вариант).

При выполнении заданий допустимо использовать справочную литературу. Ввод и вывод информации на экране сопровождать соответствующими надписями.

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		14

1. Дано четырехзначное число. Найти сумму первой и последней цифры этого числа.

```
int chislo, first, ten, hun, thous, result;  
Console.WriteLine("Введите четырехзначное число:");  
chislo = Convert.ToInt32(Console.ReadLine());  
first = chislo % 10;  
ten = (chislo / 10) % 10;  
hun = (chislo / 100) % 10;  
thous = chislo / 1000;  
result = first + thous;  
Console.WriteLine($"Результат сложения первой и последней  
цифры - {result}");  
Console.ReadKey();
```

```
Введите четырехзначное число:  
1337  
Результат сложения первой и последней цифры - 8
```

2. Дано четырехзначное число. Переставить местами первую и последнюю цифры этого числа.

```
int chislo, first, ten, hun, thous, result;  
Console.WriteLine("Введите четырехзначное число:");  
chislo = Convert.ToInt32(Console.ReadLine());  
first = chislo % 10;  
ten = (chislo / 10) % 10;  
hun = (chislo / 100) % 10;  
thous = chislo / 1000;  
  
result = first * 1000 + hun * 100 + ten * 10 + thous;  
Console.WriteLine($"Ваше число - {chislo} \nЧисло, образованное  
перестановкой первой и последней цифры - {result}");  
Console.ReadKey();
```

```

Введите четырёхзначное число:
1337
Ваше число - 1337
Число, образованное перестановкой первой и последней цифры - 7331

```

3. Дано натуральное число $n < 99$. Дописать к нему цифру k в конец и в начало.

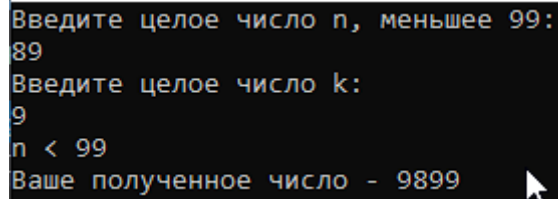
```

Console.WriteLine("Введите целое число n, меньшее 99:");
int n = Convert.ToInt32(Console.ReadLine());
int first = n % 10;
int ten = (n / 10) % 10;
Console.WriteLine("Введите целое число k:");
int k = Convert.ToInt32(Console.ReadLine());
if (n < 99)
{
    if (k < 10)
    {
        int result = k * 1000 + n * 10 + k;
        Console.WriteLine($"n < 99 \nВаше полученное число -
{result}");
    }
    else
    {
        Console.WriteLine($"Вы ввели k > 10. ({k}>10)");
    }
}
else
{
    Console.WriteLine($"Число {n} > 99. \nПолучить число не
удалось.");
}

```



```
Console.ReadKey();
```



```
Введите целое число n, меньшее 99:  
89  
Введите целое число k:  
9  
n < 99  
Ваше полученное число - 9899
```

4. Дано натуральное число N из 6 цифр. Определить сумму цифр в первой половине числа (старшие разряды). Пример. N=123 456. Сумма составляет $1+2+3=6$

```
int chislo, first, ten, hun, result, chislo_preobr;
```

```
Console.WriteLine("Введите шестизначное число:");
```

```
chislo = Convert.ToInt32(Console.ReadLine());
```

```
chislo_preobr = chislo / 1000;
```

```
first = chislo_preobr % 10;
```

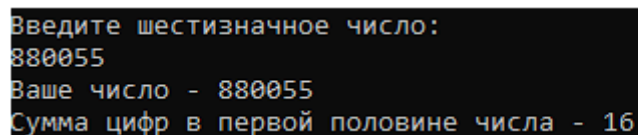
```
ten = (chislo_preobr / 10) % 10;
```

```
hun = (chislo_preobr / 100) % 10;
```

```
result = first + ten + hun;
```

```
Console.WriteLine($"Ваше число - {chislo}\nСумма цифр в  
первой половине числа - {result}");
```

```
Console.ReadKey();
```



```
Введите шестизначное число:  
880055  
Ваше число - 880055  
Сумма цифр в первой половине числа - 16
```

5. Дано натуральное число N из 6 цифр. Определить сумму цифр во второй половине числа (младшие разряды). Пример. N=123 456. Сумма составляет $4+5+6=15$

```
int chislo, first, ten, hun, result;
```

```
Console.WriteLine("Введите шестизначное число:");
```

```
chislo = Convert.ToInt32(Console.ReadLine());
```

```
first = chislo % 10;
```

```
ten = (chislo / 10) % 10;
```

```

hun = (chislo / 100) % 10;
result = first + ten + hun;
Console.WriteLine($"Ваше число - {chislo}\nСумма цифр во
второй половине числа - {result}");
Console.ReadKey();

```

```

Введите шестизначное число:
880055
Ваше число - 880055
Сумма цифр в первой половине числа - 10

```

6. Дано натуральное число из 5 цифр, получить новое число М, которое образуется путем исключения средней цифры исходного числа. Пример. N = 12345, M= 1245.

```

int chislo, first, second;

Console.WriteLine("Введите пятизначное число:");
chislo = Convert.ToInt32(Console.ReadLine());
first = chislo / 1000;
second = chislo % 100;
int result = first * 100 + second;

```

```

Console.WriteLine($"Число, образованное путем исключения
средней цифры числа {chislo} - {result}");
Console.ReadKey();

```

```

Введите пятизначное число:
13377
Число, образованное путем исключения средней цифры числа 13377 - 1377

```

7. Дано трехзначное число. Определить сумму цифр данного числа.

```

int chislo, a, b, c, d;

Console.WriteLine("Введите трехзначное число:");
chislo = Convert.ToInt32(Console.ReadLine());
a = chislo % 10;
b = (chislo / 10) % 10;

```

```
c = (chislo / 100) % 10;
```

```
int result = a + b + c;
```

```
Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}\nДесятки - {b} \nСотни - {c} \nРезультат суммы цифр - {result}");
```

```
Console.ReadKey();
```

```
Введите трехзначное число:
345
Ваше число - 345
Единицы - 5
Десятки - 4
Сотни - 3
Результат суммы цифр - 12
```

8. Дано трехзначное число. Получить новое число, полученное перестановкой цифр десятков и сотен

```
int chislo, a, b, c, d;
```

```
Console.WriteLine("Введите трехзначное число:");
```

```
chislo = Convert.ToInt32(Console.ReadLine());
```

```
a = chislo % 10;
```

```
b = (chislo / 10) % 10;
```

```
c = (chislo / 100) % 10;
```

```
int result = b * 100 + c * 10 + a;
```

```
Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}\nДесятки - {b} \nСотни - {c} \nРезультат перестановки десятков и сотен - {result}");
```

```
Console.ReadKey();
```

```
Введите трехзначное число:
345
Ваше число - 345
Единицы - 5
Десятки - 4
Сотни - 3
Результат перестановки десятков и сотен - 435
```

9. Дано трехзначное число. Получить новое число, полученное

перестановкой цифр единиц и сотен.

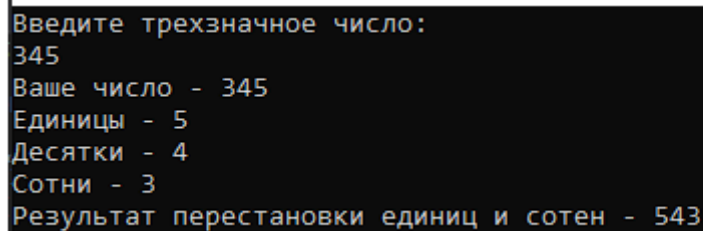
```
int chislo, a, b, c, d;

Console.WriteLine("Введите трехзначное число:");
chislo = Convert.ToInt32(Console.ReadLine());
a = chislo % 10;
b = (chislo / 10) % 10;
c = (chislo / 100) % 10;

int result = a * 100 + b * 10 + c;

Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}
\nДесятки - {b} \nСотни - {c} \nРезультат перестановки единиц и сотен -
{result}");

Console.ReadKey();
```



```
Введите трехзначное число:
345
Ваше число - 345
Единицы - 5
Десятки - 4
Сотни - 3
Результат перестановки единиц и сотен - 543
```

10. Дано четырехзначное число. Определить сумму цифр данного числа.

```
int chislo, a, b, c, d;

Console.WriteLine("Введите четырехзначное число:");
chislo = Convert.ToInt32(Console.ReadLine());
a = chislo % 10;
b = (chislo / 10) % 10;
c = (chislo / 100) % 10;
d = chislo / 1000;

int summa = a + b + c + d;

Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}
```

```
\nДесятки - {b} \nСотни - {c} \nТысячи - {d} ");
```

```
Console.WriteLine($"Сумма цифр - {summa} ");
```

```
Console.ReadKey();
```

```
Введите четырехзначное число:
3456
Ваше число - 3456
Единицы - 6
Десятки - 5
Сотни - 4
Тысячи - 3
Сумма цифр - 18
```

11. Дано четырехзначное число. Получить новое число, полученное перестановкой цифр единиц и сотен.

```
int chislo, a, b, c, d;
```

```
Console.WriteLine("Введите четырехзначное число:");
```

```
chislo = Convert.ToInt32(Console.ReadLine());
```

```
a = chislo % 10;
```

```
b = (chislo / 10) % 10;
```

```
c = (chislo / 100) % 10;
```

```
d = chislo / 1000;
```

```
int newest = d * 1000 + c * 100 + a * 10 + b;
```

```
Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}
```

```
\nДесятки - {b} \nСотни - {c} \nТысячи - {d} ");
```

```
Console.WriteLine($"Число, полученное перестановкой цифр
единиц и сотен - {newest} ");
```

```
Console.ReadKey();
```

```
Введите четырехзначное число:
3478
Ваше число - 3478
Единицы - 8
Десятки - 7
Сотни - 4
Тысячи - 3
Число, полученное перестановкой цифр тысяч и сотен - 3487
_
```

12. Дано четырехзначное число. Получить новое число, полученное

перестановкой цифр тысяч и сотен

```
int chislo, a, b, c, d;

Console.WriteLine("Введите четырехзначное число:");
chislo = Convert.ToInt32(Console.ReadLine());

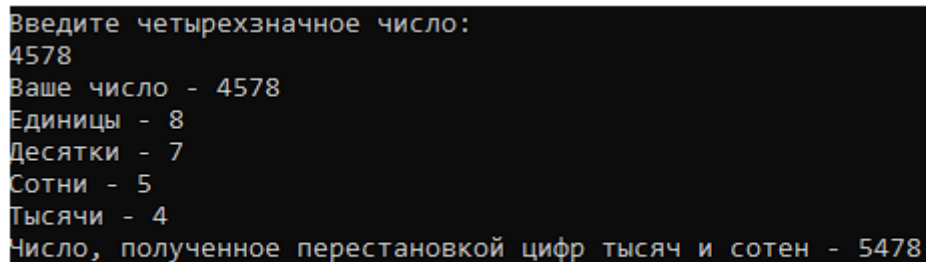
a = chislo % 10;
b = (chislo / 10) % 10;
c = (chislo / 100) % 10;
d = chislo / 1000;

int newest = c * 1000 + d * 100 + b * 10 + a;

Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}
\nДесятки - {b} \nСотни - {c} \nТысячи - {d} ");

Console.WriteLine($"Число, полученное перестановкой цифр
тысяч и сотен - {newest} ");

Console.ReadKey();
```



```
Введите четырехзначное число:
4578
Ваше число - 4578
Единицы - 8
Десятки - 7
Сотни - 5
Тысячи - 4
Число, полученное перестановкой цифр тысяч и сотен - 5478
```

13. Дано четырехзначное число. Получить новое число, полученное перестановкой цифр единиц и сотен

```
int chislo, a, b, c, d;

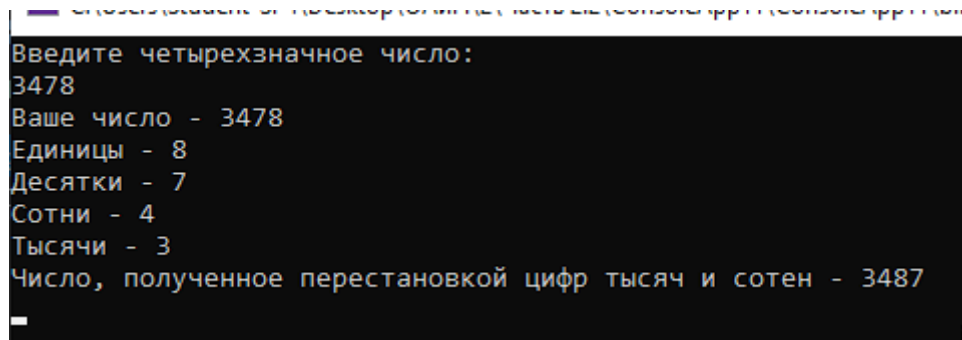
Console.WriteLine("Введите четырехзначное число:");
chislo = Convert.ToInt32(Console.ReadLine());

a = chislo % 10;
b = (chislo / 10) % 10;
c = (chislo / 100) % 10;
d = chislo / 1000;
```

```

int newest = d * 1000 + c * 100 + a * 10 + b;
Console.WriteLine($"Ваше число - {chislo}\nЕдиницы - {a}
\nДесятки - {b} \nСотни - {c} \nТысячи - {d} ");
Console.WriteLine($"Число, полученное перестановкой цифр
единиц и сотен - {newest} ");
Console.ReadKey();

```



```

Введите четырехзначное число:
3478
Ваше число - 3478
Единицы - 8
Десятки - 7
Сотни - 4
Тысячи - 3
Число, полученное перестановкой цифр тысяч и сотен - 3487

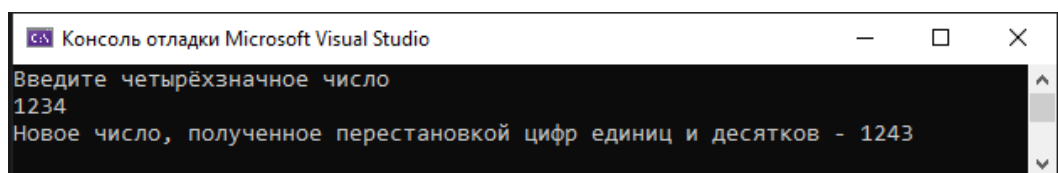
```

14. Дано четырехзначное число. Получить новое число, полученное перестановкой цифр единиц и десятков

```

int number, units, dozens, hundreds, thousand, sum;
Console.WriteLine("Введите четырёхзначное число");
number = int.Parse(Console.ReadLine());
units = number % 10;
dozens = number / 10 % 10;
hundreds = number / 100 % 10;
thousand = number / 1000;
sum = thousand * 1000 + hundreds * 100 + dozens + units * 10;
Console.WriteLine("Новое число, полученное перестановкой
цифр единиц и десятков - {0}", sum);

```



```

Консоль отладки Microsoft Visual Studio
Введите четырёхзначное число
1234
Новое число, полученное перестановкой цифр единиц и десятков - 1243

```

15. Дано четырехзначное число. Определить произведение цифр данного числа

```

int number, units, dozens, hundreds, thousand, proz;

Console.WriteLine("Введите четырёхзначное число");
number = int.Parse(Console.ReadLine());
units = number % 10;
dozens = number / 10 % 10;
hundreds = number / 100 % 10;
thousand = number / 1000;
proz = units * dozens * hundreds * thousand; ;

Console.WriteLine("Новое число, полученное перестановкой
цифр единиц и десятков - {0}", proz);

```

16. Дано четырехзначное число. Найти сумму цифр десятков и единиц этого числа

```

int number, units, dozens, hundreds, thousand, sum;

Console.WriteLine("Введите четырёхзначное число");
number = int.Parse(Console.ReadLine());
units = number % 10;
dozens = number / 10 % 10;
hundreds = number / 100 % 10;
thousand = number / 1000;
sum = dozens + units;

Console.WriteLine("Сумма цифр единиц и десятков - {0}", sum);

```

17. Дано четырехзначное число. Переставить местами вторую и последнюю цифры этого числа.


```

int number, units, dozens, hundreds, thousand, sum;

Console.WriteLine("Введите четырёхзначное число");
number = int.Parse(Console.ReadLine());
units = number % 10;
dozens = number / 10 % 10;
hundreds = number / 100 % 10;
thousand = number / 1000;
sum = thousand * 1000 + hundreds + dozens * 10 + units * 100;
Console.WriteLine("Новое число, полученное перестановкой
второй и последней цифры числа - {0}", sum);

```

18. Дано трехзначное число. Дописать к нему цифру k в конец и в начало.

```

int n, k, s;

Console.WriteLine("Введите трёхзначное число");
n = int.Parse(Console.ReadLine());
Console.WriteLine("Введите цифру k");
k = int.Parse(Console.ReadLine());
s = k * 10000 + n * 10 + k;
Console.WriteLine($"Новое число {s}");
Console.ReadKey();

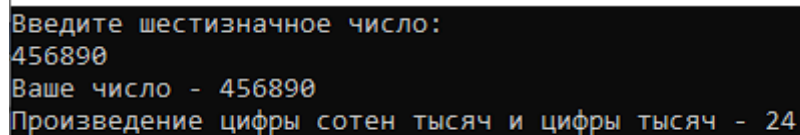
```

Контрольная работа №1

Целочисленная арифметика без цикла

Задание 1: Дано натуральное число N из 6 цифр. Определить произведение цифры сотен тысяч и цифры тысяч. Пример. N=123 456. Ответом будет $1 \cdot 3 = 3$.

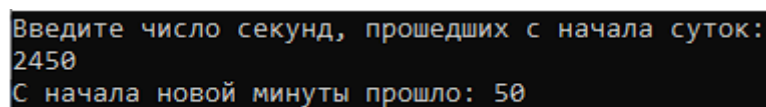
```
int chislo, first, ten, hun, result, chislo_preobr;
Console.WriteLine("Введите шестизначное число:");
chislo = Convert.ToInt32(Console.ReadLine());
chislo_preobr = chislo / 1000; //первая половина вводимого числа
first = chislo_preobr % 10; // сотни тысяч
ten = (chislo_preobr / 10) % 10; // десятки тысяч
hun = (chislo_preobr / 100) % 10; // тысячи
result = first * hun;
Console.WriteLine($"Ваше число - {chislo}\nПроизведение
цифры сотен тысяч и цифры тысяч - {result}");
Console.ReadKey();
```



```
Введите шестизначное число:
456890
Ваше число - 456890
Произведение цифры сотен тысяч и цифры тысяч - 24
```

Задание 2: С начала суток прошло n секунд (n – целое). Определить сколько полных секунд прошло с начала очередной минуты.

```
Console.WriteLine("Введите число секунд, прошедших с начала
суток:");
int n = Convert.ToInt32(Console.ReadLine());
int minute = n % 60;
Console.WriteLine($"С начала новой минуты прошло: {minute}");
```



```
Введите число секунд, прошедших с начала суток:
2450
С начала новой минуты прошло: 50
```

Лабораторная работа №3

Тема: Операторы ветвления.

Цель: Научиться использовать операторы ветвления при выполнении расчетных проектов.

Задание 1. Разработать консольные приложения в соответствии с условием заданий. Решить по одной задаче из каждого раздела. Задачи выполняются в соответствии с номером по журналу.

Задание 1. 1

Вычислить значение функции. Значение x ввести с экрана. Результат вывести на экран.

$$11. F(x) = \begin{cases} 4x^2 + 2x - 19, & \text{если } x \geq -3,5 \\ -\frac{2x}{-4x+1}, & \text{если } x < 3,5 \end{cases}$$

```
double result;
```

```
Console.WriteLine("Введите значение x:");
```

```
int x = Convert.ToInt32(Console.ReadLine());
```

```
if (x > -3.5)
```

```
{
```

```
    result = 4 * Math.Pow(x, 2) + 2 * x - 19;
```

```
    Console.WriteLine($"Значение функции 1 - {result}");
```

```
}
```

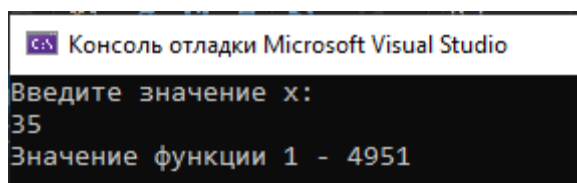
```
if (x < 3.5)
```

```
{
```

```
    result = -(2 * x / -(4 * x) + 1);
```

```
    Console.WriteLine($"Значение функции 2 - {result}");
```

```
}
```

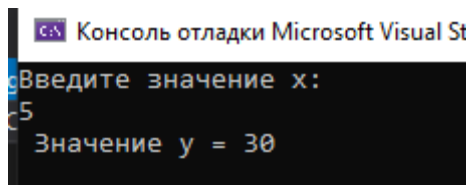


Задание 1.2

Вычислить значение функции. Значение x ввести с экрана. Результат вывести на экран. Использовать два условных оператора.

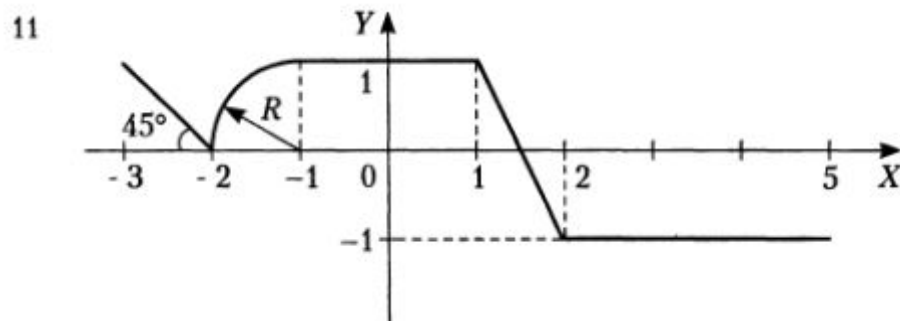
$$11. y = \begin{cases} x^2 + 5, & \text{если } x \leq 5 \\ 0, & \text{если } 5 < x < 20 \\ 1, & \text{если } x \geq 20 \end{cases}$$

```
double result = 0;
Console.WriteLine("Введите значение x:");
int x = Convert.ToInt32(Console.ReadLine());
if (x <= 5)
{
    result = Math.Pow(x, 2) + 5;
}
//if
else
{
    if (x > 5 && x < 20)
    {
        result = 0;
    }
    //if
    if (x >= 20)
    {
        result = 1;
    }
    //if
}
//else
Console.WriteLine($"Значение y = {result}");
```



Задание 2. Разработать консольные приложения в соответствии с условием заданий. Решить по одной задаче из каждого раздела. Задачи выполняются в соответствии с номером по журналу.

Задание 2.1. Написать программу, которая вычисляет значение функции по значению аргумента, введенного с клавиатуры. Функции заданы графически. Параметр R и значение аргумента вводятся с клавиатуры.



```

double y = 0;
Console.WriteLine("Введите значение x:");
double x = Convert.ToInt32(Console.ReadLine());

if (x >= -3 && x <= -2)
{
    y = 4 * x + 1;
}

if (x > -2 && x <= -1)
{
    y = Math.Sqrt(1-Math.Pow((x-1), 2));
}

if (x > -1 && x <= 1)
  
```

```

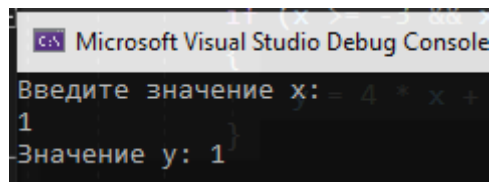
{
    y = 1;
}

if (x > 1 && x <= 2)
{
    y = 1.5 - 0.5 * x;
}

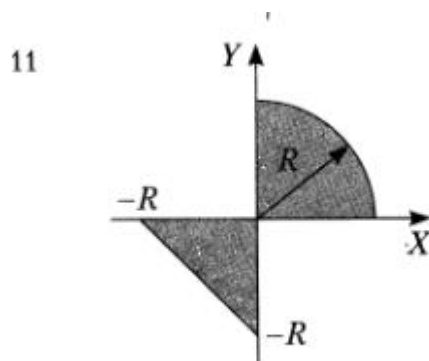
if (x > 2 && x <= 5)
{
    y = -1;
}

Console.WriteLine($"Значение y: {y}");

```



Задание 2.2. Написать программу, которая определяет, попадает ли точка с заданными координатами в область, закрашенную на рисунке серым цветом. Результат работы программы вывести в виде текстового сообщения. Значения координат точки вводить с клавиатуры.



Контрольная работа №2

Тема: Условный оператор

Задание 1. Дано натуральное число. Определить, оканчивается ли оно цифрой 7.

```
int a;  
Console.WriteLine("Введите натуральное число:");  
a = Convert.ToInt32(Console.ReadLine());  
  
int first = a % 10;  
if (first == 7)  
{  
    Console.WriteLine("Число оканчивается на 7");  
}  
  
else  
{  
    Console.WriteLine("Число не оканчивается на 7");  
}  
Console.ReadLine();
```

Задание 2.

```
double x, r;  
double y = 0;  
Console.WriteLine("Введите x:");  
x = Convert.ToDouble(Console.ReadLine());  
  
if (x >= -10 && x <= 0)  
{  
    y = -0.5 * x - 3;  
}
```

```

if (x > 0 && x <= 3)
{
    y = Math.Sqrt(9 - Math.Pow(x, 2));
}

if (x > 3 && x <= 6)
{
    y = Math.Sqrt(9 - Math.Pow((x - 6), 2));
}
Console.WriteLine($"y = {y}");

```

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		32

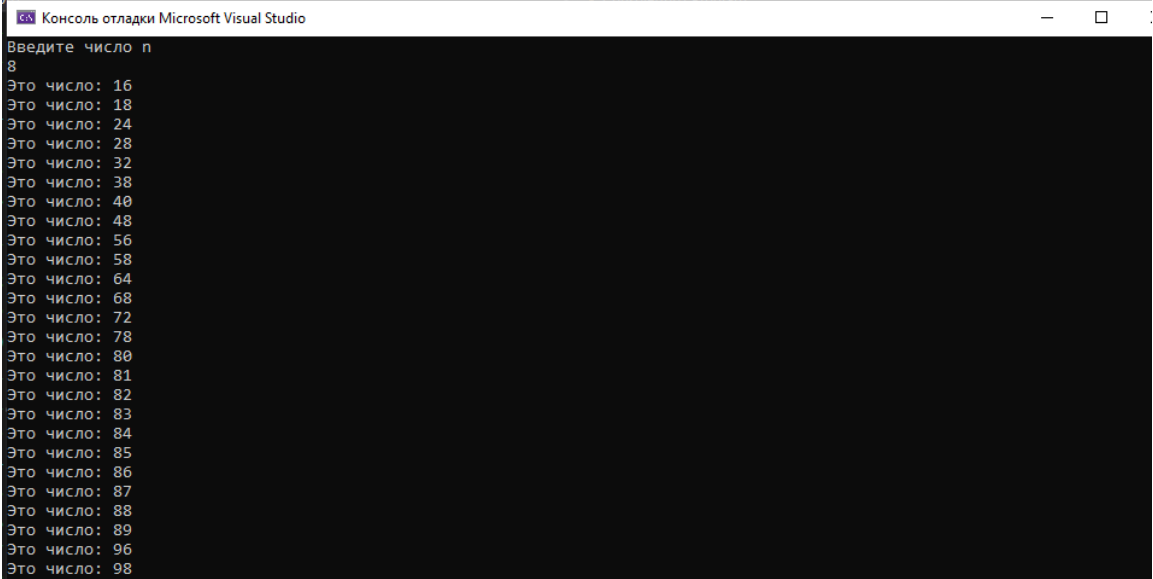
Лабораторная работа №4

Тема: Цикл с параметром

Задания для самостоятельной работы

1. Найти все двузначные числа, в которых есть цифра N или само число делится на N.

```
int n;  
Console.WriteLine("Введите число n");  
n = int.Parse(Console.ReadLine());  
for (int i = 10; i < 99; i++)  
{  
    if ((i / 10 % 10 == n) || (i % 10 == n) || (i / n == 0))  
    {  
        Console.WriteLine($"это число: {i}");  
    }  
}  
Console.ReadKey();
```



```
Консоль отладки Microsoft Visual Studio  
Введите число n  
8  
это число: 16  
это число: 18  
это число: 24  
это число: 28  
это число: 32  
это число: 38  
это число: 40  
это число: 48  
это число: 56  
это число: 58  
это число: 64  
это число: 68  
это число: 72  
это число: 78  
это число: 80  
это число: 81  
это число: 82  
это число: 83  
это число: 84  
это число: 85  
это число: 86  
это число: 87  
это число: 88  
это число: 89  
это число: 96  
это число: 98
```

2. Определить количество трехзначных натуральных чисел, сумма цифр которых равна заданному числу N.

```
double n;  
int count = 0;  
Console.WriteLine("Введите число n");
```

					ККОО.ОАXXXX.000	Лист
						33
Изм.	Лист	№ докум.	Подпись	Дата		

```

n = int.Parse(Console.ReadLine());
for (int i = 100; i < 999; i++)
{
    int one = i % 10;
    int dec = (i / 10) % 10;
    int hun = i / 100;
    int sum = one + dec + hun;
    if (sum == n)
    {
        Console.WriteLine($"Нашли число: {i}");
        count = count + 1;
        Console.WriteLine($"Прибавили к счетчику +1.");
    }

}

}

Console.WriteLine($"Количество найденных трехзначных чисел:
{count}");

Console.ReadKey();

```

```
cs Консоль отладки Microsoft Visual Studio
Нашли число: 540
Прибавили к счетчику +1.
Нашли число: 603
Прибавили к счетчику +1.
Нашли число: 612
Прибавили к счетчику +1.
Нашли число: 621
Прибавили к счетчику +1.
Нашли число: 630
Прибавили к счетчику +1.
Нашли число: 702
Прибавили к счетчику +1.
Нашли число: 711
Прибавили к счетчику +1.
Нашли число: 720
Прибавили к счетчику +1.
Нашли число: 801
Прибавили к счетчику +1.
Нашли число: 810
Прибавили к счетчику +1.
Нашли число: 900
Прибавили к счетчику +1.
Количество найденных трехзначных чисел: 45
```

3. Составить программу вычисления суммы кубов чисел от 25 до 55.

```
double sum = 0;
for (int i = 25; i < 56; i++)
{
    sum += i * i * i;
}
Console.WriteLine($"Сумма кубов чисел от 25 до 55: {sum}");
Console.ReadKey();
```

```
Сумма кубов чисел от 25 до 55: 2281600
```

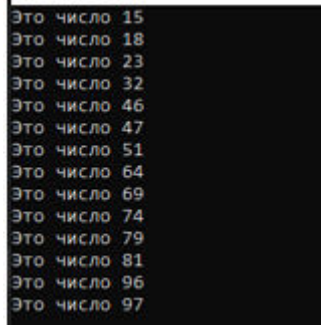
4. Среди двузначных чисел найти те, сумма квадратов цифр которых делится на 13.

```
double one, dec, sum;
for (int i = 10; i < 100; i++)
{
    one = i % 10;
    dec = i / 10;
```

```

sum = Math.Pow(one, 2) + Math.Pow(dec, 2);
if (sum % 13 == 0)
{
    Console.WriteLine($"Это число {i}");
}
}
Console.ReadKey();

```



```

Это число 15
Это число 18
Это число 23
Это число 32
Это число 46
Это число 47
Это число 51
Это число 64
Это число 69
Это число 74
Это число 79
Это число 81
Это число 96
Это число 97

```

5. Написать программу поиска двузначных чисел, таких, что если к сумме цифр этого числа прибавить квадрат этой суммы, то получится это число.

```

double one, dec, sum, sumK;
for (int i = 10; i < 100; i++)
{
    one = i % 10;
    dec = i / 10;
    sum = one + dec;
    sumK = sum + Math.Pow(sum, 2);
    if (sumK == i)
    {
        Console.WriteLine($"Это число {i}");
    }
}

Console.ReadKey();

```

```
Это число 12
Это число 42
Это число 90
```

6. Квадрат трехзначного числа оканчивается тремя цифрами, которые как раз и составляют это число. Написать программу поиска таких чисел.

```
double square, n;

for (int i = 100; i < 1000; i++)
{
    square = Math.Pow(i, 2);
    n = square % 1000;
    if (n == i)
    {
        Console.WriteLine($"Это число {i}");
    }
}

Console.ReadKey();
```

```
Это число 376
Это число 625
```

7. Написать программу поиска четырехзначного числа, которое при делении на 133 дает в остатке 125, а при делении на 134 дает в остатке 111.

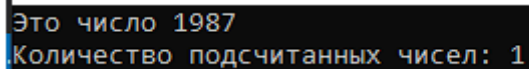
```
int count = 0;
for (int i = 1000; i < 10000; i++)
{
    if ((i % 133 == 125) && (i % 134 == 111))
    {
        Console.WriteLine($"Это число {i}");
        count += 1;
    }
}
```

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		37

```

}
Console.WriteLine($"Количество подсчитанных чисел: {count}");
Console.ReadKey();

```



```

Это число 1987
Количество подсчитанных чисел: 1

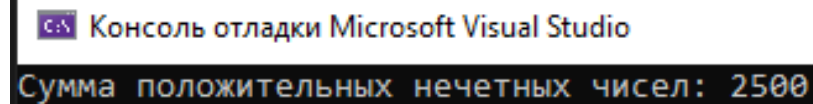
```

8. Найти сумму положительных нечетных чисел, меньших 100.

```

double sum = 0;
for (int i = 1; i < 100; i++)
{
    if (i % 2 == 1)
    {
        sum += i;
    }
}
Console.WriteLine($"Сумма положительных нечетных чисел:
{sum}");
Console.ReadKey();

```



```

Консоль отладки Microsoft Visual Studio
Сумма положительных нечетных чисел: 2500

```

8. Найти сумму целых положительных чисел из промежутка от А до В, кратных 4 (значения переменных А и В вводятся с клавиатуры).

```

int a, b, sum = 0;
Console.Write("Введите значение промежутка А (начало):");
a = int.Parse(Console.ReadLine());
Console.Write("Введите значение промежутка Б (конец):");
b = int.Parse(Console.ReadLine());
for (int i = a; i < b; i++)
{
    if (i % 4 == 0)

```

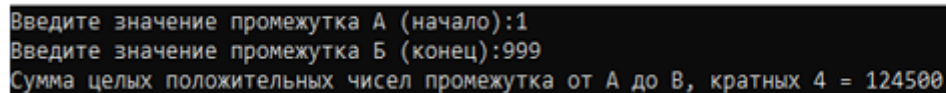
```

        {
            sum += i;
        }
    }

    Console.WriteLine($"Сумма целых положительных чисел
    промежутка от А до В, кратных 4 = {sum}");

    Console.ReadKey();

```



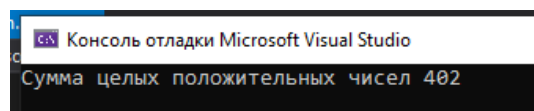
10. Найти сумму целых положительных чисел, больших 20, меньших 100, кратных 3 и заканчивающихся на 2, 4 или 8

```

int sum = 0;
for (int i = 21; i < 100; i++)
{
    if (i % 3 == 0 && (i % 10 == 2 || i % 10 == 4 || i % 10 == 8))
    {
        sum += i;
    }
}

Console.WriteLine($"Сумма целых положительных чисел
{sum}");

```



11. В трехзначном числе зачеркнули первую цифру слева, когда полученное двузначное число умножили на 7, то получили данное число. Найти это число.

```

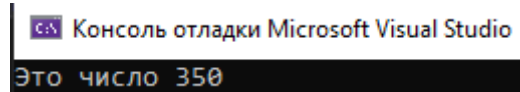
int n;
for (int i = 100; i < 1000; i++)
{
    n = i % 100;

```

```

        if (i == 7 * n)
        {
            Console.WriteLine($"Это число {i}");
        }
    }
}

```



12. Сумма цифр трехзначного числа кратна 7, само число также делится на 7. Найти все такие числа.

```

int sum, one, dec, hun;
for (int i = 100; i < 1000; i++)
{
    one = i % 10;
    dec = (i / 10) % 10;
    hun = i / 100;
    sum = one + dec + hun;
    if (sum % 7 == 0 && i % 7 == 0)
    {
        Console.WriteLine($"Это число = {i}");
    }
}
Console.ReadKey();

```



```
Консоль отладки M
Это число = 133
Это число = 266
Это число = 322
Это число = 329
Это число = 392
Это число = 399
Это число = 455
Это число = 511
Это число = 518
Это число = 581
Это число = 588
Это число = 644
Это число = 700
Это число = 707
Это число = 770
Это число = 777
Это число = 833
Это число = 966
```

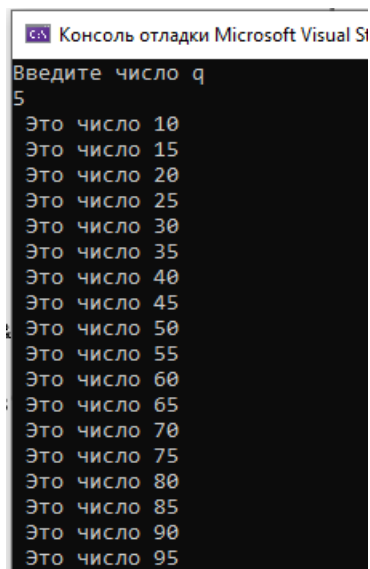
13. Среди четырехзначных чисел выбрать те, у которых все четыре цифры различны.

```
int one, dec, hun, th;
for (int i = 1000; i < 10000; i++)
{
    one = i % 10;
    dec = (i / 10) % 10;
    hun = (i / 100) % 10;
    th = i / 1000;
    if (th != one && th != dec && th != hun && hun != one && hun
!= dec && dec != one)
    {
        Console.WriteLine($"Это число = {i}");
    }
}
Console.ReadKey();
```

```
cs Консоль отладки Microsoft Visual S
Это число = 1023
Это число = 1024
Это число = 1025
Это число = 1026
Это число = 1027
Это число = 1028
Это число = 1029
Это число = 1032
Это число = 1034
Это число = 1035
Это число = 1036
Это число = 1037
Это число = 1038
Это число = 1039
Это число = 1042
Это число = 1043
Это число = 1045
Это число = 1046
Это число = 1047
Это число = 1048
```

14. Среди двузначных чисел найти те, сумма цифр которых равна n ($0 < n < 18$) и число делится без остатка на число q .

```
int sum = 0, one, dec, q;
Console.WriteLine("Введите число q");
q = int.Parse(Console.ReadLine());
for (int i = 10; i < 100; i++)
{
    one = i % 10;
    dec = i / 10;
    sum = one + dec;
    if ((0 < sum) && (18 > sum) && (i % q == 0))
    {
        Console.WriteLine($" Это число {i}");
    }
}
```



15. Одна штука некоторого товара стоит 20,4 руб. Напечатать таблицу стоимости 2, 3, ..., 20 штук этого товара.

```
double k = 20.4;
double proz;
Console.WriteLine("Одна штука некоторого товара стоит 20,4
руб");
for (int i = 2; i < 21; i++)
{
    proz = k * i;
    Console.WriteLine($"{i} штук этого товара стоит {proz:###}");
}
Console.ReadKey();
```

```
Консоль отладки Microsoft Visual Studio

Одна штука некоторого товара стоит 20,4 руб
2 штук этого товара стоит 40,8
3 штук этого товара стоит 61,2
4 штук этого товара стоит 81,6
5 штук этого товара стоит 102
6 штук этого товара стоит 122,4
7 штук этого товара стоит 142,8
8 штук этого товара стоит 163,2
9 штук этого товара стоит 183,6
10 штук этого товара стоит 204
11 штук этого товара стоит 224,4
12 штук этого товара стоит 244,8
13 штук этого товара стоит 265,2
14 штук этого товара стоит 285,6
15 штук этого товара стоит 306
16 штук этого товара стоит 326,4
17 штук этого товара стоит 346,8
18 штук этого товара стоит 367,2
19 штук этого товара стоит 387,6
20 штук этого товара стоит 408
```

16. Напечатать таблицу умножения на 7:

1 x 7 = 7

2 x 7 = 14

...

9 x 7 = 63

`int k = 7, proz;`

`Console.WriteLine("Таблица умножения на 7");`

`for (int i = 1; i < 10; i++)`

`{`

`proz = k * i;`

`Console.WriteLine($"{i} * {k} = {proz}");`

`}`

`Console.ReadKey();`

```

Консоль отладки Microsoft Vi:
Таблица умножения на 7
1 * 7 = 7
2 * 7 = 14
3 * 7 = 21
4 * 7 = 28
5 * 7 = 35
6 * 7 = 42
7 * 7 = 49
8 * 7 = 56
9 * 7 = 63

```

17. Рассчитать значения y для значений x , равных 4, 5, ..., 28:

$$y = 2t^2 + 5,5t - 2$$

$$t = x + 2$$

```

double y, t;
for (int x = 4; x < 29; x++)
{
    t = x + 2;
    y = 2 * Math.Pow(t, 2) + 5.5 * t - 2;
    Console.WriteLine($"При x = {x} y = {y}");
}
Console.ReadKey();

```

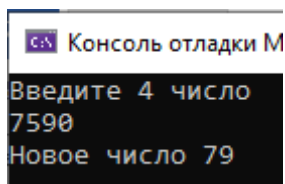
```

Консоль отладки Microsoft Visual Studio
При x = 6 y = 170
При x = 7 y = 209,5
При x = 8 y = 253
При x = 9 y = 300,5
При x = 10 y = 352
При x = 11 y = 407,5
При x = 12 y = 467
При x = 13 y = 530,5
При x = 14 y = 598
При x = 15 y = 669,5
При x = 16 y = 745
При x = 17 y = 824,5
При x = 18 y = 908
При x = 19 y = 995,5
При x = 20 y = 1087
При x = 21 y = 1182,5
При x = 22 y = 1282
При x = 23 y = 1385,5
При x = 24 y = 1493
При x = 25 y = 1604,5
При x = 26 y = 1720
При x = 27 y = 1839,5
При x = 28 y = 1963

```

18. Дано четырехзначное число n. Выбросить из записи числа n цифры 0 и 5, оставив прежним порядок остальных цифр. Например, из числа 1509 должно получиться 19.

```
int n, k = 0, l = 1, m;
Console.WriteLine("Введите четырехзначное число ");
n = int.Parse(Console.ReadLine());
m = n;
for (int i = 0; i < 4; i++)
{
    if (n % 10 != 0 && n % 10 != 5)
    {
        k = k + n % 10 * l;
        l *= 10;
    }
    n /= 10;
}
Console.WriteLine($"Новое число {k}");
```



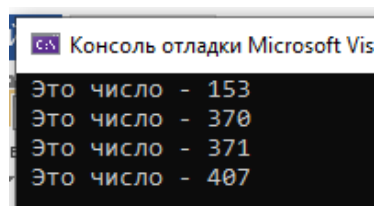
19. Натуральное число из n цифр является числом Армстронга, если сумма его цифр, возведенных в n-ю степень, равна самому числу (например, $153 = 1^3 + 5^3 + 3^3$). Получить все числа Армстронга, состоящие из трех и четырех цифр.

```
int one, dec, hun, th, m;
for (int i = 100; i < 9999; i++)
{
    m = i;
    one = i % 10;
    dec = (i / 10) % 10;
```

```

        hun = (i / 100) % 10;
        th = i / 1000;
        if ((Math.Pow(one, 3) + Math.Pow(dec, 3) + Math.Pow(hun, 3) +
Math.Pow(th, 3) == i))
        {
            Console.WriteLine($" Это число - {i}");
        }
    }
}

```



20. Дана последовательность из 20 целых чисел. Определить количество чисел в наиболее длинной подпоследовательности из подряд идущих нулей.

21. Дано натуральное число. Найти все его делители и их сумму.

```

int n, k = 0, sum = 0;
Console.WriteLine("Введите число ");
n = int.Parse(Console.ReadLine());
for (int i = 2; i <= n / 2; i++)
{
    if (n % i == 0)
    {
        Console.WriteLine($" Это делитель {i}");
        k++;
        sum += i;
    }
}
if (k == 0)

```

```

    {
        Console.WriteLine($" У этого числа есть только два делителя 1
и {n}");
    }
    Console.WriteLine($" Сумма делителей = {sum}");

```

```

Консоль отладки Microsoft V
Введите число
125
Это делитель 5
Это делитель 25
Сумма делителей = 30

```


Контрольная работа №3

Цикл с параметром

Необходимо решить три задачи.

Из первого задания необходимо решить номер по варианту и номер по варианту+10. Если такого варианта нет, отсчет продолжается с первого варианта. При оценивании работы будет учитываться вид ввода исходных данных и вывод результата. Должны присутствовать поясняющие надписи при вводе и выводе.

Из второго задания необходимо решить номер по варианту.

Работа рассчитана на 60 мин.

1. Среди трехзначных чисел найти те, сумма квадратов цифр которых делится на 18.

```
double one, dec, sum, hun;
for (int i = 100; i < 1000; i++)
{
    one = i % 10;
    dec = (i / 10) % 10;
    hun = i / 100;
    sum = Math.Pow(one, 2) + Math.Pow(dec, 2) + Math.Pow(hun,
2);

    if (sum % 18 == 0)
    {
        Console.WriteLine($"Это число {i}");
    }
}
Console.ReadKey();
```

```

Это число 581
Это число 600
Это число 606
Это число 633
Это число 639
Это число 660
Это число 666
Это число 693
Это число 699
Это число 712
Это число 721
Это число 745
Это число 754
Это число 778
Это число 787
Это число 815
Это число 822
Это число 848
Это число 851
Это число 877
Это число 884
Это число 903
Это число 909
Это число 930
Это число 936
Это число 963
Это число 969
Это число 990
Это число 996

```

2. Найти количество целых положительных чисел из промежутка от А до В, кратных 3 и оканчивающихся на 1 (значения переменных А и В вводятся с клавиатуры).

```

int i;
int one = 0;
Console.WriteLine("Введите начало отрезка А (целое и
положительное число):");
int otr_A = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Введите начало отрезка В (целое и
положительное число):");
int otr_B = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Введите шаг:");
int step = Convert.ToInt32(Console.ReadLine());
for (i = otr_A; i <= otr_B; i += step)
{

```

```

one = i % 10;

if (one == 1 && i % 3 == 0) {
    Console.WriteLine($"Найдено число {i}");
}

}

Console.ReadKey();

```

```

Найдено число 141
Найдено число 171
Найдено число 201
Найдено число 231
Найдено число 261
Найдено число 291
Найдено число 321
Найдено число 351
Найдено число 381
Найдено число 411
Найдено число 441
Найдено число 471
Найдено число 501
Найдено число 531
Найдено число 561
Найдено число 591
Найдено число 621
Найдено число 651
Найдено число 681
Найдено число 711
Найдено число 741
Найдено число 771
Найдено число 801
Найдено число 831
Найдено число 861
Найдено число 891
Найдено число 921
Найдено число 951
Найдено число 981

```

3. Составить программу для вычисления значений функции $F(x)$ на отрезке $[a, b]$ с шагом h . Результат представить в виде таблицы, первый столбец – значение аргумента, второй - соответствующее значение функции. Границы отрезка и шаг ввести с экрана. При написании формулы для вычисления функции использовать методы класса Math.

$$F(x) = x - \sin(x).$$

```

int i;

double fx = 0;

```

```

Console.WriteLine("Введите начало отрезка A:");
int otr_A = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Введите начало отрезка B:");
int otr_B = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Введите шаг:");
int step = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Функция: F(x) = x - sin(x)");
for (i = otr_A; i <= otr_B; i += step)
{
    fx = i - Math.Sin(i);
    if (i < 10)
    {
        Console.WriteLine($"x = {i} | F(x) = {fx:f2} |");
    }
    if(i > 10 && i < 99)
    {
        Console.WriteLine($"x = {i} | F(x) = {fx:f2} |");
    }
    if (i == 99)
    {
        Console.WriteLine($"x = {i} | F(x) = {fx:f2} |");
    }
    if (i > 99)
    {
        Console.WriteLine($"x = {i} | F(x) = {fx:f2} |");
    }
}
Console.ReadKey();

```

x = 222	F(x) = 221,13
x = 223	F(x) = 222,95
x = 224	F(x) = 224,81
x = 225	F(x) = 225,93
x = 226	F(x) = 226,19
x = 227	F(x) = 226,28
x = 228	F(x) = 227,03
x = 229	F(x) = 228,67
x = 230	F(x) = 230,62
x = 231	F(x) = 232,00
x = 232	F(x) = 232,46
x = 233	F(x) = 232,50
x = 234	F(x) = 233,00
x = 235	F(x) = 234,42
x = 236	F(x) = 236,37
x = 237	F(x) = 237,98
x = 238	F(x) = 238,69
x = 239	F(x) = 238,76
x = 240	F(x) = 239,05
x = 241	F(x) = 240,22
x = 242	F(x) = 242,10
x = 243	F(x) = 243,89
x = 244	F(x) = 244,86
x = 245	F(x) = 245,04
x = 246	F(x) = 245,18
x = 247	F(x) = 246,07
x = 248	F(x) = 247,82
x = 249	F(x) = 249,73
x = 250	F(x) = 250,97

Лабораторная работа 5-6

Тема: Циклы с постусловием и предусловием.

Задание 1. Задачу решить двумя способами с использованием двух видов цикла.

Постройте таблицу значений функции $y=f(x)$ для $x \in [a, b]$ с шагом h

$$11. y = \begin{cases} x^2 + 5, & \text{если } x \leq 5; \\ 0, & \text{если } 5 < x < 20; \\ 1, & \text{если } x \geq 20. \end{cases}$$

```
double y = 0;
```

```
Console.WriteLine("Введите начало отрезка A:");
```

```
int otr_A = Convert.ToInt32(Console.ReadLine());
```

```
Console.WriteLine("Введите начало отрезка B:");
```

```
int otr_B = Convert.ToInt32(Console.ReadLine());
```

```
Console.WriteLine("Введите шаг:");
```

```
int step = Convert.ToInt32(Console.ReadLine());
```

```
Console.WriteLine("Расчет у циклом while в табличном виде:");
```

```
while (otr_A <= otr_B)
```

```
{
```

```
    if (otr_A <= 5)
```

```
    {
```

```
        y = Math.Pow(otr_A, 2) + 5;
```

```
        if (otr_A >= 3 && otr_A <= 5)
```

```
        {
```

```
            Console.WriteLine($"| x = {otr_A} | y = {y} |");
```

```

    }
    else
    {
        Console.WriteLine($"| x = {otr_A} | y = {y} |");
    }
}

if (otr_A < 20 && otr_A > 5)
{
    y = 0;
    if (otr_A >= 10)
    {
        Console.WriteLine($"| x = {otr_A} | y = {y} |");
    }
    else
    {
        Console.WriteLine($"| x = {otr_A} | y = {y} |");
    }
}

if (otr_A >= 20)

{
    y = 1;
    Console.WriteLine($"| x = {otr_A} | y = {y} |");
}

otr_A += step;
} //while
Console.ReadKey();

```

Рассчет y циклом while в табличном виде:

x = 1	y = 6
x = 2	y = 9
x = 3	y = 14
x = 4	y = 21
x = 5	y = 30
x = 6	y = 0
x = 7	y = 0
x = 8	y = 0
x = 9	y = 0
x = 10	y = 0
x = 11	y = 0
x = 12	y = 0
x = 13	y = 0
x = 14	y = 0
x = 15	y = 0
x = 16	y = 0
x = 17	y = 0
x = 18	y = 0
x = 19	y = 0
x = 20	y = 1
x = 21	y = 1
x = 22	y = 1
x = 23	y = 1
x = 24	y = 1
x = 25	y = 1
x = 26	y = 1
x = 27	y = 1
x = 28	y = 1
x = 29	y = 1

```
double y = 0;
```

```
Console.WriteLine("Введите начало отрезка A:");
```

```
int otr_A = Convert.ToInt32(Console.ReadLine());
```

```
Console.WriteLine("Введите начало отрезка B:");
```

```
int otr_B = Convert.ToInt32(Console.ReadLine());
```

```
Console.WriteLine("Введите шаг:");
```

```
int step = Convert.ToInt32(Console.ReadLine());
```

```
Console.WriteLine("Рассчет y циклом do while в табличном  
виде:");
```

```
do
```



```

{
    if (otr_A <= 5)
    {
        y = Math.Pow(otr_A, 2) + 5;
        if (otr_A >= 3 && otr_A <= 5)
        {
            Console.WriteLine($"| x = {otr_A} | y = {y} |");
        }
        else
        {
            Console.WriteLine($"| x = {otr_A} | y = {y} |");
        }
    }

    if (otr_A < 20 && otr_A > 5)
    {
        y = 0;
        if (otr_A >= 10)
        {
            Console.WriteLine($"| x = {otr_A} | y = {y} |");
        }
        else
        {
            Console.WriteLine($"| x = {otr_A} | y = {y} |");
        }
    }

    if (otr_A >= 20)
    {
        y = 1;
    }

```

```

        Console.WriteLine($"| x = {otr_A}| y = {y} |");
    }

    otr_A += step;
}

while (otr_A <= otr_B);

Console.ReadKey();

```

```

Введите начало отрезка A:
1
Введите начало отрезка B:
99
Введите шаг:
1
Рассчет у циклом do while в табличном виде:
| x = 1 | y = 6 |
| x = 2 | y = 9 |
| x = 3 | y = 14 |
| x = 4 | y = 21 |
| x = 5 | y = 30 |
| x = 6 | y = 0 |
| x = 7 | y = 0 |
| x = 8 | y = 0 |
| x = 9 | y = 0 |
| x = 10 | y = 0 |
| x = 11 | y = 0 |
| x = 12 | y = 0 |
| x = 13 | y = 0 |
| x = 14 | y = 0 |
| x = 15 | y = 0 |
| x = 16 | y = 0 |
| x = 17 | y = 0 |
| x = 18 | y = 0 |
| x = 19 | y = 0 |
| x = 20 | y = 1 |
| x = 21 | y = 1 |
| x = 22 | y = 1 |
| x = 23 | y = 1 |

```

Задание 2. Задачу решить с использованием двух видов цикла.

Составить программу для вычисления значений функции $F(x)$ на отрезке

$[a, b]$ с шагом h . Результат представить в виде таблицы, первый столбец короткий — значение аргумента, второй — соответствующие значения функции.

$$11. F(x) = \sqrt{x} \cos^2 x.$$

```

double fx = 0;
Console.WriteLine("Введите начало отрезка A:");
int otr_A = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите начало отрезка B:");
int otr_B = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите шаг:");
int step = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Решение задачи с циклом while");

while (otr_A <= otr_B)
{
    fx = Math.Sqrt(otr_A) * Math.Pow(Math.Cos(otr_A), 2);
    if (otr_A < 10)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }

    if (otr_A > 10 && otr_A < 99)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }

    if (otr_A == 99)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }
}

```

```

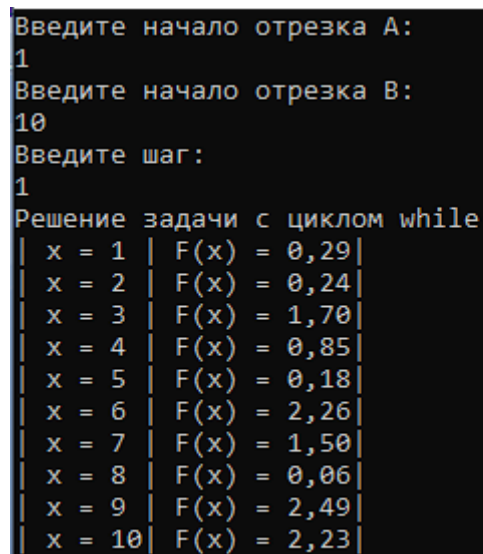
    }

    if (otr_A > 99)
    {
        Console.WriteLine($"x = {otr_A}|F(x) = {fx:f2}|");
    }

    otr_A += step;
}

Console.ReadKey();

```



```

Введите начало отрезка A:
1
Введите начало отрезка B:
10
Введите шаг:
1
Решение задачи с циклом while
| x = 1 | F(x) = 0,29 |
| x = 2 | F(x) = 0,24 |
| x = 3 | F(x) = 1,70 |
| x = 4 | F(x) = 0,85 |
| x = 5 | F(x) = 0,18 |
| x = 6 | F(x) = 2,26 |
| x = 7 | F(x) = 1,50 |
| x = 8 | F(x) = 0,06 |
| x = 9 | F(x) = 2,49 |
| x = 10 | F(x) = 2,23 |

```

```

double fx = 0;

Console.WriteLine("Введите начало отрезка A:");
int otr_A = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите начало отрезка B:");
int otr_B = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите шаг:");
int step = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Решение задачи с циклом do while");

```

```

do

{
    fx = Math.Sqrt(otr_A) * Math.Pow(Math.Cos(otr_A), 2);
    if (otr_A < 10)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }

    if (otr_A >= 10 && otr_A < 99)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }

    if (otr_A == 99)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }

    if (otr_A > 99)
    {
        Console.WriteLine($"x = {otr_A} | F(x) = {fx:f2}|");
    }

    otr_A += step;
} while (otr_A <= otr_B);
Console.ReadKey();

```

```

Введите начало отрезка A:
1
Введите начало отрезка B:
15
Введите шаг:
1
Решение задачи с циклом do while
| x = 1 | F(x) = 0,29 |
| x = 2 | F(x) = 0,24 |
| x = 3 | F(x) = 1,70 |
| x = 4 | F(x) = 0,85 |
| x = 5 | F(x) = 0,18 |
| x = 6 | F(x) = 2,26 |
| x = 7 | F(x) = 1,50 |
| x = 8 | F(x) = 0,06 |
| x = 9 | F(x) = 2,49 |
| x = 10 | F(x) = 2,23 |
| x = 11 | F(x) = 0,00 |
| x = 12 | F(x) = 2,47 |
| x = 13 | F(x) = 2,97 |
| x = 14 | F(x) = 0,07 |
| x = 15 | F(x) = 2,24 |

```

Задание 3. Решить все задачи двумя способами с использованием двух видов цикла while и do while

1. Напечатать "столбиком":

а) все целые числа от 20 до 35;

```
int i = 20;
```

```
int k = 36;
```

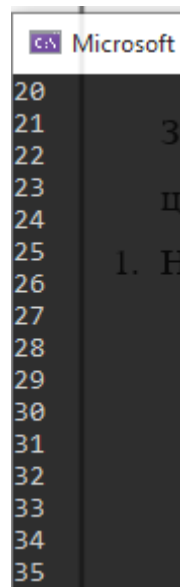
```
while (i < k)
```

```
{
```

```
    Console.WriteLine($"{i}");
```

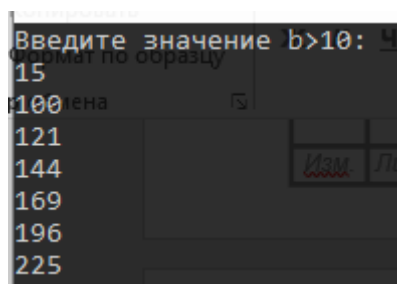
```
    i++;
```

```
}
```



б) квадраты всех целых чисел от 10 до b (значение b вводится с клавиатуры; $b > 10$);

```
int i = 10;
Console.WriteLine("Введите значение b>10:");
int b = Convert.ToInt32(Console.ReadLine());
while (i <= b)
{
    Console.WriteLine($"{i*i}");
    i++;
}
```



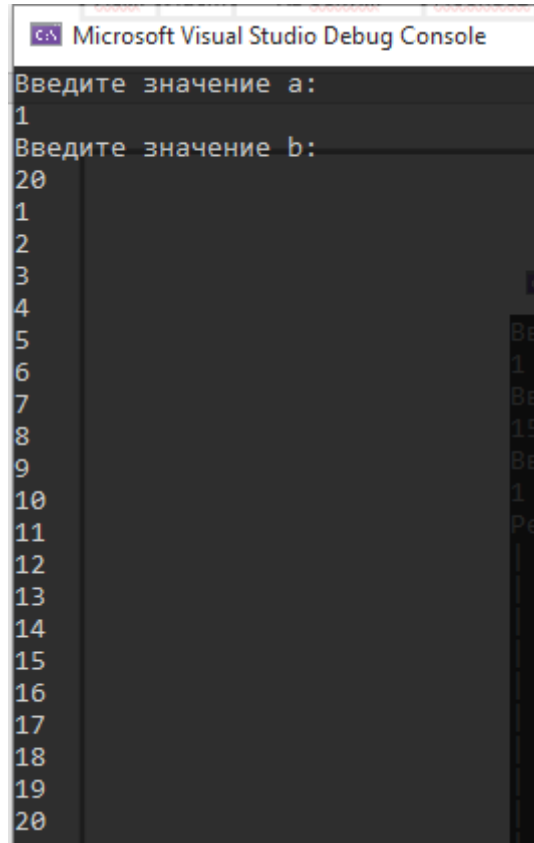
г) все целые числа от a до b (значения a и b вводятся с клавиатуры; $b > a$).

```
Console.WriteLine("Введите значение a:");
int a = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("Введите значение b:");
int b = Convert.ToInt32(Console.ReadLine());
while (a <= b)
```

```

{
    Console.WriteLine($"{a}");
    a++;
}

```



2. Одна штука некоторого товара стоит 20,4 руб. Напечатать таблицу стоимости 2, 3, 20 штук этого товара.

```

double k = 20.4;
double proz;
Console.WriteLine("Одна штука некоторого товара стоит 20,4
руб");
Console.WriteLine("_____");
int i = 1;
while (i < 21)
{
    proz = k * i;
    Console.WriteLine($"{i,6} | {proz,6:#.##}");
}

```



```

        i++;
    }
    Console.ReadKey();
}

```

Одна штука некоторого товара стоит 20,4 руб

1	20,4
2	40,8
3	61,2
4	81,6
5	102
6	122,4
7	142,8
8	163,2
9	183,6
10	204
11	224,4
12	244,8
13	265,2
14	285,6
15	306
16	326,4
17	346,8
18	367,2
19	387,6
20	408

3. Напечатать таблицу умножения на число n (значение n вводится с клавиатуры; $1 < n < 9$).

1 x n = ...

2 x n = ...

...

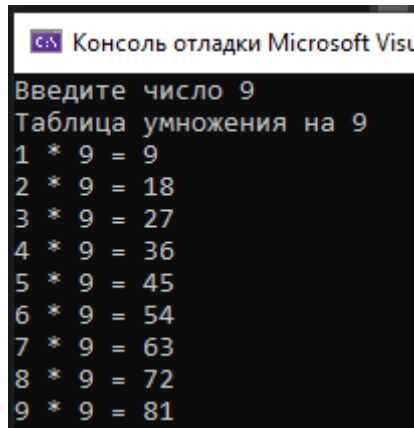
9 x n = ...

```

Console.Write("Введите число ");
int n = int.Parse(Console.ReadLine());
Console.WriteLine($"Таблица умножения на {n}");
int k = n, proz, i = 1;
while ( i < 10 )
{
    proz = k * i;
    Console.WriteLine($"{i} * {k} = {proz}");
    i++;
}

```

```
}
Console.ReadKey();
```



4. Найти все двузначные числа, в которых есть цифра N или само число делится на N.

```
int n, i = 10;
Console.WriteLine("Введите число n");
n = int.Parse(Console.ReadLine());
while (i < 100)
{
    if ((i / 10 == n) || (i % 10 == n))
    {
        Console.WriteLine($"Число {i} содержит число {n}");
    }
    if (i % n == 0)
    {
        Console.WriteLine($"Число {i} делится на число {n}");
    }
    i++;
}
Console.ReadKey();
```

```

cs Консоль отладки Microsoft Visual Studio
Введите число n
4
Число 12 делится на число 4
Число 14 содержит число 4
Число 16 делится на число 4
Число 20 делится на число 4
Число 24 содержит число 4
Число 24 делится на число 4
Число 28 делится на число 4
Число 32 делится на число 4
Число 34 содержит число 4
Число 36 делится на число 4
Число 40 содержит число 4
Число 40 делится на число 4
Число 41 содержит число 4

```

5. Определить количество трехзначных натуральных чисел, сумма цифр которых равна заданному числу N.

```

int n, k = 0, i = 100;
Console.WriteLine("Введите число ");
n = int.Parse(Console.ReadLine());
while (i < 1000)
{
    int one = i % 10;
    int dec = i / 10 % 10;
    int hun = i / 100;
    if (one + dec + hun == n)
    {
        k++;
    }
    i++;
}
Console.WriteLine($"Количество трехзначных натуральных чисел, сумма цифр которых равна заданному числу {n} = {k}");
Console.ReadKey();

```

```
Консоль отладки Microsoft Visual Studio
Введите число
12
Количество трехзначных натуральных чисел, сумма цифр которых равна заданному числу 12 = 66
```

6. Квадрат трехзначного числа оканчивается тремя цифрами, которые как раз и составляют это число. Написать программу поиска таких чисел.

```
double kon;

int i = 100;

while (i < 1000)
{
    kon = Math.Pow(i, 2) % 1000;

    if (i == kon)
    {
        Console.WriteLine($"Это число {i} ");
    }

    i++;
}

Console.ReadKey();
```

```
Консоль отладки Mi
Это число 376
Это число 625
```

7. Написать программу поиска четырехзначного числа, которое при делении на 133 дает в остатке 125, а при делении на 134 дает в остатке 111.

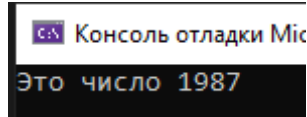
```
int i = 1000;

while (i < 10000)
{
    if ((i % 133 == 125) && (i % 134 == 111))
    {
        Console.WriteLine($"Это число {i} ");
    }
}
```

```

    }
    i++;
}
Console.ReadKey();

```



8. Найти сумму целых положительных чисел из промежутка от А до В, кратных 4 (значения переменных А и В вводятся с клавиатуры).

```

int b, a, sum = 0;

Console.WriteLine("Введите начальное ограничение a");
a = int.Parse(Console.ReadLine());
Console.WriteLine("Введите конечное ограничение b");
b = int.Parse(Console.ReadLine());
while (a <= b)
{
    if (a % 4 == 0)
    {
        sum = sum + a;
    }
    a++;
}

Console.WriteLine($"Сумма чисел от a до b = {sum}");
Console.ReadKey();

```

9. В трехзначном числе зачеркнули первую цифру слева, когда полученное двузначное число умножили на 7, то получили данное число. Найти это число.

```

int hun, i = 100;

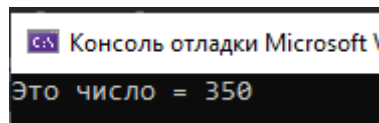
while (i < 1000)

```

```

{
    hun = i % 100;
    if (i == 7 * hun)
    {
        Console.WriteLine($"Это число = {i}");
    }
    i++;
}
Console.ReadKey();

```



10. Среди четырехзначных чисел выбрать те, у которых все четыре цифры различны.

```

int one, dec, hun, th, i = 1000;
while (i < 10000)
{
    one = i % 10;
    dec = (i / 10) % 10;
    hun = (i / 100) % 10;
    th = i / 1000;
    if (th != one && th != dec && th != hun && hun != one && hun
    != dec && dec != one)
    {
        Console.WriteLine($"Это число = {i}");
    }
    i++;
}
Console.ReadKey();

```

```
Консоль отладки Microsoft Visual S
Это число = 9846
Это число = 9847
Это число = 9850
Это число = 9851
Это число = 9852
Это число = 9853
Это число = 9854
Это число = 9856
Это число = 9857
Это число = 9860
Это число = 9861
Это число = 9862
Это число = 9863
Это число = 9864
```

11. Дано четырехзначное число n . Выбросить из записи числа n цифры 0 и 5, оставив прежним порядок остальных цифр. Например, из числа 1509 должно получиться 19.

```
int n, k = 0, l = 1, m;
Console.WriteLine("Введите четырехзначное число ");
n = int.Parse(Console.ReadLine());
m = n;
int i = 0;
while (i < 4)
{
    if (n % 10 != 0 && n % 10 != 5)
    {
        k = k + n % 10 * l;
        l *= 10;
    }
    n /= 10;
    i++;
}
Console.WriteLine($"Новое число {k}");
```

```

Консоль отладки Microsoft Visual Studio
Введите четырёхзначное число
1075
Новое число 17

```

12. Натуральное число из n цифр является числом Армстронга, если сумма его цифр, возведенных в n -ю степень, равна самому числу (например, $153=1^3+5^3+3^3$). Получить все числа Армстронга, состоящие из трех и четырех цифр.

```

int one, dec, hun, th, m, i = 1001;
while (i > 1000 && i < 10000)
{
    m = i;
    one = i % 10;
    dec = (i / 10) % 10;
    hun = (i / 100) % 10;
    th = i / 1000;
    if ((Math.Pow(one, 4) + Math.Pow(dec, 4) + Math.Pow(hun, 4) +
Math.Pow(th, 4) == i))
    {
        Console.WriteLine($" Это число - {i}");
    }
    i++;
}

```

```

Консоль отладки Microsoft Visual Studio
Это число - 1634
Это число - 8208
Это число - 9474

```

13. Дано натуральное число. Найти все его делители и их сумму.

```

int one, dec, hun, m, i = 101;
while (i > 100 && i < 1000)
{

```

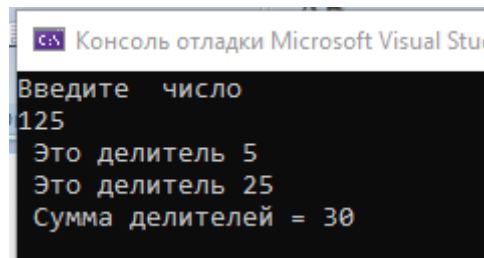


```

        m = i;
        one = i % 10;
        dec = (i / 10) % 10;
        hun = i / 100;
        if ((Math.Pow(one, 3) + Math.Pow(dec, 3) + Math.Pow(hun, 3) ==
i))

        {
            Console.WriteLine($" Это число - {i}");
        }
        i++;

```



Задание 4. Решить все задачи. При решении задач используйте один вид цикла while или do while. Какой именно решите сами.

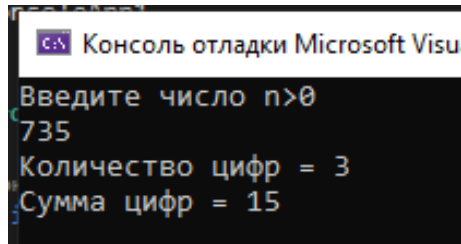
1. Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.

```

int n, k = 0, sum = 0, s;
Console.WriteLine("Введите число n>0");
n = int.Parse(Console.ReadLine());
while (n > 0)
{
    s = n % 10;
    n /= 10;
    sum = s + sum;
    k++;
}
Console.WriteLine($"Количество цифр = {k}");

```

```
Console.WriteLine($"Сумма цифр = {sum}");
```



2. Дано целое число $N (> 0)$. С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа N цифра «2». Если

имеется, то вывести true, если нет — вывести false.

```
Console.WriteLine("Введите число n>0");
```

```
int n = int.Parse(Console.ReadLine());
```

```
int m = n, M = n, k = 0;
```

```
while (m > 0)
```

```
{
```

```
    m %= 10;
```

```
    if (m == 2)
```

```
    {
```

```
        k++;
```

```
    }
```

```
    m = M;
```

```
    M /= 10;
```

```
}
```

```
if (k > 0)
```

```
{
```

```
    Console.WriteLine("True");
```

```
}
```

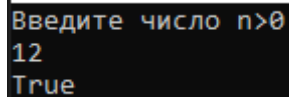
```
else
```

```
{
```

```
    Console.WriteLine("False");
```

```
}
```

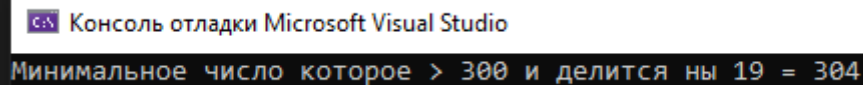
```
Console.ReadKey();
```



```
Введите число n>0  
12  
True
```

3. Найти минимальное число, большее 300, которое нацело делится на 19.

```
int m = 300;  
while (m >= 300 && m%19 != 0)  
{  
    m++;  
    if ( m % 19 == 0)  
    {  
        Console.WriteLine($"Минимальное число которое > 300 и  
делится на 19 = {m}");  
    }  
}  
Console.ReadKey();
```



```
Консоль отладки Microsoft Visual Studio  
Минимальное число которое > 300 и делится на 19 = 304
```

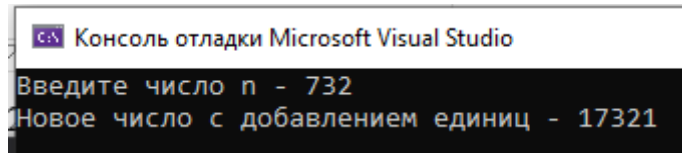
4. Приписать по 1 в начало и в конец записи числа n. Например, было n=3456, стало n=134561.

```
int n, k, x = 1, s;  
Console.Write("Введите число n - ");  
n = int.Parse(Console.ReadLine());  
k = n;  
do  
{  
    n /= 10;  
    x *= 10;  
}
```

```

while (n > 0);
s = (x + k)*10 + 1;
Console.WriteLine($"Новое число с добавлением единиц - {s}");
Console.ReadKey();

```



5. Поменять местами первую и последнюю цифры числа. Например, из числа 8547 должно быть получено число 7548.

```

int chislo, first, ten, hun, thous, result;
while (true) {
    Console.WriteLine("Введите четырёхзначное число:");
    chislo = Convert.ToInt32(Console.ReadLine());
    first = chislo % 10;
    ten = (chislo / 10) % 10;
    hun = (chislo / 100) % 10;
    thous = chislo / 1000;

    result = first * 1000 + hun * 100 + ten * 10 + thous;

    Console.WriteLine($"Ваше число - {chislo} \nЧисло,
образованное перестановкой первой и последней цифры - {result}");
    Console.WriteLine("Завершить работу? (Да, Нет)");
    string end = Console.ReadLine();
    if (end.Equals("Да") == true) {
        break;
    }
    else{
        continue;
    }
}

```

```

Microsoft Visual Studio Debug Console
Введите четырёхзначное число:
3489
Ваше число - 3489
Число, образованное перестановкой первой и последней цифры - 9483
Завершить работу? (Да, Нет)
Нет
4. Найти все двузначные числа, в которых есть цифра N или сам
Введите четырёхзначное число:
4589
Ваше число - 4589
Число, образованное перестановкой первой и последней цифры - 9584
Завершить работу? (Да, Нет)
Да
равна заданному числу N.

```

6. Приписать к исходному числу n такое же число. Например, из числа 1903 должно быть получено число 19031903.

```

string result;
while (true) {
    Console.WriteLine("Введите четырёхзначное число:");
    string chislo = Convert.ToString(Console.ReadLine());

    result = chislo + chislo;
    Console.WriteLine($"{result}");

    Console.WriteLine("Завершить работу? (Да, Нет)");
    string end = Console.ReadLine();
    if (end.Equals("Да") == true) {
        break;
    }
    else{
        continue;
    }
}

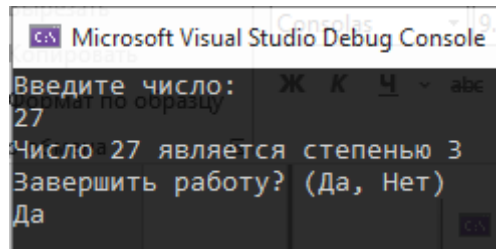
```

```
Введите четырёхзначное число:
1903
19031903
Завершить работу? (Да, Нет)
Нет
Введите четырёхзначное число:
1789
17891789
Завершить работу? (Да, Нет)
```

7. Определить, является ли заданное число степенью 3.

```
while (true) {
    Console.WriteLine("Введите число:");
    int chislo = Convert.ToInt32(Console.ReadLine());
    int i = 1;
    while (i < chislo) {
        i = i * 3;
        if (i == chislo)
        {
            Console.WriteLine($"Число {chislo} является степенью
3");
            break;
        }
    }

    Console.WriteLine("Завершить работу? (Да, Нет)");
    string end = Console.ReadLine();
    if (end.Equals("Да") == true) {
        break;
    }
    else{
        continue;
    }
}
```



8. Составить программу, проверяющую, является ли заданное натуральное число палиндромом, то есть таким, десятичная запись которого читается одинаково слева направо и справа налево.

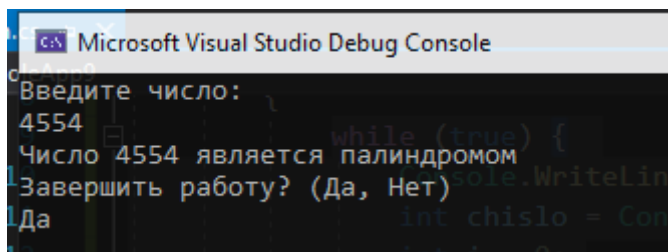
```
while (true) {
    Console.WriteLine("Введите число:");
    int chislo = Convert.ToInt32(Console.ReadLine());
    int i = 0;
    int tmp = chislo;
    while (tmp > 0) {
        i = i * 10 + tmp % 10;
        tmp /= 10;
        if (i == chislo)
        {
            Console.WriteLine($"Число {chislo} является
палиндромом");
            break;
        }
    }

    Console.WriteLine("Завершить работу? (Да, Нет)");
    string end = Console.ReadLine();
    if (end.Equals("Да") == true) {
        break;
    }
    else{
        continue;
    }
}
```

```

    }
}

```



9. Определить является ли введенное целое число простым.

```
Console.Write("Введите число n - ");
```

```
int n = int.Parse(Console.ReadLine());
```

```
int b = 1, k = 0;
```

```
while (b <= n)
```

```
{
```

```
    if (n % b == 0)
```

```
    {
```

```
        k++;
```

```
    }
```

```
    b++;
```

```
}
```

```
if (k == 2)
```

```
{
```

```
    Console.WriteLine($"{n} - это простое число");
```

```
}
```

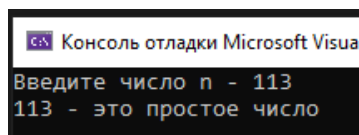
```
else
```

```
{
```

```
    Console.WriteLine($"{n} это не простое число");
```

```
}
```

```
Console.ReadKey();
```



10. Дано целое число $N (> 1)$. Найти наименьшее целое число K , при котором выполняется неравенство $3K > N$.

```
while (true) {
    Console.WriteLine("Введите число:");
    int n = Convert.ToInt32(Console.ReadLine());
    int k = 1;
    while (3*k <= n) {
        k++;
    }
    Console.WriteLine($"Найти наименьшее целое число K, при
котором выполняется неравенство  $3K > N$ : {k}");

    Console.WriteLine("Завершить работу? (Да, Нет)");
    string end = Console.ReadLine();
    if (end.Equals("Да") == true) {
        break;
    }
    else{
        continue;
    }
}
```

```
Microsoft Visual Studio Debug Console
Введите число: 35
Найти наименьшее целое число K, при котором выполняется неравенство  $3K > N$ : 12
Завершить работу? (Да, Нет) Нет
Введите число: 10
Найти наименьшее целое число K, при котором выполняется неравенство  $3K > N$ : 4
Завершить работу? (Да, Нет) Да
```

11. Дано целое число $N (> 1)$. Вывести наименьшее из целых чисел K , для которых сумма $1 + 2 + \dots + K$ будет больше или равна N , и саму эту сумму.

Изм.	Лист	№ докум.	Подпись	Дата

```

Console.Write("Введите N:");

int n = Convert.ToInt32(Console.ReadLine());

int k = 0;
int sum = 0;
while (sum <= n)
{
    k++;
    sum = ((1 + k)/2) * k;
    if (sum >= n)
    {
        sum = sum - k;
        k--;
        Console.WriteLine($"Сумма: {sum} \nНаименьшее из целых
чисел К: {k}");
        break;
    }
}
Console.ReadKey();

```

```

Введите N:10
Сумма: 10
Наименьшее из целых чисел К: 4

```

12. Дано число $A (> 1)$. Вывести наименьшее из целых чисел K , для которых сумма $1 + 1/2 + \dots + 1/K$ будет больше A , и саму эту сумму.

```

while (true) {
    Console.WriteLine("Введите число:");
    double a = Convert.ToDouble(Console.ReadLine());
    int k = 0;
    double tmp = 0;

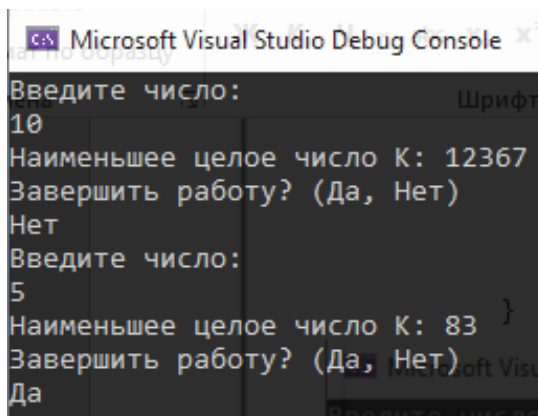
```

```

while (tmp <= a) {
    k++;
    tmp = tmp + 1 / (double)k;
}
Console.WriteLine($"Наименьшее целое число К: {k}");

Console.WriteLine("Завершить работу? (Да, Нет)");
string end = Console.ReadLine();
if (end.Equals("Да") == true) {
    break;
}
else{
    continue;
}
}

```



13. Дано целое число $N (> 0)$. Используя операции деления нацело и взятия остатка от деления, вывести все его цифры, начиная с самой правой (разряда единиц).

```

while (true) {
    Console.WriteLine("Введите число:");

    int n = Convert.ToInt32(Console.ReadLine());
}

```

```

while (n > 0) {
    Console.WriteLine($"{n%10}");
    n /= 10;
}

Console.WriteLine("Завершить работу? (Да, Нет)");
string end = Console.ReadLine();
if (end.Equals("Да") == true) {
    break;
}
else{
    continue;
}
}

```

```

c:\ Microsoft Visual Studio Debug Console
Введите число:
10
0
1
Завершить работу? (Да, Нет)
Нет
Введите число:
59
9
5
Завершить работу? (Да, Нет)
Да

```

14. Совершенным числом называется число, равное сумме всех своих делителей, меньших, чем оно само. Например: $6=1+2+3$, $28=1+2+4+7+14$. Древним грекам были известны только четыре первых числа. Составить программу, проверяющую, является ли заданное натуральное число совершенным.

```

while (true) {
    Console.WriteLine("Введите число:");
    int n = Convert.ToInt32(Console.ReadLine());
}

```

```

int i = 1;
int sum = 0;
while (i < n) {
    if (n % i == 0)
    {
        sum += i;
    }
    if (n == sum){
        Console.WriteLine($"Число {n} совершенное");
        break;
    }
    i++;
}

Console.WriteLine("Завершить работу? (Да, Нет)");
string end = Console.ReadLine();
if (end.Equals("Да") == true) {
    break;
}
else{
    continue;
}
}

```

```

Введите число:
6
Число 6 совершенное
Завершить работу? (Да, Нет)
Нет
Введите число:
28
Число 28 совершенное
Завершить работу? (Да, Нет)
-

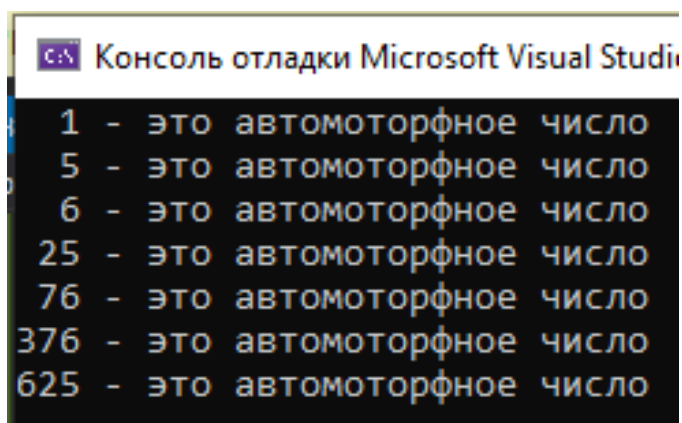
```

15. Автоморфным называется такое число, которое равно последним цифрам своего квадрата. Например: $5^2=25$, $25^2=625$. Очевидно, что автоморфные числа должны оканчиваться либо на 1, либо на 5, либо на 6. Составить программу нахождения автоморфных чисел (с учетом приведенного факта), не превышающих значения 999.

```
int a = 1;

while (a <= 999)
{
    if (a == Math.Pow(a, 2) % 10 || a == Math.Pow(a, 2) % 100 || a ==
Math.Pow(a, 2) % 1000)
    {
        Console.WriteLine($"{a,3} - это автомоторфное число");
    }
    a++;
}

Console.ReadKey();
```



```
Консоль отладки Microsoft Visual Studio
1 - это автомоторфное число
5 - это автомоторфное число
6 - это автомоторфное число
25 - это автомоторфное число
76 - это автомоторфное число
376 - это автомоторфное число
625 - это автомоторфное число
```

16. Кубические автоморфные числа равны последним цифрам своих кубов. Например: $6^3=216$. Верно ли, что и такие числа должны оканчиваться либо на 1, либо на 5, либо на 6? С учетом этого факта составить программу нахождения двузначных, трехзначных кубических автоморфных чисел.

```
int a = 1;

while (a <= 999)
{
```

```

        if (a == Math.Pow(a,3) % 10 || a == Math.Pow(a,3) % 100 || a ==
Math.Pow(a,3) % 1000)
        {
            Console.WriteLine($"{a,3} - это кубическое автомоторфное
число");
        }
        a++;
    }
    Console.ReadKey();

```

Консоль отладки Microsoft Visual Studio

```

1 - это кубическое автомоторфное число
4 - это кубическое автомоторфное число
5 - это кубическое автомоторфное число
6 - это кубическое автомоторфное число
9 - это кубическое автомоторфное число
24 - это кубическое автомоторфное число
25 - это кубическое автомоторфное число
49 - это кубическое автомоторфное число
51 - это кубическое автомоторфное число
75 - это кубическое автомоторфное число
76 - это кубическое автомоторфное число
99 - это кубическое автомоторфное число
125 - это кубическое автомоторфное число
249 - это кубическое автомоторфное число
251 - это кубическое автомоторфное число
375 - это кубическое автомоторфное число
376 - это кубическое автомоторфное число
499 - это кубическое автомоторфное число
501 - это кубическое автомоторфное число
624 - это кубическое автомоторфное число
625 - это кубическое автомоторфное число
749 - это кубическое автомоторфное число
751 - это кубическое автомоторфное число
875 - это кубическое автомоторфное число
999 - это кубическое автомоторфное число

```

Контрольная работа №4

Циклы с предусловием и постусловием

Необходимо решить три задачи.

Из задания 1 необходимо решить номер по варианту.

Из задания 2 необходимо решить номер по варианту и номер по варианту+10. Если такого варианта нет, отсчет продолжается с первого варианта. При оценивании работы будет учитываться вид ввода исходных данных и вывод результата. Должны присутствовать поясняющие надписи при вводе и выводе.

Работа рассчитана на 60 мин.

Критерии оценки.

Выполнено одно задание, на вопросы преподавателя по заданию даны положительные ответы – оценка удовлетворительно.

Выполнены два задания, на вопросы преподавателя по заданиям даны положительные ответы – оценка хорошо.

Выполнены три задания, на вопросы преподавателя по заданиям даны положительные ответы – оценка отлично.

Задание 1. Постройте таблицу значений функции $y=f(x)$ для $x \in [a, b]$ с шагом h

Границы диапазона и шаг изменения аргумента ввести с клавиатуры.

Задания под четным вариантом необходимо выполнить, используя цикл с предусловием, а нечетные варианты, используя цикл с постусловием.

$$y = \begin{cases} 1, & \text{если } x = 1 \text{ или } x = -1; \\ \frac{-1}{1-x}, & \text{если } x \geq 0 \text{ и } x \neq 1; \\ \frac{1}{1+x}, & \text{если } x < 0 \text{ и } x \neq 1. \end{cases}$$

`double` y = 0;

`Console.WriteLine("Введите начало отрезка A:");`

`double` otr_A = Convert.ToDouble(Console.ReadLine());


```

Console.WriteLine("Введите начало отрезка B:");
double otr_B = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите шаг:");
int step = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Расчет у циклом do while в табличном
виде:");

do
{
    if (otr_A == 1 || otr_A == -1)
    {
        y = 1;
        Console.WriteLine($"x = {otr_A,2}|y = {y,2:f2}");
        otr_A += step;
    }

    if (otr_A >= 0 && otr_A != 1)
    {
        y = -1 / (1 - otr_A);
        Console.WriteLine($"x = {otr_A,2}|y = {y,2:f2}");
        otr_A += step;
    }

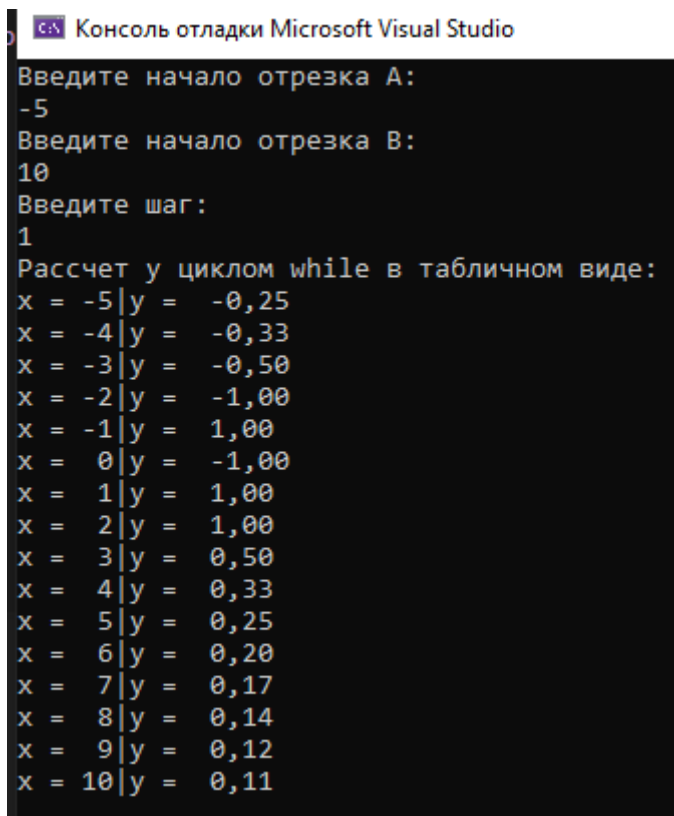
    if (otr_A < 0 && otr_A != 1)
    {
        y = 1 / (1 + otr_A);
        Console.WriteLine($"x = {otr_A,2}|y = {y,2:f2}");
        otr_A += step;
    }
}

```

```
}
```

```
} while (otr_A <= otr_B);
```

```
Console.ReadKey();
```



```
Консоль отладки Microsoft Visual Studio
Введите начало отрезка A:
-5
Введите начало отрезка B:
10
Введите шаг:
1
Расчет у циклом while в табличном виде:
x = -5 | y = -0,25
x = -4 | y = -0,33
x = -3 | y = -0,50
x = -2 | y = -1,00
x = -1 | y = 1,00
x = 0 | y = -1,00
x = 1 | y = 1,00
x = 2 | y = 1,00
x = 3 | y = 0,50
x = 4 | y = 0,33
x = 5 | y = 0,25
x = 6 | y = 0,20
x = 7 | y = 0,17
x = 8 | y = 0,14
x = 9 | y = 0,12
x = 10 | y = 0,11
```

Задание 2

1. Натуральное число из n цифр является числом Армстронга, если сумма его цифр, возведенных в n -ю степень, равна самому числу (например, $153=1^3+5^3+3^3$).
2. Получить все числа Армстронга, состоящие из трех цифр.

```
int one, dec, hun, m, i = 100;
```

```
do
```

```
{
```

```
    m = i;
```

```
    one = i % 10;
```

```
    dec = (i / 10) % 10;
```

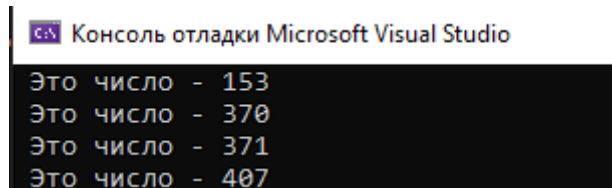
```
    hun = (i / 100) % 10;
```

```

i))

    {
        Console.WriteLine($" Это число - {i}");
    }
    i++;
} while (i < 1000);

```



2. Дано число A (> 1). Вывести наименьшее из целых чисел K , для которых сумма $1 + 1/2 + \dots + 1/K$ будет больше A , и саму эту сумму.

```

do
{
    Console.WriteLine("Введите число:");
    double a = Convert.ToDouble(Console.ReadLine());
    int k = 0;
    double tmp = 0;
    do
    {
        k++;
        tmp = tmp + 1 / (double)k;
    } while (tmp <= a);
    Console.WriteLine($"Наименьшее целое число K: {k}");

    Console.WriteLine("Завершить работу? (Да, Нет)");
    string end = Console.ReadLine();

    if (end.Equals("Да") == true)

```

```

{
    break;
}
else
{
    continue;
}
} while (true);

```

```

Введите число:
10
Наименьшее целое число K: 12367
Завершить работу? (Да, Нет)
Нет
Введите число:
5
Наименьшее целое число K: 83
Завершить работу? (Да, Нет)
Нет
Введите число:
2
Наименьшее целое число K: 4
Завершить работу? (Да, Нет)
_

```

Лабораторная работа №7

Тема: Одномерные массивы.

Цель: Научиться реализовывать алгоритмы для работы с одномерными массивами средствами Visual Studio, используя консольное приложение.

Задание 1. Составить программу для решения задачи: Задан массив X. Рассчитать значения элементов массива Y согласно заданному варианту (см. таблицу 1). Вывести на экран массивы X и Y (в столбец). Найти максимальный и минимальный элементы массива Y, а также вычислить среднее арифметическое элементов массива Y.

Вариант	Формула для расчета y_i	Массив X
1	$y_i = \frac{0.8 \cos^2(x_i + 2x_i^2)}{\sqrt[4]{0.21x_i + 2.3}}$	2.3, 3, 3.85, 4.2, 5.61, 6.12, 7.5, 8.1

```
double[] x = {2.3, 3, 3.85, 4.2, 5.61, 6.12, 7.5, 8.1};
```

```
double[] y = new double[8];
```

```
double y_i;
```

```
double summ_y = 0;
```

```
for (int i = 0; i < 8; i++)
```

```
{
```

```
    y_i = 0.8 * Math.Pow(Math.Cos(x[i] + 2 * Math.Pow(x[i], 2)), 2) /
```

```
    Math.Pow(0.21 * x[i] + 2.3, 1/4);
```

```
    Console.WriteLine($"|x = {x[i],2:f1 }|y = {y_i,2:f2 }|");
```

```
    y.SetValue(y_i, i); // запись в массив
```

```
}
```

```
foreach (int a in y)
```

```

    {
        summ_y += y[a]; // сумма элементов массива y
    }

    Array.Sort(y); // сортировка массива
    double maxValue_y = y[^1]; //максимальное значение
    Console.WriteLine($"Максимальное значение массива y:
{maxValue_y:f2}");

    double minValue_y = y[0]; //минимальное значение
    Console.WriteLine($"Минимальное значение массива y:
{minValue_y:f2}");

    Console.WriteLine($"Сумма элементов массива y: {summ_y}");
    double sr_ar = summ_y / 8;
    Console.WriteLine($"Среднее арифметическое элементов
массива y: {sr_ar}");

```

```

Консоль отладки Microsoft Visual Studio
x = 2,3 | y = 0,72 |
x = 3,0 | y = 0,24 |
x = 3,9 | y = 0,19 |
x = 4,2 | y = 0,03 |
x = 5,6 | y = 0,57 |
x = 6,1 | y = 0,50 |
x = 7,5 | y = 0,53 |
x = 8,1 | y = 0,17 |
Максимальное значение массива y: 0,72
Минимальное значение массива y: 0,03
Сумма элементов массива y: 5,790846242964985
Среднее арифметическое элементов массива y: 0,7238557803706231

```

Задание 2. Задания выполнить с применением целочисленного одномерного массива. Размерность массива вводить с клавиатуры. Способ заполнения массива выбрать самостоятельно.

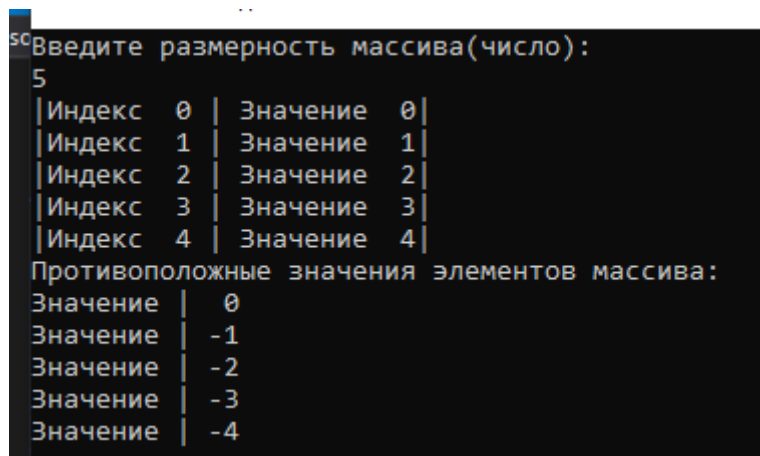
1) заменить все положительные элементы противоположными им числами;

```
Console.WriteLine("Введите размерность массива(число:");
int n = Convert.ToInt32(Console.ReadLine());
int[] A = Enumerable.Range(0, n).ToArray();

foreach (int i in A)
{
    Console.WriteLine($"|Индекс {i,2} | Значение {A[i],2}|");
}

Console.WriteLine($"Противоположные значения элементов
массива:");

foreach (int i in A)
{
    A[i] = -A[i];
    Console.WriteLine($"Значение | {A[i], 2}");
}
```



```
sc Введите размерность массива(число):
5
|Индекс 0 | Значение 0|
|Индекс 1 | Значение 1|
|Индекс 2 | Значение 2|
|Индекс 3 | Значение 3|
|Индекс 4 | Значение 4|
Противоположные значения элементов массива:
Значение | 0
Значение | -1
Значение | -2
Значение | -3
Значение | -4
```

2) заменить все элементы, меньшие заданного числа, этим числом;

```
Console.WriteLine("Введите размерность массива(число:");
int n = Convert.ToInt32(Console.ReadLine());
int[] A = Enumerable.Range(0, n).ToArray();
foreach (int i in A)
```

```

{
    Console.WriteLine($"|Индекс {i,2} | Значение {A[i],2}|");
}
Console.WriteLine("Введите число:");
int ch = Convert.ToInt32(Console.ReadLine());
Console.WriteLine($"Элементы, меньшие заданного числа,
заменённые этим числом:");
foreach (int i in A)
{
    if (A[i] < ch)
    {
        A[i] = ch;
    }
    Console.WriteLine($"Новый элемент(возможно) | {A[i]}");
}

```

Консоль отладки Microsoft Visual Studio

```

Введите размерность массива(число):
10
|Индекс 0 | Значение 0|
|Индекс 1 | Значение 1|
|Индекс 2 | Значение 2|
|Индекс 3 | Значение 3|
|Индекс 4 | Значение 4|
|Индекс 5 | Значение 5|
|Индекс 6 | Значение 6|
|Индекс 7 | Значение 7|
|Индекс 8 | Значение 8|
|Индекс 9 | Значение 9|
Введите число:
5
Элементы, меньшие заданного числа, заменённые этим числом:
Новый элемент(возможно) | 5
Новый элемент(возможно) | 5
Новый элемент(возможно) | 5
Новый элемент(возможно) | 5
Новый элемент(возможно) | 5
Новый элемент(возможно) | 5
Новый элемент(возможно) | 6
Новый элемент(возможно) | 7
Новый элемент(возможно) | 8
Новый элемент(возможно) | 9

```

- 1) заменить все элементы, попадающие в интервал $[a, b]$, нулем;


```

int[] A;

int m = 0;
Console.Write("Введите число a - ");
int a = int.Parse(Console.ReadLine());
Console.Write("Введите число b - ");
int b = int.Parse(Console.ReadLine());
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem >= a && elem <= b)
    {
        Console.Write($"{m} ");
    }
    else
    {
        Console.Write($"{elem} ");
    }
}

```

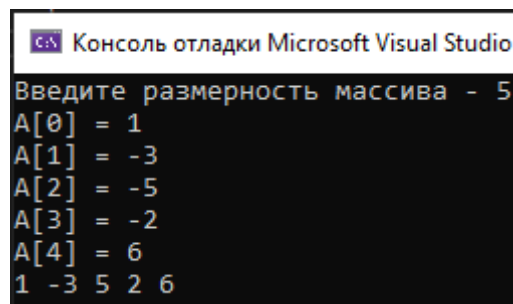
```

c:\ Консоль отладки Microsoft Visual Studio
Введите число a - 1
Введите число b - 4
Введите размерность массива - 5
A[0] = 2
A[1] = 3
A[2] = -1
A[3] = 8
A[4] = 4
0 0 -1 8 0

```

2) заменить все отрицательные элементы, не кратные 3, противоположными им числами;

```
int[] A;  
int y;  
Console.Write("Введите размерность массива - ");  
int n = int.Parse(Console.ReadLine());  
A = new int[n];  
for (int i = 0; i < n; i++)  
{  
    Console.Write($"A[{i}] = ");  
    A[i] = int.Parse(Console.ReadLine());  
}  
foreach (int elem in A)  
{  
    if (elem < 0 && elem % 3 != 0)  
    {  
        y = elem * -1;  
        Console.Write($" {y} ");  
    }  
    else  
    {  
        Console.Write($" {elem} ");  
    }  
}
```



```
Консоль отладки Microsoft Visual Studio  
Введите размерность массива - 5  
A[0] = 1  
A[1] = -3  
A[2] = -5  
A[3] = -2  
A[4] = 6  
1 -3 5 2 6
```

3) все элементы, меньшие заданного числа, увеличить в два раза;

```
int[] A;
```

```

int y;
Console.Write("Введите число - ");
int ch = int.Parse(Console.ReadLine());
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem < ch)
    {
        y = elem * 2;
        Console.Write($"{y} ");
    }
    else
    {
        Console.Write($"{elem} ");
    }
}

```

```

c:\> Консоль отладки Microsoft Visual Studio
Введите число - 5
Введите размерность массива - 5
A[0] = 1
A[1] = 2
A[2] = 7
A[3] = 8
A[4] = 5
2 4 7 8 5

```

4) подсчитать среднее арифметическое элементов;

```
int[] A;
```

```

double y = 0;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    y += elem;
}
double sr = y / n;
Console.Write($"Среднее арифметическое элементов массива -
{sr:f1} ");

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 4
A[0] = 8
A[1] = 4
A[2] = 2
A[3] = 2
Среднее арифметическое элементов массива - 4

```

- 5) подсчитать среднее арифметическое отрицательных элементов;

```

int[] A;
int z = 0;
double y = 0;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{

```

```

        Console.Write($"A[{i}] = ");
        A[i] = int.Parse(Console.ReadLine());
    }
    foreach (int elem in A)
    {
        if (elem < 0)
        {
            y += elem;
            z++;
        }
    }
    double sr = y / z;
    Console.Write($"Среднее арифметическое элементов массива -
{sr:f1} ");

```

```

cs Консоль отладки Microsoft Visual Studio
Введите размерность массива - 5
A[0] = 17
A[1] = 8
A[2] = -1
A[3] = -6
A[4] = -3
Среднее арифметическое элементов массива - -3,3

```

6) подсчитать количество нечетных элементов;

```

int[] A;
int z = 0;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}

```

```

foreach (int elem in A)
{
    if (elem % 2 != 0)
    {
        z++;
    }
}

Console.Write($"Количество нечетных элементов массива = {z}");

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 7
A[0] = 1
A[1] = 2
A[2] = 3
A[3] = 7
A[4] = 9
A[5] = 4
A[6] = 2
Количество нечетных элементов массива = 4

```

7) подсчитать сумму элементов, попадающих в заданный интервал;

```

int[] A;

int sum = 0;

Console.Write("Введите число a - ");

int a = int.Parse(Console.ReadLine());

Console.Write("Введите число b - ");

int b = int.Parse(Console.ReadLine());

Console.Write("Введите размерность массива - ");

int n = int.Parse(Console.ReadLine());

A = new int[n];

for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}

```

```

foreach (int elem in A)
{
    if (elem >= a && elem <= b)
    {
        sum += elem;
    }
}

Console.WriteLine($"Сумма элементов массива попадающих в
интервал [{a};{b}] = {sum} ");

```

```

Консоль отладки Microsoft Visual Studio
Введите число a - 3
Введите число b - 9
Введите размерность массива - 5
A[0] = 1
A[1] = 2
A[2] = 12
A[3] = 5
A[4] = 7
Сумма элементов массива попадающих в интервал [3;9] = 12

```

8) подсчитать сумму элементов, кратных 9;

```

int[] A;

int sum = 0;

Console.WriteLine("Введите размерность массива - ");

int n = int.Parse(Console.ReadLine());

A = new int[n];

for (int i = 0; i < n; i++)
{
    Console.WriteLine($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}

foreach (int elem in A)
{
    if (elem % 9 == 0)
    {

```

```

        sum += elem;
    }
}

Console.WriteLine($"Сумма элементов массива кратных 9 = {sum} ");

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 5
A[0] = 1
A[1] = 9
A[2] = 18
A[3] = 14
A[4] = 16
Сумма элементов массива кратных 9 = 27

```

9) подсчитать количество элементов, не попадающих в заданный интервал;

```

int[] A;
int sum = 0;
Console.WriteLine("Введите число a - ");
int a = int.Parse(Console.ReadLine());
Console.WriteLine("Введите число b - ");
int b = int.Parse(Console.ReadLine());
Console.WriteLine("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.WriteLine($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem <= a || elem >= b)
    {
        sum += elem;
    }
}

```



```

    }
}

```

```

Console.Write($"Сумма элементов массива не попадающих в
интервал [{a};{b}] = {sum} ");

```

```

cs Консоль отладки Microsoft Visual Studio
Введите число a - 3
Введите число b - 7
Введите размерность массива - 5
A[0] = 1
A[1] = 2
A[2] = 5
A[3] = 6
A[4] = 10
Сумма элементов массива не попадающих в интервал [3;7] = 13

```

10) подсчитать сумму квадратов четных элементов;

```

int[] A;

int sum = 0;

Console.Write("Введите размерность массива - ");

int n = int.Parse(Console.ReadLine());

A = new int[n];

for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}

foreach (int elem in A)
{
    if (elem % 2 == 0)
    {
        sum += elem*elem;
    }
}

Console.Write($"Сумма квадратов четных элементов массива =
{sum} ");

```

```
Консоль отладки Microsoft Visual Studio
Введите размерность массива - 6
A[0] = 10
A[1] = 3
A[2] = 5
A[3] = 2
A[4] = 10
A[5] = 6
Сумма квадратов четных элементов массива = 240
```

11) вывести на экран номера всех элементов, больших заданного числа;

```
int[] A;
int u = 0;
Console.Write("Введите число a - ");
int a = int.Parse(Console.ReadLine());
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem > a)
    {
        Console.WriteLine($"Номера элемента большего a - A[{u}]");
    }
    u++;
}
```

```
Консоль отладки Microsoft Visual Studio
Введите число а - 7
Введите размерность массива - 5
A[0] = 12
A[1] = 13
A[2] = 3
A[3] = 4
A[4] = 16
Номера элемента большего а - A[0]
Номера элемента большего а - A[1]
Номера элемента большего а - A[4]
```

12) вывести на экран номера всех нечетных элементов;

```
int[] A;
int u = 0;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem % 2 != 0)
    {
        Console.WriteLine($"Номера четного элемента массива -
A[{u}] ");
    }
    u++;
}
```

13) вывести на экран номера всех элементов, которые не делятся на 7;

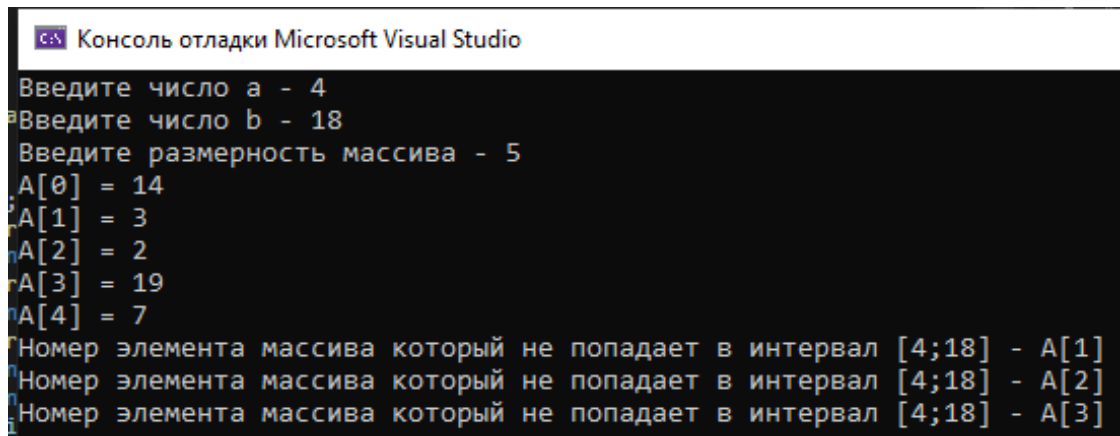
```
int[] A;  
int u = 0;  
Console.Write("Введите размерность массива - ");  
int n = int.Parse(Console.ReadLine());  
A = new int[n];  
for (int i = 0; i < n; i++)  
{  
    Console.Write($"A[{i}] = ");  
    A[i] = int.Parse(Console.ReadLine());  
}  
foreach (int elem in A)  
{  
    if (elem % 7 != 0)  
    {  
        Console.WriteLine($"Номера элемента массива который не  
делится на 7 - A[{u}] ");  
    }  
    u++;  
}
```

```
Консоль отладки Microsoft Visual Studio
Введите размерность массива - 6
A[0] = 2
A[1] = 14
A[2] = 6
A[3] = 7
A[4] = 21
A[5] = 9
Номера элемента массива который не делится на 7 - A[0]
Номера элемента массива который не делится на 7 - A[2]
Номера элемента массива который не делится на 7 - A[5]
```

14) вывести на экран номера всех элементов, не попадающих в заданный интервал;

```
int[] A;
int u = 0;
Console.Write("Введите число a - ");
int a = int.Parse(Console.ReadLine());
Console.Write("Введите число b - ");
int b = int.Parse(Console.ReadLine());
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem < a || elem > b)
    {
        Console.WriteLine($"Номер элемента массива который не
попадает в интервал [{a};{b}] - A[{u}] ");
        u++;
    }
}
```

u++;



```
Консоль отладки Microsoft Visual Studio
Введите число a - 4
Введите число b - 18
Введите размерность массива - 5
A[0] = 14
A[1] = 3
A[2] = 2
A[3] = 19
A[4] = 7
Номер элемента массива который не попадает в интервал [4;18] - A[1]
Номер элемента массива который не попадает в интервал [4;18] - A[2]
Номер элемента массива который не попадает в интервал [4;18] - A[3]
```

15) определить, является ли произведение элементов трехзначным числом;

```
int[] A;
int proz = 1;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    proz *= elem;
}
if (proz >= 100 && proz < 1000)
{
    Console.WriteLine($"Произведение элементов массива
является трёхзначным числом, оно равно {proz}");
}
else
```

```

{
    Console.WriteLine($"Произведение элементов массива не
является трёхзначным числом, оно равно {proz}");

```

Консоль отладки Microsoft Visual Studio

```

Введите размерность массива - 3
A[0] = 12
A[1] = 10
A[2] = 3
Произведение элементов массива является трёхзначным числом, оно равно 360

```

16) определить, является ли сумма элементов двухзначным числом;

```

int[] A;

int sum = 0;

Console.Write("Введите размерность массива - ");

int n = int.Parse(Console.ReadLine());

A = new int[n];

for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}

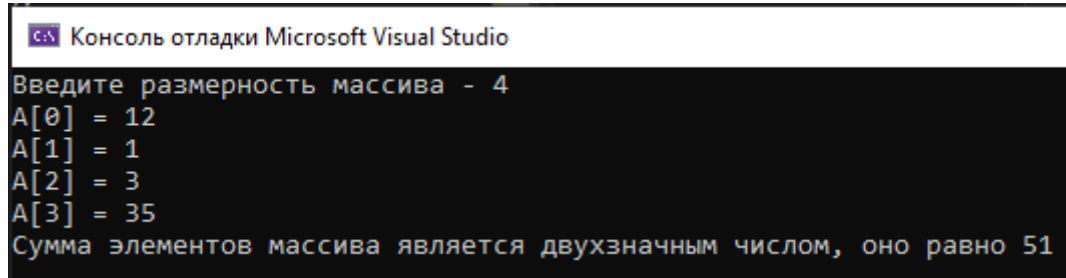
foreach (int elem in A)
{
    sum += elem;
}

if (sum >= 10 && sum < 100)
{
    Console.WriteLine($"Сумма элементов массива является
двухзначным числом, оно равно {sum}");
}

else
{

```

Console.WriteLine(\$"Сумма элементов массива не является
двухзначным числом, оно равно {sum}");



```
Консоль отладки Microsoft Visual Studio
Введите размерность массива - 4
A[0] = 12
A[1] = 1
A[2] = 3
A[3] = 35
Сумма элементов массива является двухзначным числом, оно равно 51
```

17) вывести на экран элементы с четными индексами (для
двумерного массива сумма индексов должна быть четной);

```
int[] A;
int u = 0;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    u++;
    if (u % 2 != 0 && u > 1)
    {
        Console.WriteLine($"Элементы массива с четными  
индексами {elem}");
    }
}
```



```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 8
A[0] = 1
A[1] = 2
A[2] = 3
A[3] = 4
A[4] = 5
A[5] = 6
A[6] = 7
A[7] = 8
Элементы массива с четными индексами 3
Элементы массива с четными индексами 5
Элементы массива с четными индексами 7

```

18) вывести на экран положительные элементы с нечетными индексами (для двумерного массива первый индекс должен быть нечетным).

```

int[] A;
int u = 0;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    u++;
    if (u % 2 == 0 && u != 0 && elem > 0)
    {
        Console.WriteLine($"Положительные элементы массива с
нечетными индексами {elem}");
    }
}

```

```
Консоль отладки Microsoft Visual Studio
Введите размерность массива - 8
A[0] = 1
A[1] = -2
A[2] = 3
A[3] = 4
A[4] = 5
A[5] = -6
A[6] = 7
A[7] = 8
Положительные элементы массива с нечетными индексами 4
Положительные элементы массива с нечетными индексами 8
```

Задание 3. Задания выполнить с применением целочисленного одномерного массива. Размерность массива вводить с клавиатуры. Способ заполнения массива выбрать самостоятельно.

- 1) подсчитать количество максимальных элементов;

`int[] A;`

`Random rnd = new Random();`

`Console.Write("Введите размерность массива:");`

`int n = int.Parse(Console.ReadLine());`

`Console.Write("Введите верхнюю границу заполнения: ");`

`int n_1 = int.Parse(Console.ReadLine());`

`Console.Write("Введите нижнюю границу заполнения: ");`

`int n_2 = int.Parse(Console.ReadLine());`

`A = new int[n];`

`for (int i = 0; i < n; i++)`

`{`

`if (n_1 < n_2)`

`{`

`A[i] = rnd.Next(n_1, n_2);`

`}`

`else`

`{`

`A[i] = rnd.Next(n_2, n_1);`

```

    }
    Console.WriteLine($"{i} элемент массива = {A[i]} \n");
}
Array.Sort(A);
int max = A[0];
int maxi = 0;
for (int k = 0; k < n; k++)
{
    if (A[k] > max)
    {
        max = A[k];
        maxi = k;
    }

}
int l = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] == max)
    {
        l++;
    }
}
Console.WriteLine($"Количество максимальных элементов: {l}");

```

```
Консоль отладки Microsoft Visual Studio
Введите размерность массива:20
Введите верхнюю границу заполнения: 250
Введите нижнюю границу заполнения: -250
0 элемент массива = 7
1 элемент массива = 86
2 элемент массива = -117
3 элемент массива = -64
4 элемент массива = -188
5 элемент массива = 180
6 элемент массива = -44
7 элемент массива = -4
8 элемент массива = -97
9 элемент массива = 246
10 элемент массива = 24
11 элемент массива = -247
12 элемент массива = 45
13 элемент массива = 112
14 элемент массива = 72
15 элемент массива = -140
16 элемент массива = -26
17 элемент массива = -139
18 элемент массива = -59
19 элемент массива = 118
Количество максимальных элементов: 1
```

2) вывести на экран номера всех минимальных элементов;

```
int[] A;
Random rnd = new Random();
Console.Write("Введите размерность массива:");
int n = int.Parse(Console.ReadLine());
Console.Write("Введите верхнюю границу заполнения: ");
int n_1 = int.Parse(Console.ReadLine());
Console.Write("Введите нижнюю границу заполнения: ");
int n_2 = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    if (n_1 < n_2)
    {
        A[i] = rnd.Next(n_1, n_2);
    }
}
```

```

else
{
    A[i] = rnd.Next(n_2, n_1);
}
Console.Write($"{i} элемент массива = {A[i]} \n");

}

Array.Sort(A);
int min = A[0];
int mini = 0;
for (int k = 0; k < n; k++)
{

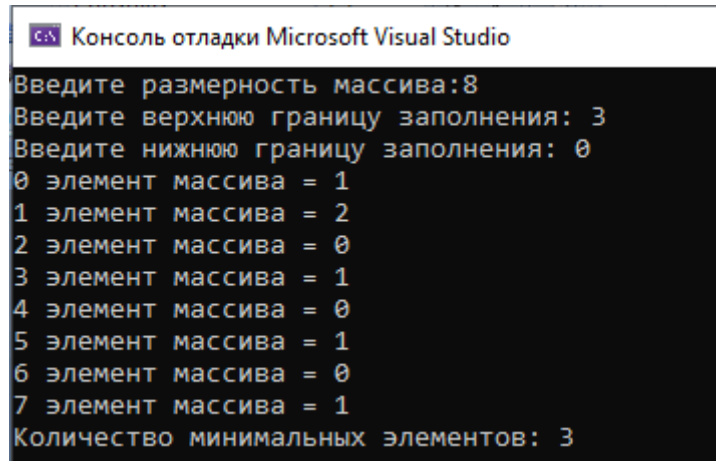
    if (A[k] < min)
    {
        min = A[k];
        mini = k;
    }

}
int l = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] == min)
    {
        l++;
    }
}
Console.WriteLine($"Количество минимальных элементов: {l}");

```

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		117

}



```
Консоль отладки Microsoft Visual Studio
Введите размерность массива: 8
Введите верхнюю границу заполнения: 3
Введите нижнюю границу заполнения: 0
0 элемент массива = 1
1 элемент массива = 2
2 элемент массива = 0
3 элемент массива = 1
4 элемент массива = 0
5 элемент массива = 1
6 элемент массива = 0
7 элемент массива = 1
Количество минимальных элементов: 3
```

- 3) заменить все максимальные элементы нулями;

```
int[] A;

Random rnd = new Random();

Console.Write("Введите размерность массива:");
int n = int.Parse(Console.ReadLine());

Console.Write("Введите верхнюю границу заполнения: ");
int n_1 = int.Parse(Console.ReadLine());

Console.Write("Введите нижнюю границу заполнения: ");
int n_2 = int.Parse(Console.ReadLine());

A = new int[n];

for (int i = 0; i < n; i++)
{
    if (n_1 < n_2)
    {
        A[i] = rnd.Next(n_1, n_2);
    }
    else
    {
        A[i] = rnd.Next(n_2, n_1);
    }
}
```

```

        Console.Write($"{i} элемент массива = {A[i]} \n");
    }

    Array.Sort(A);
    int max = A[0];
    int maxi = 0;
    for (int k = 0; k < n; k++)
    {

        if (A[k] > max)
        {
            max = A[k];
            maxi = k;
        }

    }
    for (int i = 0; i < n; i++)
    {
        if (A[i] == max)
        {
            A[i] = 0;
        }
    }
    Console.WriteLine("-----");
    for (int i = 0; i < n; i++)
    {
        Console.Write($"{i} элемент массива = {A[i]} \n");
    }

```

4) заменить все минимальные элементы на противоположные;

```
int[] A;
```

```

Random rnd = new Random();
Console.Write("Введите размерность массива:");
int n = int.Parse(Console.ReadLine());
Console.Write("Введите верхнюю границу заполнения: ");
int n_1 = int.Parse(Console.ReadLine());
Console.Write("Введите нижнюю границу заполнения: ");
int n_2 = int.Parse(Console.ReadLine());
A = new int[n];

for (int i = 0; i < n; i++)
{
    if (n_1 < n_2)
    {
        A[i] = rnd.Next(n_1, n_2);
    }
    else
    {
        A[i] = rnd.Next(n_2, n_1);
    }
    Console.Write($"{i} элемент массива = {A[i]} \n");
}

Array.Sort(A);
int min = A[0];
int mini = 0;
for (int k = 0; k < n; k++)
{

    if (A[k] < min)
    {

```



```

        min = A[k];
        mini = k;
    }

}

for (int i = 0; i < n; i++)
{
    if (A[i] == min)
    {
        A[i] = A[i] * -1;
    }
}

Console.WriteLine("-----");
for (int i = 0; i < n; i++)
{
    Console.Write($"{i} элемент массива = {A[i]} \n");
}

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива:15
Введите верхнюю границу заполнения: 1
Введите нижнюю границу заполнения: 15
0 элемент массива = 3
1 элемент массива = 12
2 элемент массива = 12
3 элемент массива = 12
4 элемент массива = 1
5 элемент массива = 6
6 элемент массива = 4
7 элемент массива = 3
8 элемент массива = 8
9 элемент массива = 12
10 элемент массива = 6
11 элемент массива = 3
12 элемент массива = 10
13 элемент массива = 6
14 элемент массива = 4
-----
0 элемент массива = -1
1 элемент массива = 3
2 элемент массива = 3
3 элемент массива = 3
4 элемент массива = 4
5 элемент массива = 4
6 элемент массива = 6
7 элемент массива = 6
8 элемент массива = 6
9 элемент массива = 8
10 элемент массива = 10
11 элемент массива = 12
12 элемент массива = 12
13 элемент массива = 12
14 элемент массива = 12

```

5) поменять местами максимальный элемент и первый;

```
int[] A;
```

```
Random rnd = new Random();
```

```
Console.Write("Введите размерность массива:");
```

```
int n = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите верхнюю границу заполнения: ");
```

```
int n_1 = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите нижнюю границу заполнения: ");
```

```
int n_2 = int.Parse(Console.ReadLine());
```

```
A = new int[n];
```

```

for (int i = 0; i < n; i++)
{
    if (n_1 < n_2)
    {
        A[i] = rnd.Next(n_1, n_2);
    }
    else
    {
        A[i] = rnd.Next(n_2, n_1);
    }
    Console.Write($"{i} элемент массива = {A[i]} \n");
}

Array.Sort(A);
int max = A[0];
int maxi = 0;
for (int k = 0; k < n; k++)
{

    if (A[k] > max)
    {
        max = A[k];
        maxi = k;
    }

}
int tmp = 0;
int first = A[0];
for (int i = 0; i < n; i++)
{

```

```

        if (A[i] == max)
        {
            tmp = first;
            first = A[i];
            A[i] = tmp;
        }
    }
    Console.WriteLine("-----");
    for (int i = 0; i < n; i++)
    {
        Console.Write($"{i} элемент массива = {A[i]} \n");
    }

```

Консоль отладки Microsoft Visual Studio

```

Введите размерность массива:10
Введите верхнюю границу заполнения: 1
Введите нижнюю границу заполнения: 5
0 элемент массива = 1
1 элемент массива = 3
2 элемент массива = 2
3 элемент массива = 4
4 элемент массива = 3
5 элемент массива = 1
6 элемент массива = 2
7 элемент массива = 3
8 элемент массива = 3
9 элемент массива = 2
-----
0 элемент массива = 1
1 элемент массива = 1
2 элемент массива = 2
3 элемент массива = 2
4 элемент массива = 2
5 элемент массива = 3
6 элемент массива = 3
7 элемент массива = 3
8 элемент массива = 3
9 элемент массива = 1

```

б) вывести на экран номера всех элементов, не совпадающих с максимальным;

using System;

```

namespace ConsoleApp5
{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();

            int[] A;

            Console.Write("Введите размерность массива - ");
            int n = int.Parse(Console.ReadLine());
            Console.Write("Введите нижнюю границу массива - ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите верхнюю границу массива - ");
            int b = int.Parse(Console.ReadLine());

            A = new int[n];
            for (int i = 0; i < n; i++)
            {
                A[i] = rnd.Next(a, b);
                Console.Write($"{A[i]} ");
            }

            int max = A[0];
            int maxi = 0;
            for (int i = 0; i < n; i++)
            {
                if (A[i] > max)
                {
                    max = A[i];
                    maxi = i;
                }
            }
        }
    }
}

```

```

    }
    Console.WriteLine(" ");
    Console.Write($"Номера элементов, не совпадающих с
максимальным: ");
    int k = 0;
    int y = A[0];
    for (int i = 0; i < n; i++)
    {
        if (A[i] != max)
        {
            Console.Write($" {k} ");
        }
        k++;
    }
    Console.ReadKey();
}
}
}

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 6
Введите нижнюю границу массива - 2
Введите верхнюю границу массива - 5
4 4 4 3 3 3
Номера элементов, не совпадающих с максимальным: 3 4 5

```

7) найти номер первого минимального элемента;

using System;

namespace ConsoleApp5

{

class Program

{

static void Main(string[] args)

```

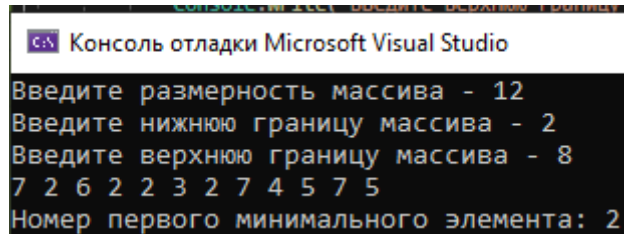
{
    Random rnd = new Random();
    int[] A;
    Console.Write("Введите размерность массива - ");
    int n = int.Parse(Console.ReadLine());
    Console.Write("Введите нижнюю границу массива - ");
    int a = int.Parse(Console.ReadLine());
    Console.Write("Введите верхнюю границу массива - ");
    int b = int.Parse(Console.ReadLine());
    A = new int[n];
    for (int i = 0; i < n; i++)
    {
        A[i] = rnd.Next(a, b);
        Console.Write($"{A[i]} ");
    }
    int min = A[0];
    int maxi = 0;
    for (int i = 0; i < n; i++)
    {
        if (A[i] < min)
        {
            min = A[i];
            maxi = i;
        }
    }
    Console.WriteLine(" ");
    Console.Write($"Номер первого минимального элемента: ");
    for (int i = 0; i < n; i++)
    {
        if (A[i] < min)

```

```

        {
            min = A[i];
            break;
        }
    }
    Console.WriteLine($"{min} ");
    Console.ReadKey();
}
}
}

```



8) найти номер последнего максимального элемента;

`using System;`

`namespace ConsoleApp5`

```

{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();
            int[] A;
            Console.WriteLine("Введите размерность массива - ");
            int n = int.Parse(Console.ReadLine());
            Console.WriteLine("Введите нижнюю границу массива - ");
            int a = int.Parse(Console.ReadLine());
            Console.WriteLine("Введите верхнюю границу массива - ");

```



```

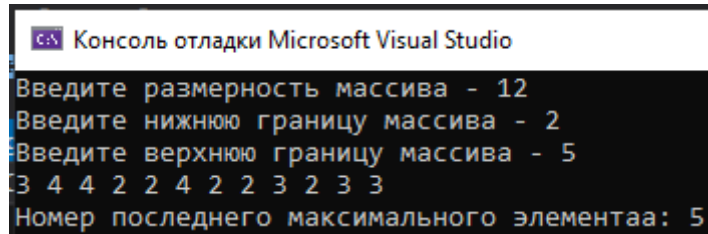
int b = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    A[i] = rnd.Next(a, b);
    Console.Write($"{A[i]} ");
}
int max = A[0];
int maxi = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] > max)
    {
        max = A[i];
        maxi = i;
    }
}
Console.WriteLine(" ");
Console.Write($"Номер последнего максимального элемента: ");
int k = 0;
int t = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] >= max)
    {
        t = k;
    }
    k++;
}
Console.Write($"{t} ");

```

```

        Console.ReadKey();
    }
}

```



9) подсчитать сумму элементов, расположенных между максимальным и минимальным элементами (минимальный и максимальный элементы в массиве единственные); если максимальный элемент встречается позже минимального, то выдать сообщение об этом;

```
using System;
```

```
namespace ConsoleApp5
```

```

{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();
            int[] A;
            int sum = 0;
            Console.Write("Введите размерность массива - ");
            int n = int.Parse(Console.ReadLine());
            Console.Write("Введите нижнюю границу массива - ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите верхнюю границу массива - ");
            int b = int.Parse(Console.ReadLine());
            A = new int[n];

```

```

for (int i = 0; i < n; i++)
{
    A[i] = rnd.Next(a, b);
    Console.Write($"{A[i]} ");
}
int max = A[0];
int maxi = 0;
int min = A[0];
int mini = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] > max)
    {
        max = A[i];
        maxi = i;
    }
    if (A[i] < min)
    {
        min = A[i];
        mini = i;
    }
}
if (maxi < mini)
{
    Console.WriteLine("\nОшибка, максимальный элемент
встречается после минимального");
    return;
}
for (int i = mini + 1; i < maxi; ++i)
{

```

```

        sum += A[i];
    }
    Console.WriteLine(" ");
    Console.Write($"Сумма элементов между максимальным и
МИНИМАЛЬНЫМ: {sum}");
    Console.ReadKey();
}
}
}

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 8
Введите нижнюю границу массива - 2
Введите верхнюю границу массива - 9
4 5 6 2 5 3 3 7
Сумма элементов между максимальным и минимальным: 11

```

10) найти номер первого максимального элемента;

`using System;`

`namespace ConsoleApp5`

```

{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();

            int[] A;
            Console.Write("Введите размерность массива - ");
            int n = int.Parse(Console.ReadLine());
            Console.Write("Введите нижнюю границу массива - ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите верхнюю границу массива - ");

```

```

int b = int.Parse(Console.ReadLine());
A = new int[n];
for (int i = 0; i < n; i++)
{
    A[i] = rnd.Next(a, b);
    Console.Write($"{A[i]} ");
}
int max = A[0];
int maxi = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] > max)
    {
        max = A[i];
        maxi = i;
    }
}
Console.WriteLine(" ");
Console.Write($"Номер первого максимального элемента: ");
for (int i = 0; i < n; i++)
{
    if (A[i] > max)
    {
        max = A[i];
        maxi = i;
        break;
    }
}
Console.Write($"{maxi} ");
Console.ReadKey();

```

```

    }
}
}

```

Консоль отладки Microsoft Visual Studio

```

Введите размерность массива - 12
Введите нижнюю границу массива - 2
Введите верхнюю границу массива - 8
6 4 2 7 6 4 7 7 3 5 3 4
Номер первого максимального элемента: 3

```

11) найти номер последнего минимального элемента;

```
using System;
```

```
namespace ConsoleApp5
```

```
{
```

```
class Program
```

```
{
```

```
static void Main(string[] args)
```

```
{
```

```
Random rnd = new Random();
```

```
int[] A;
```

```
Console.Write("Введите размерность массива - ");
```

```
int n = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите нижнюю границу массива - ");
```

```
int a = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите верхнюю границу массива - ");
```

```
int b = int.Parse(Console.ReadLine());
```

```
A = new int[n];
```

```
for (int i = 0; i < n; i++)
```

```
{
```

```
A[i] = rnd.Next(a, b);
```

```
Console.Write($"{A[i]} ");
```

```
}
```

```
}
```

```

int min = A[0];
int maxi = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] < min)
    {
        min = A[i];
        maxi = i;
    }
}
Console.WriteLine(" ");
Console.WriteLine($"Номер последнего минимального элемента: ");
int k = 0;
int t = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] == min)
    {
        t = k;
    }
    k++;
}
Console.WriteLine($"{t} ");
Console.ReadKey();
}
}
}

```

12) подсчитать сумму элементов, расположенных между первым максимальным и последним минимальными элементами; если максимальный элемент встречается позже минимального, то выдать сообщение об этом;

`using System;`

`namespace ConsoleApp5`

`{`

`class Program`

`{`

`static void Main(string[] args)`

`{`

`int[] A;`

`int sum = 0;`

`Console.Write("Введите количество элементов массива: ");`

`int n = int.Parse(Console.ReadLine());`

`A = new int[n];`

`Console.WriteLine("Введите левую границу");`

`int C = int.Parse(Console.ReadLine());`

`Console.WriteLine("Введите правую границу");`

`int D = int.Parse(Console.ReadLine());`

`Random rand = new Random();`

`Console.Write("Массив: ");`

`for (int i = 0; i < n; i++)`

`{`

`A[i] = rand.Next(C, D);`

`Console.Write($"{ A[i]} ");`


```

    }
    int max = A[0];
    int maxi = 0;
    int min = A[0];
    int mini = 0;
    for (int i = 0; i < n; i++)
    {
        if (A[i] < min)
        {
            min = A[i];
            mini = i;
        }
        if (A[i] > max)
        {
            max = A[i];
            maxi = i;
        }
    }
    if (maxi < mini)
    {
        Console.WriteLine("\nОшибка, минимального элемент
встречается после максимальный");
        return;
    }
    for (int i = mini + 1; i < maxi; ++i)
    {
        sum += A[i];
    }
    Console.WriteLine(" ");

```

```

        Console.WriteLine($"Сумма элементов между минимальным и
максимальным: {sum}");
        Console.ReadKey();
    }
}
}

```

```

Консоль отладки Microsoft Visual Studio
Введите количество элементов массива: 15
Введите левую границу
1
Введите правую границу
10
Массив: 6 2 6 2 4 1 7 3 5 5 5 9 7 8 2
Сумма элементов между минимальным и максимальным: 25

```

13) поменять местами первый минимальный и последний максимальный элементы;

```
using System;
```

```
namespace ConsoleApp5
```

```
{
```

```
class Program
```

```
{
```

```
static void Main(string[] args)
```

```
{
```

```
Random rnd = new Random();
```

```
int[] A;
```

```
Console.WriteLine("Введите размерность массива - ");
```

```
int n = int.Parse(Console.ReadLine());
```

```
Console.WriteLine("Введите нижнюю границу массива - ");
```

```
int a = int.Parse(Console.ReadLine());
```

```
Console.WriteLine("Введите верхнюю границу массива - ");
```

```
int b = int.Parse(Console.ReadLine());
```

```
A = new int[n];
```

```

for (int i = 0; i < n; i++)
{
    A[i] = rnd.Next(a, b);
    Console.Write($"{A[i]} ");
}
int min = A[0];
int mini = 0;
for (int i = 0; i < n; i++)
{
    if (A[i] < min)
    {
        min = A[i];
        mini = i;
    }
}
Console.WriteLine(" ");
Console.Write($"Первый минимальный элемент: ");
for (int i = 0; i < n; i++)
{
    if (A[i] < min)
    {
        min = A[i];
        break;
    }
}
Console.Write($"{min} ");
int max = A[0];
int maxi = 0;
for (int i = 0; i < n; i++)
{

```

```

        if (A[i] < max)
        {
            max = A[i];
            maxi = i;
        }
    }
    Console.WriteLine(" ");
    Console.WriteLine($"Последний максимальный элемент: ");
    for (int i = 0; i < n; i++)
    {
        if (A[i] > max)
        {
            max = A[i];
        }
    }
    Console.WriteLine($"{max} ");
    int y = min;
    int x = max;
    for (int i = 0; i < n; i++)
    {
        if (A[i] == min)
        {
            A[i] = min;
        }
        if (A[i] == max)
        {
            A[i] = max;
        }
    }
    Console.WriteLine(" ");

```

```

        Console.Write($"Новый массив: ");
        for (int i = 0; i < n; i++)
        {
            Console.Write($"{A[i]} ");
        }
        Console.ReadKey();
    }
}

```

14) найти максимум из отрицательных элементов;

```
using System;
```

```
namespace ConsoleApp5
```

```

{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();
            int[] A;
            Console.Write("Введите размерность массива - ");
            int n = int.Parse(Console.ReadLine());
            Console.Write("Введите нижнюю границу массива - ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите верхнюю границу массива - ");
            int b = int.Parse(Console.ReadLine());
            A = new int[n];
            for (int i = 0; i < n; i++)
            {

```

```

        A[i] = rnd.Next(a, b);
        Console.Write($"{A[i]} ");
    }
    int max = A[0];
    for (int i = 0; i < n; i++)
    {
        if ( max < 0 && A[i] < 0 && A[i] > max)
        {
            max = A[i];
        }
    }
    Console.WriteLine(" ");
    Console.Write($"Максимум из отрицательных элементов:
{max}");
    Console.ReadKey();
}
}
}

```

```

c:\> Консоль отладки Microsoft Visual Studio
Введите размерность массива - 10
Введите нижнюю границу массива - -12
Введите верхнюю границу массива - -1
-10 -10 -6 -3 -5 -10 -2 -10 -4 -3
Максимум из отрицательных элементов: -2

```

15) найти минимум из положительных элементов;

using System;

namespace ConsoleApp5

{

class Program

{

static void Main(string[] args)

```

{
    Random rnd = new Random();
    int[] A;
    Console.Write("Введите размерность массива - ");
    int n = int.Parse(Console.ReadLine());
    Console.Write("Введите нижнюю границу массива - ");
    int a = int.Parse(Console.ReadLine());
    Console.Write("Введите верхнюю границу массива - ");
    int b = int.Parse(Console.ReadLine());
    A = new int[n];
    for (int i = 0; i < n; i++)
    {
        A[i] = rnd.Next(a, b);
        Console.Write($"{A[i]} ");
    }
    int min = A[0];
    for (int i = 0; i < n; i++)
    {
        if (A[i] > 0 && min > 0 && A[i] < min)
        {
            min = A[i];
        }
    }
    Console.WriteLine(" ");
    Console.Write($"Минимум из положительных элементов:
{min}");

    Console.ReadKey();
}
}

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 10
Введите нижнюю границу массива - -4
Введите верхнюю границу массива - 3
2 -4 -2 -4 2 0 1 0 0 2
Минимум из положительных элементов: 1

```

16) найти максимум из модулей элементов;

```
using System;
```

```
namespace ConsoleApp5
```

```
{
```

```
class Program
```

```
{
```

```
static void Main(string[] args)
```

```
{
```

```
Random rnd = new Random();
```

```
int[] A;
```

```
Console.Write("Введите размерность массива - ");
```

```
int n = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите нижнюю границу массива - ");
```

```
int a = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите верхнюю границу массива - ");
```

```
int b = int.Parse(Console.ReadLine());
```

```
A = new int[n];
```

```
for (int i = 0; i < n; i++)
```

```
{
```

```
A[i] = rnd.Next(a, b);
```

```
Console.Write($"{A[i]} ");
```

```
}
```

```
int max = A[0];
```

```
for (int i = 0; i < n; i++)
```

```
{
```



```

        if (Math.Abs(A[i]) > Math.Abs(max))
        {
            max = A[i];
        }
    }
    Console.WriteLine(" ");
    Console.WriteLine($"Максимум из модулей элементов: {max}");
    Console.ReadKey();
}
}
}

```

```

cs Консоль отладки Microsoft Visual Studio
Введите размерность массива - 10
Введите нижнюю границу массива - -6
Введите верхнюю границу массива - 2
-5 0 -1 -4 -1 -1 -3 -1 1 -5
Максимум из модулей элементов: -5

```

Контрольная работа (Одномерные массивы) №5

11. Дан массив целых чисел. Размерность массива указать с клавиатуры. Элементы массива вводить с клавиатуры. Найти все элементы, кратные 3 или 5.

```
int[] A;
Console.Write("Введите размерность массива - ");
int n = int.Parse(Console.ReadLine());

A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}
foreach (int elem in A)
{
    if (elem % 5 == 0 || elem % 3 == 0)
    {
        Console.WriteLine($"Найден элемент кратный 3 или 5:
{elem}");
    }
}
```

```
Консоль отладки Microsoft Visual Studio
Введите размерность массива - 10
A[0] = 15
A[1] = 16
A[2] = 17
A[3] = 18
A[4] = 19
A[5] = 10
A[6] = 20
A[7] = 25
A[8] = 50
A[9] = 534
Найден элемент кратный 3 или 5: 15
Найден элемент кратный 3 или 5: 18
Найден элемент кратный 3 или 5: 10
Найден элемент кратный 3 или 5: 20
Найден элемент кратный 3 или 5: 25
Найден элемент кратный 3 или 5: 50
Найден элемент кратный 3 или 5: 534
```

16. Дан массив 15 целых чисел. Элементы массива заполняются случайно с помощью генератора случайных чисел из диапазона [15,35]. Найти сумму элементов, имеющих нечетное значение.

```
Random rnd = new Random();
int[] A;

int n = 15;
int a = 15;
int b = 36;

A = new int[n];
for (int i = 0; i < n; i++)
{
    A[i] = rnd.Next(a, b);
    Console.WriteLine($"A[{i}] = {A[i]}");
}
int sum = 0;
foreach (int elem in A)
{
```

```

        if (elem % 2 == 1)
        {
            sum += elem;
        }
    }

    Console.WriteLine($"Сумма нечетных элементов массива A:
{sum}");

    Console.ReadKey();

```

```

A[0] = 23
A[1] = 29
A[2] = 22
A[3] = 21
A[4] = 30
A[5] = 16
A[6] = 16
A[7] = 19
A[8] = 27
A[9] = 31
A[10] = 21
A[11] = 23
A[12] = 31
A[13] = 18
A[14] = 18
Сумма нечетных элементов массива A: 225

```

21. Дан массив 20 целых чисел. Элементы массива вводить с клавиатуры. Найти сумму отрицательных элементов.

```

int[] A;

int n = 20;

A = new int[n];
for (int i = 0; i < n; i++)
{
    Console.Write($"A[{i}] = ");
    A[i] = int.Parse(Console.ReadLine());
}

```

```
int sum = 0;
```

```

foreach (int elem in A)
{
    if (elem < 0)
    {
        sum += elem;
    }
}

Console.WriteLine($"Сумма нечетных элементов массива A:
{sum}");

Console.ReadKey();

```

```

A[0] = 10
A[1] = -1
A[2] = -15
A[3] = 10
A[4] = 15
A[5] = 18
A[6] = 19
A[7] = -25
A[8] = -300
A[9] = 500
A[10] = 54
A[11] = 59
A[12] = 10
A[13] = -5
A[14] = -19
A[15] = -101
A[16] = 101
A[17] = 300
A[18] = 259
A[19] = 689
Сумма нечетных элементов массива A: -466

```

Лабораторная работа 8

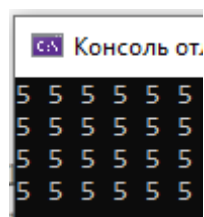
Тема: Циклы (Вложенные циклы)

Напечатать числа в виде следующей таблицы:

<p>a) 5 5 5 5 5 5</p> <p> 5 5 5 5 5 5</p> <p> 5 5 5 5 5 5</p> <p> 5 5 5 5 5 5</p>	<p>б) 1 2 ... 10</p> <p> 1 2 ... 10</p> <p> 1 2 ... 10</p> <p> 1 2 ... 10</p>	<p>в) 41 42 ... 50</p> <p> 51 52 ... 60</p> <p> </p> <p> 71 72 ... 80</p>
---	---	--

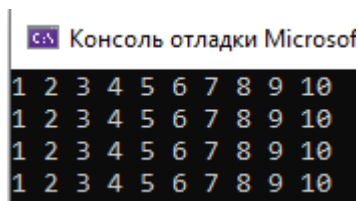
A)

```
int i = 0;
int j = 0;
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 5; j++)
    {
        Console.Write("5 ");
    }
    Console.WriteLine("5 ");
}
```



Б)

```
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 11; j++)
    {
        if (j > 0)
        {
            Console.Write($"{j} ");
        }
    }
    Console.WriteLine("");
}
```



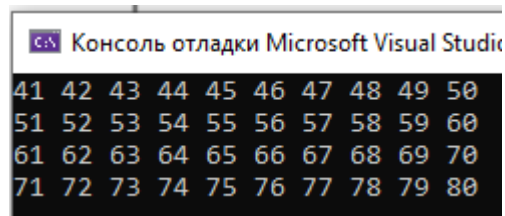
В)

```
int i = 0;
int j = 0;
int a = 41;
int b = 50;
for (i = 0; i < 4; i++)
```

```

{
    for (j = a; j <= b; j++)
    {
        Console.Write($"{j} ");
    }
    a += 10;
    b += 10;
    Console.WriteLine("");
}

```



Напечатать числа в виде следующей таблицы:

а)

```

5
5 5
5 5 5
5 5 5 5
5 5 5 5 5

```

б)

```

1 1 1 1 1
1 1 1 1
1 1 1
1 1
1

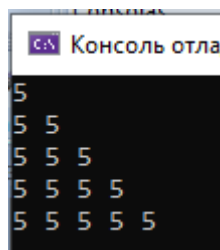
```

А)

```

int i = 0;
int j = 0;
int a = 2;
for (i = 0; i < 5; i++)
{
    for (j = 1; j < a; j++)
    {
        Console.Write($"{5} ");
    }
    a++;
    Console.WriteLine("");
}

```



Б)

```

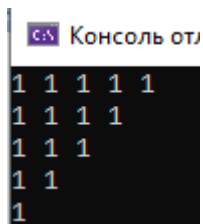
int i = 0;
int j = 0;
int a = 0;
for (i = 0; i < 5; i++)
{
    for (j = 5; j > a; j--)
    {

```

```

        Console.Write($"1 ");
    }
    a++;
    Console.WriteLine("");
}

```



Напечатать числа в виде следующей таблицы:

а) 1
2 2
3 3 3
4 4 4 4
5 5 5 5 5

в) 10
20 20
30 30 30
40 40 40 40
50 50 50 50 50

б) 5 5 5 5 5
6 6 6 6
7 7 7
8 8
9

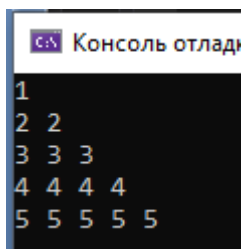
г) 5 5 5 5 5
10 10 10 10
15 15 15
20 20
25

А)

```

int i = 0;
int j = 0;
int a = 2;
int b = 1;
for (i = 0; i < 5; i++)
{
    for (j = 1; j < a; j++)
    {
        Console.Write($" {b} ");
    }
    a++;
    b++;
    Console.WriteLine("");
}

```

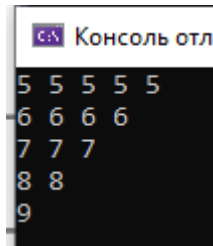


Б)


```

int i = 0;
int j = 0;
int a = 0;
int b = 5;
for (i = 0; i < 5; i++)
{
    for (j = 5; j > a; j--)
    {
        Console.Write($"{b} ");
    }
    a++;
    b++;
    Console.WriteLine("");
}

```



```

5 5 5 5 5
6 6 6 6
7 7 7
8 8
9

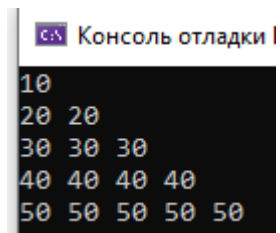
```

В)

```

int i = 0;
int j = 0;
int a = 2;
int b = 10;
for (i = 0; i < 5; i++)
{
    for (j = 1; j < a; j++)
    {
        Console.Write($"{b} ");
    }
    a++;
    b += 10;
    Console.WriteLine("");
}

```



```

10
20 20
30 30 30
40 40 40 40
50 50 50 50 50

```

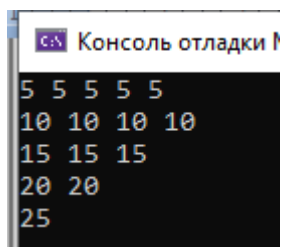
Г)

```

int i = 0;
int j = 0;
int a = 0;
int b = 5;
for (i = 0; i < 5; i++)
{
    for (j = 5; j > a; j--)
    {
        Console.Write($"{b} ");
    }
    a++;
    b += 5;
    Console.WriteLine("");
}

```

}



Напечатать числа в виде следующей таблицы:

а) 0
1 0
2 1 0
3 2 3 0
4 3 4 1 0

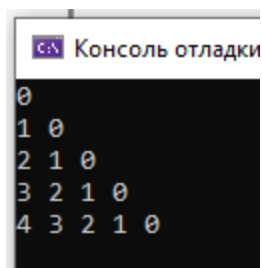
в) 30
29 30
28 29 30
27 28 29 30
26 27 28 29 30

б) 6 5 4 3 2
7 4 3 2
8 3 2
9 2
2

г) 20 21 22 23 24
19 20 21 22
18 19 20
17 18
16

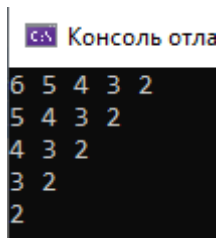
А)

```
int i;
for (i = 0; i < 5; i++)
{
    int j;
    for (j = 0; j <= i ; j++)
    {
        if (j == i)
        {
            Console.Write("0");
        }
        else
        {
            Console.Write(i - j + " ");
        }
    }
    Console.WriteLine(" ");
}
```



Б)

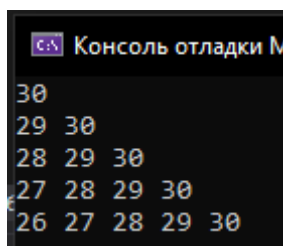
```
int i;
for (i = 0; i < 5; i++)
{
    int j;
    for (j = 6; j > i; j--)
    {
        if (j > i+1)
        {
            if (j == i)
            {
                Console.Write(2);
            }
            else
            {
                Console.Write(j - i + " ");
            }
        }
    }
    Console.WriteLine("");
}
```



```
6 5 4 3 2
5 4 3 2
4 3 2
3 2
2
```

В)

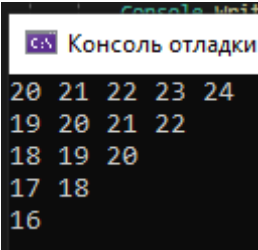
```
for (int n = 0; n < 5; n++)
{
    for (int i = 0; i <= n; i++)
    {
        if (i == n)
        {
            Console.Write(30);
        }
        else
        {
            Console.Write( 30 + i - n + " ");
        }
    }
    Console.WriteLine(" ");
}
```



```
30
29 30
28 29 30
27 28 29 30
26 27 28 29 30
```

Г)

```
for (int n = 20; n > 15; n--)  
{  
    for (int i = 0; i < n - 15; i++)  
    {  
        if (i == n)  
        {  
            Console.Write(16);  
        }  
        else  
        {  
            Console.Write(n + i + " ");  
        }  
    }  
    Console.WriteLine(" ");  
}
```



```
20 21 22 23 24  
19 20 21 22  
18 19 20  
17 18  
16
```

Лабораторная работа 9

Тема. Статические методы.

Цель: научиться использовать статические методы с параметрами и без параметров для выполнения консольных приложений расчетного характера.

Задание 1. Разработать консольное приложение в соответствии с условием задания.

Задания сгруппированы. Решить по одной задаче из каждой группы.

Критерии оценки задания.

Решены правильно две задачи – оценка удовлетворительно.

Решены правильно или с недочетами четыре задачи – оценка хорошо.

Решены пять или шесть задач, ответы на дополнительные вопросы правильные – оценка отлично.

Группа 1.

Разработать метод $\min(a,b)$ для нахождения минимального из двух чисел. Вычислить с его помощью значение выражения $z = \min(3x, 2y) + \min(x-y, x+y)$.

```
static void Main(string[] args)
{
    Console.WriteLine("Введите первое число:");
    double a = Convert.ToDouble(Console.ReadLine());

    Console.WriteLine("Введите второе число:");
    double b = Convert.ToDouble(Console.ReadLine());

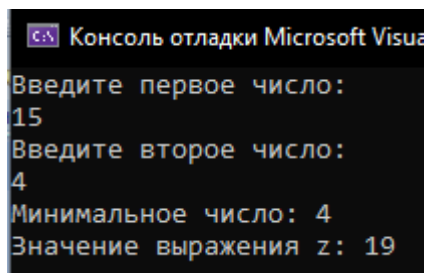
    Console.WriteLine($"Минимальное число: {Min(a,b)}");

    double z = Min(3 * a, 2 * b) + Min(a - b, a + b);
    Console.WriteLine($"Значение выражения z: {z}");
}

static double Min(double a, double b)
{
    if (a > b)
    {
        return b;
    }
    else
    {
        return a;
    }
}
```

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		157

}



Группа 2.

Разработать метод $f(x)$, который вычисляет значение по следующей формуле: $f(x)=x^3-\sin(x)$.

Определить, в какой из точек: а или b, функция принимает наибольшее значение. Значения а и b ввести с клавиатуры.

```
static void Main(string[] args)
{
    double maxSin = 0;
    Console.WriteLine("Введите число:");
    double x = Convert.ToDouble(Console.ReadLine());

    Console.WriteLine("Введите первую точку:");
    double a = Convert.ToDouble(Console.ReadLine());

    Console.WriteLine("Введите вторую точку:");
    double b = Convert.ToDouble(Console.ReadLine());

    Console.WriteLine($"Значение выражения: {FuncSin(x):f2}");

    double f1 = FuncSin(a);
    double f2 = FuncSin(b);
    if (f1 > f2)
    {
        maxSin = a;
    }
    else
    {
        maxSin = b;
    }

    Console.WriteLine($"Максимальное значение функция принимает в точке:
{maxSin}");
}

static double FuncSin(double x)
{
    double f = Math.Pow(x, 3) - Math.Sin(x);
    return f;
}
```

Группа 3.

Разработать метод $f(n, x)$, которая для заданного натурального числа n и вещественного x находит значение выражения $\frac{x^n}{n}$. Вычислить с помощью данного метода значение выражения $\frac{x^2}{2} + \frac{x^4}{4} + \frac{x^6}{6}$.

```
static void Main(string[] args)
{
    int n = 2;
    double x = 2;
    double expression = FuncExpr(x, n) + FuncExpr(x, 2 * n) + FuncExpr(x, 3 * n);
    Console.WriteLine($"Значение выражения: {expression}");
}

static double FuncExpr(double x, int n)
{
    double f = Math.Pow(x, n) / n;
    return f;
}
```

Группа 4.

Разработать метод $f(x)$, который нечетное число заменяет на 0, а четное число уменьшает в два раза. Продемонстрировать работу данного метода на примере.

```
static void Main(string[] args)
{
    while (true)
    {
        Console.WriteLine("Введите число:");
        double x = Convert.ToDouble(Console.ReadLine());

        Console.WriteLine($"Значение выполненной функции: {FuncChetNechet(x)}");
    }
}

static double FuncChetNechet(double x)
{
    double f = 0;
    if (x % 2 == 0)
    {
        f = x / 2;
    }
}
```

```

    }
    else
    {
        f = 0;
    }
    return f;
}

```

```

Введите число:
5
Значение выполненной функции: 0
Введите число:
4
Значение выполненной функции: 2
Введите число:
1
Значение выполненной функции: 0
Введите число:
2
Значение выполненной функции: 1
Введите число:

```

Группа 5.

Разработать метод $f(a, b)$, который по катетам a и b вычисляет гипотенузу. С помощью данного метода найти периметр фигуры ABDC по заданным сторонам AB, AC и DC.

```

using System;

namespace ConsoleApp5
{
    class Program
    {
        static double Func1(double AB, double AC)
        {
            double gip = Math.Sqrt(Math.Pow(AB, 2) + Math.Pow(AC, 2));
            return gip;
        }
        static double Func2(double DC, double BC)
        {
            double gip = Math.Sqrt(Math.Pow(DC, 2) + Math.Pow(BC, 2));
            return gip;
        }
        static void Main(string[] args)
        {
            Console.WriteLine("Введите сторону AB: ");
            double AB = double.Parse(Console.ReadLine());
            Console.WriteLine("Введите сторону AC: ");
            double AC = double.Parse(Console.ReadLine());

```



```

        Console.WriteLine($"Гипотенуза BC: {Func1(AB, AC):f2}");
        Console.Write("Введите сторону DC: ");
        double DC = double.Parse(Console.ReadLine());

        double BC = Func1(AB, AC);
        Console.WriteLine($"{BC}");
        Console.WriteLine($"Гипотенуза BD: {Func2(DC, BC):f2}");
        double per = AB + AC + DC + Func2(DC, BC);
        Console.WriteLine($"Периметр ABDC: {per:f2}");
        Console.ReadKey();
    }
}
}

```

Группа 6.

1. Разработать метод $f(x_1, y_1, x_2, y_2)$, который вычисляет длину отрезка по координатам вершин (x_1, y_1) и (x_2, y_2) , и метод $d(a, b, c)$, который вычисляет периметр треугольника по длинам сторон a, b, c . С помощью данных методов найти периметр треугольника, заданного координатами своих вершин.

```

using System;

namespace ConsoleApp5
{
    class Program
    {
        static double Func1(double x1, double x2, double y1, double y2)
        {
            double d = Math.Sqrt(Math.Pow(x2 - x1, 2) + Math.Pow(y2 - y1, 2));
            return d;
        }

        static double Func2(double a, double b, double c)
        {
            double per = a + b + c;

```

```

        return per;
    }
    static void Main(string[] args)
    {
        Console.Write("Введите сторону x1: ");
        double x1 = double.Parse(Console.ReadLine());
        Console.Write("Введите сторону y1: ");
        double y1 = double.Parse(Console.ReadLine());
        Console.Write("Введите сторону x2: ");
        double x2 = double.Parse(Console.ReadLine());
        Console.Write("Введите сторону y2: ");
        double y2 = double.Parse(Console.ReadLine());
        Console.WriteLine($"Длина: {Func1(x1, x2, y1, y2):f2}");
        Console.Write("Введите длину стороны a: ");
        double a = double.Parse(Console.ReadLine());
        Console.Write("Введите длину стороны b: ");
        double b = double.Parse(Console.ReadLine());
        Console.Write("Введите длину стороны c: ");
        double c = double.Parse(Console.ReadLine());
        Console.WriteLine($"Периметр треугольника со сторонами a, b, c:
{Func2(a, b, c):f2}");
        Console.ReadKey();
    }
}

```

```

Консоль отладки Microsoft Visual Studio
Введите сторону x1: 2
Введите сторону y1: 3
Введите сторону x2: 8
Введите сторону y2: 5
Длина: 6,32
Введите длину стороны a: 4
Введите длину стороны b: 4
Введите длину стороны c: 5,65
Периметр треугольника со сторонами a, b, c: 13,65

```

Задание 2. Разработать консольное приложение по варианту.

1. Разработать метод, который для заданного натурального числа N возвращает значение true, если число простое и false, если число составное. С помощью данного метода:

- 1) вывести на экран все простые числа на отрезке [a, b];
- 2) найти количество всех простых чисел на отрезке [a, b];

- 3) найти сумму всех составных чисел на отрезке [a, b];
- 4) для заданного числа A вывести на экран ближайшее предшествующее по отношению к нему простое число.

```
using System;

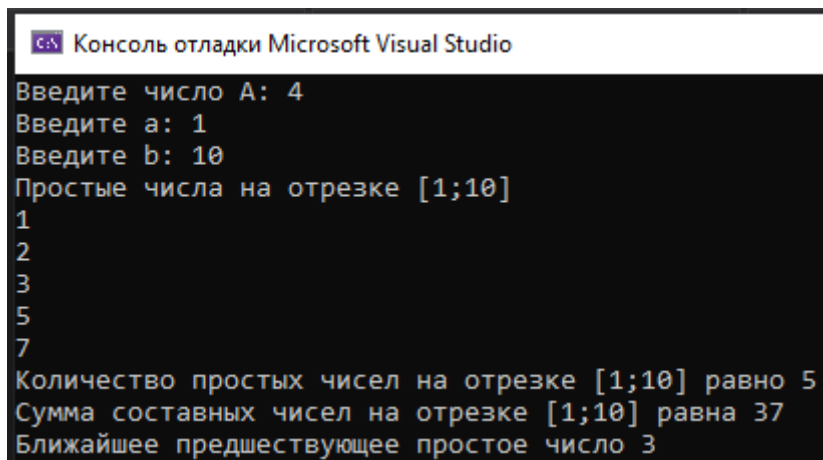
namespace ConsoleApp1
{
    class Program
    {
        static void Func(int n, ref bool t)
        {
            int x = n;
            int k = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (x % i == 0)
                {
                    k++;
                }
            }
            if (k > 1)
            {
                t = false;
            }
            else
            {
                t = true;
            }
        }
        static void Main()
        {
            int kol = 0;
            int sum = 0;
            int B = 0;
            Console.Write("Введите число A: ");
            int A = int.Parse(Console.ReadLine());
            Console.Write("Введите a: ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите b: ");
            int b = int.Parse(Console.ReadLine());
            Console.WriteLine($"Простые числа на отрезке [{a};{b}]");
        }
    }
}
```

```

bool t = false;
for (int i = a; i <= b; i++)

{
    Func(i, ref t);
    if (t == true)
    {
        Console.WriteLine($"{i}");
        kol++;
        if (A > i)
        {
            B = i;
        }
    }
    else
    {
        sum += i;
    }
    t = false;
}
Console.WriteLine($"Количество простых чисел на отрезке [{a};{b}]
равно {kol}");
Console.WriteLine($"Сумма составных чисел на отрезке [{a};{b}] равна
{sum}");
Console.WriteLine($"Ближайшее предшествующее простое число {B}");
}
}
}

```



```

Консоль отладки Microsoft Visual Studio
Введите число A: 4
Введите a: 1
Введите b: 10
Простые числа на отрезке [1;10]
1
2
3
5
7
Количество простых чисел на отрезке [1;10] равно 5
Сумма составных чисел на отрезке [1;10] равна 37
Ближайшее предшествующее простое число 3

```

2. Разработать метод, который для заданного натурального числа N возвращает количество его делителей. С помощью данного метода:

```
using System;
```

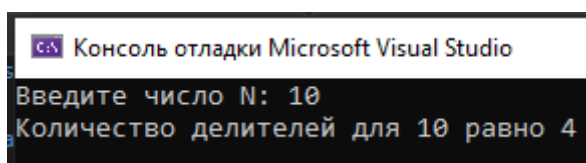
					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		164

```

namespace ConsoleApp1

{
    class Program
    {
        static int Func(int n)
        {
            int del = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (n % i == 0)
                {
                    del++;
                }
            }
            del = del + 1;
            return del;
        }
        static void Main()
        {
            Console.Write("Введите число N: ");
            int N = int.Parse(Console.ReadLine());
            Console.WriteLine($"Количество делителей для {N} равно {Func(N)}");
        }
    }
}

```



1) для каждого целого числа на отрезке $[a, b]$ вывести на экран количество делителей;

```
using System;
```

```
namespace ConsoleApp1
```

```

{
    class Program
    {

```

```
        static int Func(int n)
```

					KKOO.OAXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		165

```

    {
        int del = 0;
        for (int i = 1; i <= n / 2; i++)

        {
            if (n % i == 0)
            {
                del++;
            }
        }
        del = del + 1;
        return del;
    }
    static void Main()
    {
        Console.Write("Введите a: ");
        int a = int.Parse(Console.ReadLine());
        Console.Write("Введите b: ");
        int b = int.Parse(Console.ReadLine());
        for (int i = a; i <= b; i++)
        {
            Console.WriteLine($"Количество делителей для числа {i} -
{Func(i)}");
        }
    }
}

```

```

cs Консоль отладки Microsoft Visual Studio
Введите a: 1
Введите b: 5
Количество делителей для числа 1 - 1
Количество делителей для числа 2 - 2
Количество делителей для числа 3 - 2
Количество делителей для числа 4 - 3
Количество делителей для числа 5 - 2

```

2) вывести на экран только те целые числа отрезка [a, b], у которых количество делителей равно заданному числу;

```

using System;

namespace ConsoleApp1
{
    class Program

```

```

{
    static int Func(int n)
    {

        int del = 0;
        for (int i = 1; i <= n / 2; i++)
        {
            if (n % i == 0)
            {
                del++;
            }
        }
        del = del + 1;
        return del;
    }
    static void Main()
    {
        Console.Write("Введите N: ");
        int N = int.Parse(Console.ReadLine());
        Console.Write("Введите a: ");
        int a = int.Parse(Console.ReadLine());
        Console.Write("Введите b: ");
        int b = int.Parse(Console.ReadLine());
        for (int i = a; i <= b; i++)
        {
            if (Func(i) == N)
            {
                Console.WriteLine($"Количество делителей для числа {i}
- {Func(i)}");
            }
        }
    }
}

```

```

Консоль отладки Microsoft Visual Studio
Введите N: 2
Введите a: 1
Введите b: 8
Количество делителей для числа 2 - 2
Количество делителей для числа 3 - 2
Количество делителей для числа 5 - 2
Количество делителей для числа 7 - 2

```

3) вывести на экран только те целые числа отрезка [a, b], у которых количество делителей максимально;

```
using System;
```

```
namespace ConsoleApp1
```

```
{
```

```
class Program
```

```
{
```

```
static int Func(int n)
```

```
{
```

```
int del = 0;
```

```
for (int i = 1; i <= n / 2; i++)
```

```
{
```

```
if (n % i == 0)
```

```
{
```

```
del++;
```

```
}
```

```
}
```

```
del = del + 1;
```

```
return del;
```

```
}
```

```
static void Main()
```

```
{
```

```
int max = 0;
```

```
Console.Write("Введите a: ");
```

```
int a = int.Parse(Console.ReadLine());
```

```
Console.Write("Введите b: ");
```

```
int b = int.Parse(Console.ReadLine());
```

```
for (int i = a; i <= b; i++)
```

```
{
```

```
if (Func(i) > max)
```

```
{
```

```
max = Func(i);
```

```
}
```

```
}
```

```
Console.Write($"Числа, у которых количество делителей максимально: ");
```

```
for (int i = a; i <= b; i++)
```

```
{
```

```
if (Func(i) == max)
```

```
{
```

```
Console.Write($" {i}");
```

ККОО.ОАXXXX.000

Лист

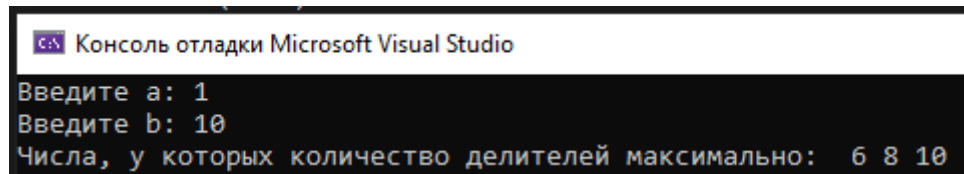
168

Изм.	Лист	№ докум.	Подпись	Дата


```

    }
    }
}
}
}

```



4) для заданного числа A вывести на экран ближайшее следующее по отношению к нему число, имеющее столько же делителей, сколько и число A .

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int n)
        {
            int del = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (n % i == 0)
                {
                    del++;
                }
            }
            del = del + 1;
            return del;
        }

        static void Main()
        {
            Console.Write("Введите число A: ");
            int A = int.Parse(Console.ReadLine());
            Console.Write("Введите a: ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите b: ");
            int b = int.Parse(Console.ReadLine());
            for (int i = a; i <= b; i++)

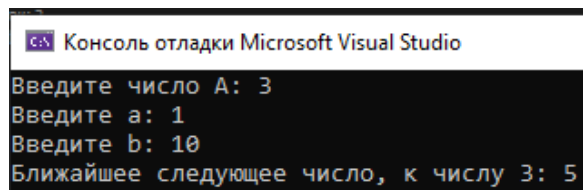
```

```

    {
        if(Func(i) == Func(A) && A < i)
        {
            Console.WriteLine($"Ближайшее следующее число, к числу {A}: {i}");

            break;
        }
    }
}
}
}
}

```



3. Разработать метод, который для заданного натурального числа N возвращает сумму его делителей. С помощью данного метода:

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (n % i == 0)
                {
                    sum += i;
                }
            }
            sum = sum + n;
            return sum;
        }

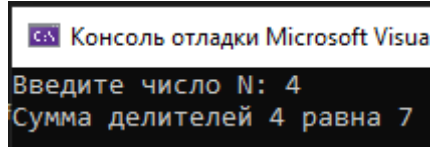
        static void Main()
        {

```

```

        Console.Write("Введите число N: ");
        int N = int.Parse(Console.ReadLine());
        Console.WriteLine($"Сумма делителей {N} равна {Func(N)}");
    }
}

```



1) для каждого целого числа на отрезке $[a, b]$ вывести на экран сумму его делителей;

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (n % i == 0)
                {
                    sum += i;
                }
            }
            sum = sum + n;
            return sum;
        }

        static void Main()
        {
            Console.Write("Введите a: ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите b: ");
            int b = int.Parse(Console.ReadLine());
            for (int i = a; i <= b; i++)
            {
                Console.WriteLine($"Сумма делителей для числа {i} - {Func(i)}");
            }
        }
    }
}

```

					ККОО.ОАХХХХ.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		171

```

    }
}
}

```

```

cs Консоль отладки Microsoft Visual Studio
Введите a: 1
Введите b: 10
Сумма делителей для числа 1 - 1
Сумма делителей для числа 2 - 3
Сумма делителей для числа 3 - 4
Сумма делителей для числа 4 - 7
Сумма делителей для числа 5 - 6
Сумма делителей для числа 6 - 12
Сумма делителей для числа 7 - 8
Сумма делителей для числа 8 - 15
Сумма делителей для числа 9 - 13
Сумма делителей для числа 10 - 18

```

2) вывести на экран только те целые числа отрезка $[a, b]$, у которых сумма делителей равна заданному числу;

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (n % i == 0)
                {
                    sum += i;
                }
            }
            sum = sum + n;

            return sum;
        }
        static void Main()
        {
            Console.Write("Введите N: ");
            int N = int.Parse(Console.ReadLine());
            Console.Write("Введите a: ");
            int a = int.Parse(Console.ReadLine());

```

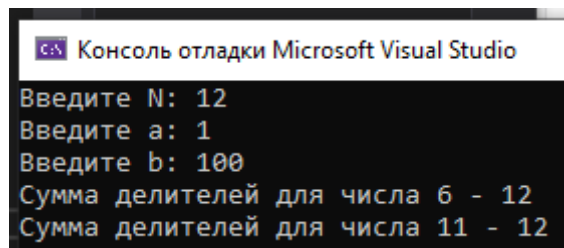
					KKOO.OAXXXX.000	Лист
						172
Изм.	Лист	№ докум.	Подпись	Дата		

```

        Console.WriteLine("Введите b: ");
        int b = int.Parse(Console.ReadLine());

        for (int i = a; i <= b; i++)
        {
            if (Func(i) == N)
            {
                Console.WriteLine($"Сумма делителей для числа {i} -
{Func(i)}");
            }
        }
    }
}

```



3) вывести на экран только те целые числа отрезка $[a, b]$, у которых сумма делителей максимальна;

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            for (int i = 1; i <= n / 2; i++)
            {
                if (n % i == 0)
                {
                    sum += i;
                }
            }
            sum = sum + n;

            return sum;
        }
    }
}

```

```

    }

    static void Main()
    {

        int max = 0;
        Console.Write("Введите a: ");
        int a = int.Parse(Console.ReadLine());
        Console.Write("Введите b: ");
        int b = int.Parse(Console.ReadLine());
        for (int i = a; i <= b; i++)
        {
            if (Func(i) > max)
            {
                max = Func(i);
            }
        }
        Console.WriteLine($"Числа, у которых сумма делителей максимально: ");
        for (int i = a; i <= b; i++)
        {
            if (Func(i) == max)
            {
                Console.WriteLine($"{i}");
            }
        }
    }
}

```

Консоль отладки Microsoft Visual Studio

```

Введите a: 1
Введите b: 50
Числа, у которых сумма делителей максимально:  48

```

4) для заданного числа A вывести на экран ближайшее предшествующее по отношению к нему число, сумма делителей которого равна сумме делителей числа A .

```

using System;

namespace ConsoleApp1
{
    class Program
    {

```

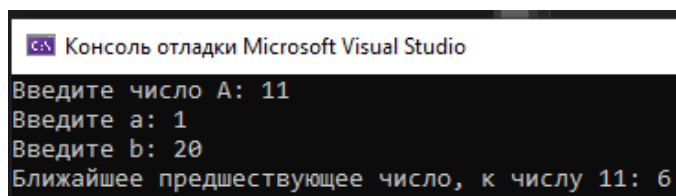
```

static int Func(int n)
{
    int sum = 0;

    for (int i = 1; i <= n / 2; i++)
    {
        if (n % i == 0)
        {
            sum += i;
        }
    }
    sum = sum + n;
    return sum;
}

static void Main()
{
    Console.Write("Введите число A: ");
    int A = int.Parse(Console.ReadLine());
    Console.Write("Введите a: ");
    int a = int.Parse(Console.ReadLine());
    Console.Write("Введите b: ");
    int b = int.Parse(Console.ReadLine());
    for (int i = a; i <= b; i++)
    {
        if (Func(i) == Func(A) && A > i)
        {
            Console.Write($"Ближайшее предшествующее число, к числу {A}:
{i}");
        }
    }
}
}

```



4. Разработать метод, которая для заданного натурального числа N возвращает сумму его цифр. С помощью данной функции:

```
using System;
```

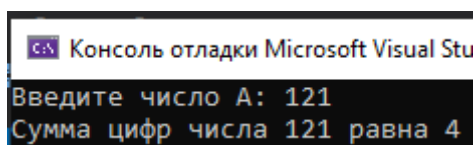
					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		175

```

namespace ConsoleApp1

{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            while(n > 0)
            {
                sum += n % 10;
                n = n / 10;
            }
            return sum;
        }
        static void Main()
        {
            Console.Write("Введите число A: ");
            int A = int.Parse(Console.ReadLine());
            Console.WriteLine($"Сумма цифр числа {A} равна {Func(A)} ");
        }
    }
}

```



1) для каждого целого числа на отрезке $[a, b]$ вывести на экран сумму его цифр;

```

using System;

namespace ConsoleApp1

{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            while (n > 0)

```



```

        {
            sum += n % 10;
            n = n / 10;
        }

        return sum;
    }
    static void Main()
    {
        Console.Write("Введите a: ");
        int a = int.Parse(Console.ReadLine());
        Console.Write("Введите b: ");
        int b = int.Parse(Console.ReadLine());
        for (int i = a; i <= b; i++)
        {
            Console.WriteLine($"Сумма цифр числа числа {i} равна {Func(i)}");
        }
    }
}

```

```

Консоль отладки Microsoft Visual Studio
Введите a: 11
Введите b: 21
Сумма цифр числа числа 11 равна 2
Сумма цифр числа числа 12 равна 3
Сумма цифр числа числа 13 равна 4
Сумма цифр числа числа 14 равна 5
Сумма цифр числа числа 15 равна 6
Сумма цифр числа числа 16 равна 7
Сумма цифр числа числа 17 равна 8
Сумма цифр числа числа 18 равна 9
Сумма цифр числа числа 19 равна 10
Сумма цифр числа числа 20 равна 2
Сумма цифр числа числа 21 равна 3

```

2) вывести на экран только те целые числа отрезка [a, b], у которых сумма цифр числа равна заданному значению;

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int n)
        {

```

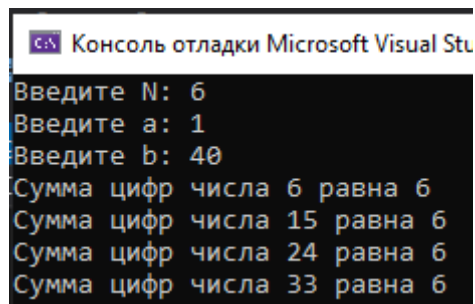
```

        int sum = 0;
        while (n > 0)
        {

            sum += n % 10;
            n = n / 10;
        }
        return sum;
    }
    static void Main()
    {
        Console.Write("Введите N: ");
        int N = int.Parse(Console.ReadLine());
        Console.Write("Введите a: ");
        int a = int.Parse(Console.ReadLine());
        Console.Write("Введите b: ");
        int b = int.Parse(Console.ReadLine());
        for (int i = a; i <= b; i++)

        {
            if (Func(i) == N)
            {
                Console.WriteLine($"Сумма цифр числа {i} равна {Func(i)}");
            }
        }
    }
}

```



```

Консоль отладки Microsoft Visual Stu
Введите N: 6
Введите a: 1
Введите b: 40
Сумма цифр числа 6 равна 6
Сумма цифр числа 15 равна 6
Сумма цифр числа 24 равна 6
Сумма цифр числа 33 равна 6

```

3) вывести на экран только те целые числа отрезка [a, b], у которых сумма цифр нечетная;

```
using System;
```

```
namespace ConsoleApp1
```

```

{
    class Program
    {
        static int Func(int n)
        {
            int sum = 0;
            while (n > 0)
            {
                sum += n % 10;
                n = n / 10;
            }
            return sum;
        }
        static void Main()
        {
            Console.Write("Введите a: ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите b: ");
            int b = int.Parse(Console.ReadLine());
            for (int i = a; i <= b; i++)
            {
                if (Func(i) % 2 == 0)
                {
                    Console.WriteLine($"Сумма цифр числа {i} чётная и равна
{Func(i)}");
                }
            }
        }
    }
}

```

4) для заданного числа А вывести на экран ближайшее предшествующее по отношению к нему число, сумма цифр которого равна сумме цифр числа А.

```
using System;
```

```

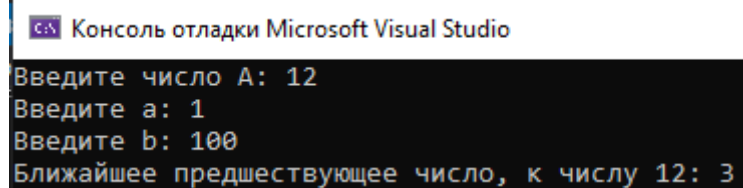
namespace ConsoleApp1

{
    class Program

    {
        static int Func(int n)
        {
            int sum = 0;
            while (n > 0)
            {
                sum += n % 10;
                n = n / 10;
            }
            return sum;
        }

        static void Main()
        {
            Console.Write("Введите число A: ");
            int A = int.Parse(Console.ReadLine());
            Console.Write("Введите a: ");
            int a = int.Parse(Console.ReadLine());
            Console.Write("Введите b: ");
            int b = int.Parse(Console.ReadLine());
            for (int i = a; i <= b; i++)
            {
                if (Func(i) == Func(A) && A > i)
                {
                    Console.Write($"Ближайшее предшествующее число, к числу {A}:
{i}");
                }
            }
        }
    }
}

```



```

Консоль отладки Microsoft Visual Studio
Введите число A: 12
Введите a: 1
Введите b: 100
Ближайшее предшествующее число, к числу 12: 3

```

5. Разработать метод, который для заданных натуральных чисел возвращает их наибольший общий делитель. С помощью данной функции:

```
using System;

namespace ConsoleApp1

{
    class Program
    {
        static int Func(int a, int b)
        {
            int max, min;
            if (Math.Abs(a) > Math.Abs(b))
            {
                max = Math.Abs(a);
                min = Math.Abs(b);
            }
            Else
            {
                max = Math.Abs(b);
                min = Math.Abs(a);
            }
            int r = max % min;
            if (r == 0)
            {
                return min;
            }
            else
            {
                if (min % r == 0)
                {
                    return r;
                }
                else
                {
                    return 1;
                }
            }
        }
    }
}
```

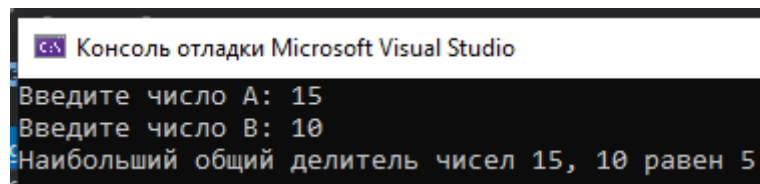
```
static void Main()
```

					KKOO.OAXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		181

```

    {
        Console.Write("Введите число A: ");
        int A = int.Parse(Console.ReadLine());
        Console.Write("Введите число B: ");
        int B = int.Parse(Console.ReadLine());
        Console.WriteLine($"Наибольший общий делитель чисел {A}, {B} равен
{Func(A, B)}");
    }
}

```



1) сократить дробь вида a/b ;

```
using System;
```

```
namespace ConsoleApp1
```

```

{
    class Program
    {
        static int Func(int a, int b)
        {
            int max, min;
            if (Math.Abs(a) > Math.Abs(b))
            {
                max = Math.Abs(a);
                min = Math.Abs(b);
            }
            else
            {
                max = Math.Abs(b);
                min = Math.Abs(a);
            }
            int r = max % min;
            if (r == 0)
            {
                return min;
            }

```

```
else
```

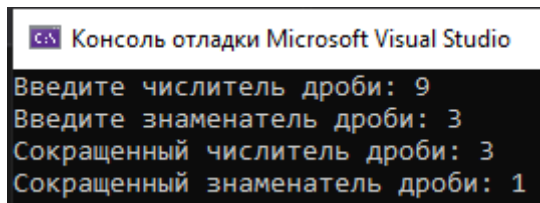
					ККОО.ОАXXXX.000	Лист
						182
Изм.	Лист	№ докум.	Подпись	Дата		

```

    {
        if (min % r == 0)
        {
            return r;
        }
        else
        {
            return 1;
        }
    }
}

static void Main()
{
    Console.Write("Введите числитель дроби: ");
    int A = int.Parse(Console.ReadLine());
    Console.Write("Введите знаменатель дроби: ");
    int B = int.Parse(Console.ReadLine());
    int ch = A / Func(A, B);
    int zn = B / Func(A, B);
    Console.WriteLine($"Сокращенный числитель дроби: {ch}");
    Console.WriteLine($"Сокращенный знаменатель дроби: {zn}");
}
}
}

```



2) найти наименьшее общее кратное для двух натуральных чисел;

```
using System;
```

```
namespace ConsoleApp1
```

```

{
    class Program
    {
        static int Func(int a, int b)
        {
            int max, min;
            if (Math.Abs(a) > Math.Abs(b))
            {

```

```

        max = Math.Abs(a);
        min = Math.Abs(b);
    }
    else
    {
        max = Math.Abs(b);
        min = Math.Abs(a);
    }

    int r = max % min;
    if (r == 0)
    {
        return min;
    }
    else
    {
        return r;
    }
}

static void Main()
{
    Console.Write("Введите число A: ");
    int A = int.Parse(Console.ReadLine());
    Console.Write("Введите число B: ");
    int B = int.Parse(Console.ReadLine());
    int krat = (A * B) / Func(A, B);
    Console.WriteLine($"Наименьшее общее кратное чисел {A} и {B}:
{krat}");
}
}
}

```

```

cs Консоль отладки Microsoft Visual Studio
Введите число A: 12
Введите число B: 10
Наименьшее общее кратное чисел 12 и 10: 60

```

3) вычислить значение выражения $\frac{a}{b} + \frac{d}{c}$; результат представить в виде обыкновенной дроби, выполнив сокращение;

```
using System;
```

```
namespace ConsoleApp1
```

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		184


```

{
    class Program
    {
        static int Func(int a, int b)
        {
            int max, min;
            if (Math.Abs(a) > Math.Abs(b))
            {

                max = Math.Abs(a);
                min = Math.Abs(b);
            }
            else
            {
                max = Math.Abs(b);
                min = Math.Abs(a);
            }
            int r = max % min;

            if (r == 0)
            {
                return min;
            }
            else
            {
                if (min % r == 0)
                {
                    return r;
                }
                else
                {
                    return 1;
                }
            }
        }
        static void Main()
        {
            Console.WriteLine("Введите числитель дроби: ");
            int A = int.Parse(Console.ReadLine());
            Console.WriteLine("Введите знаменатель дроби: ");
            int B = int.Parse(Console.ReadLine());
            Console.WriteLine("Введите числитель дроби: ");

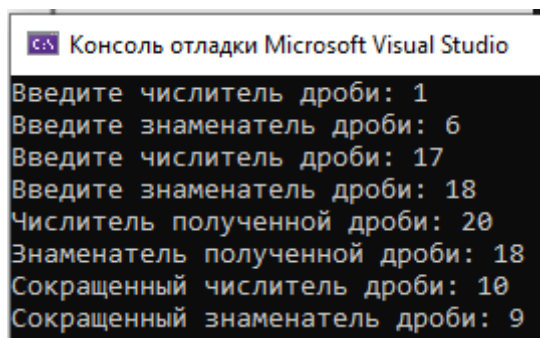
```

```

        int D = int.Parse(Console.ReadLine());
        Console.WriteLine("Введите знаменатель дроби: ");
        int C = int.Parse(Console.ReadLine());
        int krat = (B * C) / Func(B, C);
        int zn1 = (krat / B) * A;
        int zn2 = (krat / C) * D;
        int sum = zn1 + zn2;
        Console.WriteLine($"Числитель полученной дроби: {sum}");
        Console.WriteLine($"Знаменатель полученной дроби: {krat}");
        int ch = sum / Func(sum, krat);
        int zn = krat / Func(sum, krat);

        Console.WriteLine($"Сокращенный числитель дроби: {ch}");
        Console.WriteLine($"Сокращенный знаменатель дроби: {zn}");
    }
}

```



4) найти наибольший общий делитель для n натуральных чисел.

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static int Func(int a, int b)
        {
            int max, min;
            if (Math.Abs(a) > Math.Abs(b))
            {
                max = Math.Abs(a);
                min = Math.Abs(b);
            }
            else
            {

```

```

        max = Math.Abs(b);
        min = Math.Abs(a);
    }
    int r = max % min;
    if (r == 0)
    {
        return min;
    }
    else
    {
        if (min % r == 0)
        {
            return r;
        }
        else
        {
            return 1;
        }
    }
}

static void Main()
{
    int[] A;
    int r = 0;
    int t = 0;
    Console.Write("Введите размерность массива - ");
    int n = int.Parse(Console.ReadLine());
    A = new int[n];
    for (int i = 0; i < n; i++)
    {
        Console.Write($"A[{i}] = ");
        A[i] = int.Parse(Console.ReadLine());
    }
    int z = 1;
    while (z < n)
    {
        r = Func(A[0], A[z]);
        t = Func(r, A[z]);
        z++;
    }
    Console.WriteLine($"Наибольший общий делитель: {t}");
}

```

```

    }
}

```

```

Консоль отладки Microsoft Visual Studio
Введите размерность массива - 5
A[0] = 10000
A[1] = 10000
A[2] = 1000
A[3] = 100
A[4] = 10
Наибольший общий делитель: 10

```

Контрольная работа

Тема. Статические методы

Задание 1. При выполнении задания использовать метод с типом результата отличным от void.

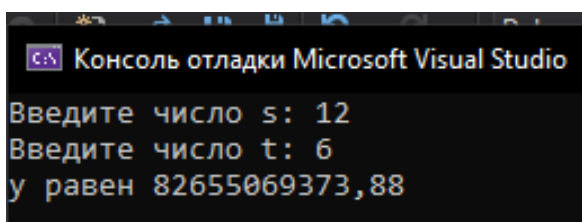
4. Даны действительные числа s , t . Вычислить значение y , используя метод для вычисления функции $h(a,b)$

$$y = h(s, t) + h^2(s - t, st) + h(1, 1)$$

$$h(s, t) = \frac{a}{1 + b^2} + \frac{b}{1 + a^2} - (a - b)^3$$

```
using System;

namespace ConsoleApp2
{
    class Program
    {
        static double h(double a, double b)
        {
            double h = (a / (1 + Math.Pow(b, 2)) + (b / (1 + Math.Pow(a, 2))) -
Math.Pow((a - b), 3));
            return h;
        }
        static void Main(string[] args)
        {
            Console.WriteLine("Введите число s: ");
            int s = int.Parse(Console.ReadLine());
            Console.WriteLine("Введите число t: ");
            int t = int.Parse(Console.ReadLine());
            double y = h(s, t) + Math.Pow(h(s-t, s*t), 2) + h(1, 1);
            Console.WriteLine($"y равен {y:f2}");
        }
    }
}
```



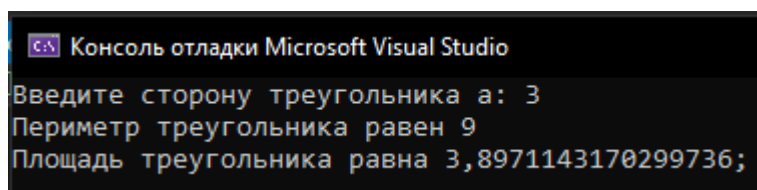
Задание 2. При выполнении задания использовать метод с типом результата void.

4. Разработать метод TrianglePS(), вычисляющий по стороне a равностороннего треугольника его периметр $P = 3 \cdot a$ и площадь $S = \frac{a^2 \sqrt{3}}{4}$

С помощью метода TrianglePS() найти в главном методе периметры и площади трех равносторонних треугольников с данными сторонами.

```
using System;

namespace ConsoleApp2
{
    class Program
    {
        static void TrianglePS(double a, ref double P, ref double S)
        {
            P = a * 3;
            S = (Math.Pow(a, 2) * Math.Sqrt(3)) / 4;
        }
        static void Main(string[] args)
        {
            for (int i = 0; i < 3; i++)
            {
                Console.Write("Введите сторону треугольника а: ");
                int a = int.Parse(Console.ReadLine());
                double S = 0;
                double P = 0;
                TrianglePS(a, ref P, ref S);
                Console.WriteLine($"Периметр треугольника равен {P} ");
                Console.WriteLine($"Площадь треугольника равна {S}; ");
            }
        }
    }
}
```



Задание 3. При выполнении задания использовать метод с типом результата отличным от void.

4. Разработать метод - функцию RingS(R1, R2) вещественного типа, находящую площадь кольца, заключенного между двумя окружностями с общим центром и радиусами R1 и R2 (R1 и R2 — вещественные, R1 > R2). В главном методе с ее помощью найти площади трех колец, для которых даны внешние и внутренние радиусы. Воспользоваться формулой площади круга радиуса R: $S = \pi \cdot R^2$. В качестве значения π использовать 3.14.

```
using System;

namespace ConsoleApp2
{
    class Program
    {
```

```

static double RingS(double R1, double R2)
{
    return 3.14 * (R1 + R2) * (R1 - R2);
}

static void Main(string[] args)
{
    for (int i = 0; i < 3; i++)
    {
        Console.Write("Введите внешний радиус кольца: ");
        int a = int.Parse(Console.ReadLine());
        Console.Write("Введите внутренний радиус кольца: ");
        int b = int.Parse(Console.ReadLine());
        Console.WriteLine($"Площадь кольца равна: {RingS(a, b)}");
    }
}

```

Консоль отладки Microsoft Visual Studio

```

Введите внешний радиус кольца: 12
Введите внутренний радиус кольца: 4
Площадь кольца равна: 401,92
Введите внешний радиус кольца: 15
Введите внутренний радиус кольца: 6
Площадь кольца равна: 593,46
Введите внешний радиус кольца: 4
Введите внутренний радиус кольца: 1
Площадь кольца равна: 47,1

```

Лабораторная работа 12

Тема. Строки

Цель: изучить встроенную поддержку C# работы со строками, изучить множество встроенных методов для сравнения, поиска, сортировки и управления строковыми значениями

Задание 1. Для закрепления теоретического материала по теме Строки необходимо изучить теоретический материал и выполнить предложенные задания.

При решении задач использовать тип string.

Задание. Разработать программу, которая позволяет ввести строку с экрана и для введенной строки s:

1) подсчитывает общее число вхождений символов x и y;

```
static void Print(char[] arr)
{
    foreach (char elem in arr)
    {
        Console.Write(elem);
    }
    Console.WriteLine();
}

static void Main(string[] args)
{
    string s;
    char x, y;

    char[] arr;

    int count = 0;

    x = 'a';
    y = 'п';

    Console.WriteLine("Введите строку: ");
    s = Convert.ToString(Console.ReadLine());

    arr = s.ToCharArray();

    Console.WriteLine("Ваш массив символов: ");
    Print(arr);

    for (int i = 0; i < arr.Length; i++)
    {
        if (arr[i] == x || arr[i] == y)
        {
```

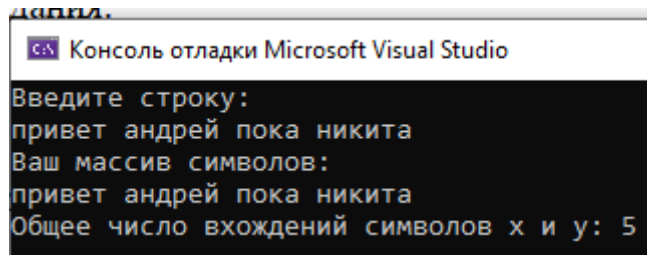
					KKOO.OAXXXX.000	Лист
						192
Изм.	Лист	№ докум.	Подпись	Дата		


```

        count++;
    }
}

Console.WriteLine($"Общее число вхождений символов x и y: {count}");
}

```



2) определяет, какой из двух заданных символов встречается в строке чаще всего;

```

    static void Print(char[] arr)
    {
        foreach (char elem in arr)
        {
            Console.Write(elem);
        }
        Console.WriteLine();
    }

    static void Main(string[] args)
    {
        string s;
        char x, y;

        char[] arr;

        int countx = 0;
        int county = 0;

        x = 'a';
        y = 'и';

        Console.WriteLine("Введите строку: ");
        s = Convert.ToString(Console.ReadLine());

        arr = s.ToCharArray();

        Console.WriteLine("Ваш массив символов: ");
        Print(arr);

        for (int i = 0; i < arr.Length; i++)
        {
            if (arr[i] == x)
            {
                countx++;
            }

            if (arr[i] == y)
            {
                county++;
            }
        }

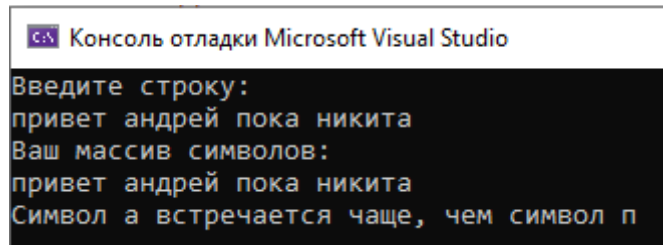
        if (countx > county)
        {
            Console.WriteLine($"Символ {x} встречается чаще, чем символ {y}");
        }
        if (countx < county)
    }

```

```

{
    Console.WriteLine($"Символ {y} встречается чаще, чем символ {x}");
}
else
{
    Console.WriteLine($"Символы встречаются одинаково.");
}
}

```



- 3) выводит на экран символы, которые наиболее часто встречается в строке;
- 4) выводит на экран символы, которые встречаются в строке только один раз;
- 5) определяет, имеются ли в строке два соседствующих одинаковых символа;
- 6) определяет, является ли строка палиндромом;
- 7) определяет, упорядочены ли по алфавиту символы строки;

```

static void Print(char[] arr)
{
    foreach (char elem in arr)
    {
        Console.Write(elem);
    }
    Console.WriteLine();
}

static void Main(string[] args)
{
    string s;
    bool tf = true;

    char[] arr_sort;
    char[] arr_nonesort;

    Console.WriteLine("Введите строку: ");
    s = Convert.ToString(Console.ReadLine());
    arr_nonesort = s.ToCharArray();
    arr_sort = s.ToCharArray();
    Array.Sort(arr_sort);
    Console.WriteLine("Отсортированный массив: ");
    Print(arr_sort);

    for (int i = 0; i < arr_sort.Length; i++)
    {
        if (arr_sort[i] == arr_nonesort[i])
        {
            tf = true;

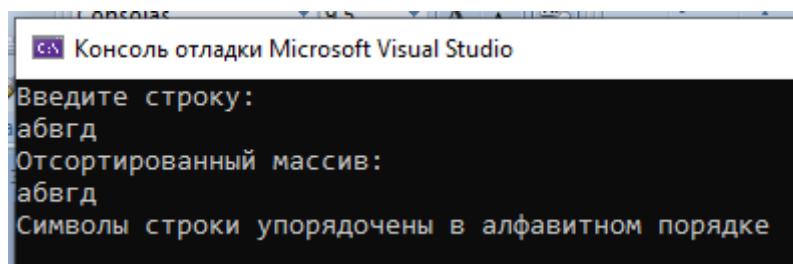
```

```

    }
    else
    {
        tf = false;
    }
}

if (tf == true)
{
    Console.WriteLine("Символы строки упорядочены в алфавитном порядке");
}
else
{
    Console.WriteLine("Символы строки не упорядочены в алфавитном порядке");
}
}

```



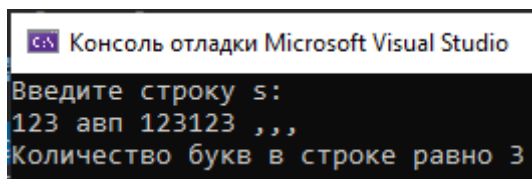
8) подсчитывает количество букв в строке;

```

using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите строку s:");
            string s = Console.ReadLine( );
            char[] a;
            int sh1 = 0;
            a = s.ToCharArray();
            for (int i = 0; i < a.Length; i++)
            {
                if (char.IsLetter(a[i]))
                {
                    sh1++;
                }
            }
            Console.WriteLine($"Количество букв в строке равно {sh1}");
        }
    }
}

```



9) подсчитывает количество цифр в строке;

```

static void Print(char[] arr)
{
    foreach (char elem in arr)

```

					ККОО.ОАХХХХ.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		195

```

        {
            Console.Write(elem);
        }
        Console.WriteLine();
    }

    static void Main(string[] args)
    {
        string s;

        int count = 0;
        char[] arr;

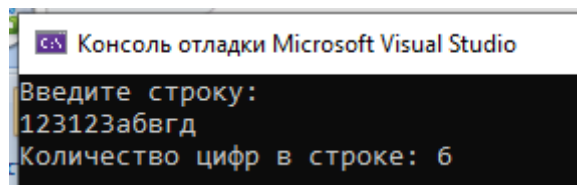
        Console.WriteLine("Введите строку: ");
        s = Convert.ToString(Console.ReadLine());

        arr = s.ToCharArray();

        for (int i = 0; i < arr.Length; i++)
        {
            if (char.IsDigit(arr[i]))
            {
                count++;
            }
        }

        Console.WriteLine($"Количество цифр в строке: {count}");
    }

```



10) подсчитывает сумму всех содержащихся в строке цифр;

```

    static void Print(char[] arr)
    {
        foreach (char elem in arr)
        {
            Console.Write(elem);
        }
        Console.WriteLine();
    }

    static void Main(string[] args)
    {
        string s;
        int num = 0;

        char[] arr;

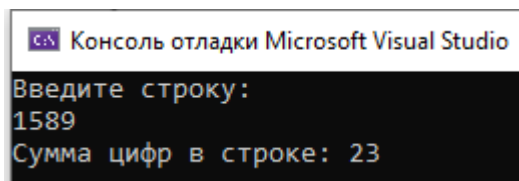
        Console.WriteLine("Введите строку: ");
        s = Convert.ToString(Console.ReadLine());

        arr = s.ToCharArray();

        for (int i = 0; i < arr.Length; i++)
        {
            if (char.IsDigit(arr[i]))
            {
                num += arr[i] - '0';
            }
        }
    }

```

```
Console.WriteLine($"Сумма цифр в строке: {num}");
}
```



11) выводит на экран последовательность символов, расположенных до первого двоеточия;

```
static void Print(char[] arr)
{
    foreach (char elem in arr)
    {
        Console.Write(elem);
    }
    Console.WriteLine();
}

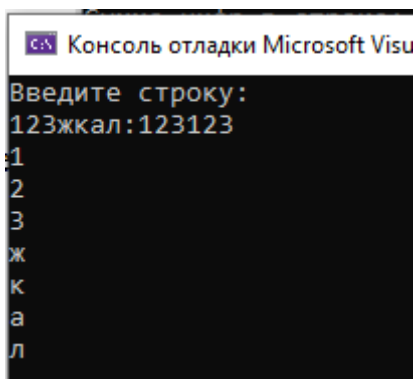
static void Main(string[] args)
{
    string s;
    int num = 0;

    char[] arr;
    char dvoethochie = ':';

    Console.WriteLine("Введите строку: ");
    s = Convert.ToString(Console.ReadLine());

    arr = s.ToCharArray();

    for (int i = 0; i < arr.Length; i++)
    {
        if (dvoethochie == arr[i])
        {
            break;
        }
        else
        {
            Console.WriteLine(arr[i]);
        }
    }
}
```



12) выводит на экран последовательность символов, расположенных после последнего двоеточия;

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Введите строку S: ");
            string s = Console.ReadLine();
            Console.Write("Строка после последнего двоеточия = ");
            int index = s.LastIndexOf(':');
            if (index >= 0)
            {
                Console.WriteLine(s.Substring(index+1));
            }
            else
            {
                Console.WriteLine(s);
            }
        }
    }
}
```

13) выводит на экран последовательность символов, расположенных между круглыми скобками (считается, что в строке ровно одна пара круглых скобок);

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Введите строку S: ");
            string s = Console.ReadLine();
            Console.Write("Между скобками = ");
            int index = s.IndexOf('(');
            int index2 = s.LastIndexOf(')');
            if (index >= 0 && index2 >= 0)
            {
                Console.WriteLine(s.Substring(index+1, index2 - (index + 1)));
            }
        }
    }
}
```

```
        else
        {
            Console.WriteLine(s);
        }
    }
}
```

					ККОО.ОАXXXX.000	Лист
						199
Изм.	Лист	№ докум.	Подпись	Дата		

Лабораторная работа

Тема. Строки

Цель: изучить встроенную поддержку C# работы со строками, изучить множество встроенных методов для сравнения, поиска, сортировки и управления строковыми значениями

Задание 2. Для закрепления теоретического материала по теме Строки необходимо изучить теоретический материал и выполнить предложенные задания.

При решении задач использовать класс StringBuilder

Задание. Разработать программу, которая позволяет ввести строку с экрана и выполняет следующие действия с введенной строкой:

- 1) вставляет в строку символ x после каждого вхождения символа y;

```
static void Main(string[] args)
{

    Console.WriteLine("Введите любую строку: ");
    StringBuilder user_str = new StringBuilder(Console.ReadLine());

    Console.WriteLine("Введите символ, который хотите вставлять после
каждого символа вашей строки: ");
    string x = Console.ReadLine();

    Console.WriteLine("Введите символ, после которого надо вставить
предыдущий символ: ");
    char y = Convert.ToChar(Console.ReadLine());

    for (int i = 0; i < user_str.Length; i++)
    {
        if (user_str[i] == y)
        {
            user_str.Insert(i + 1, x);
        }
    }

    Console.WriteLine($"Измененная строка: {user_str}");
}
```



```

Консоль отладки Microsoft Visual Studio
Введите любую строку:
hello androllo
Введите символ, который хотите вставлять после каждого символа вашей строки:
!
Введите символ, после которого надо вставить предыдущий символ:
1
Измененная строка: hel!!lo androl!!lo

```

2) вставляет в строку подстроку x после каждого вхождения подстроки y;

```

static void Main(string[] args)
{
    Console.WriteLine("Введите любую строку: ");
    StringBuilder user_str = new StringBuilder(Console.ReadLine());

    Console.WriteLine("Введите подстроку: ");
    string x = Console.ReadLine();

    Console.WriteLine("Введите другую подстроку: ");
    string y = Console.ReadLine();

    user_str.Replace(x, x + y);

    Console.WriteLine($"Измененная строка: {user_str}");
}

```

```

Консоль отладки Microsoft Visual Studio
Введите любую строку:
привет андрей
Введите подстроку:
привет
Введите другую подстроку:
мой друг
Измененная строка: привет мой друг андрей

```

3) удваивает каждое вхождение заданного символа x;

```

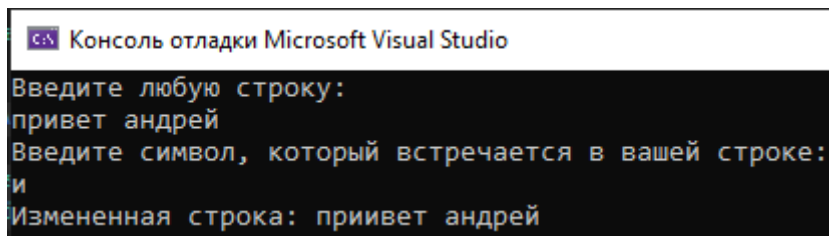
static void Main(string[] args)
{
    Console.WriteLine("Введите любую строку: ");
    StringBuilder user_str = new StringBuilder(Console.ReadLine());

    Console.WriteLine("Введите символ, который встречается в вашей строке: ");
    string x = Console.ReadLine();

    user_str.Replace(x, x + x);

    Console.WriteLine($"Измененная строка: {user_str}");
}

```



```

Консоль отладки Microsoft Visual Studio
Введите любую строку:
привет андрей
Введите символ, который встречается в вашей строке:
и
Измененная строка: привет андрей

```

4) удваивает каждое вхождение заданной подстроки x;

```

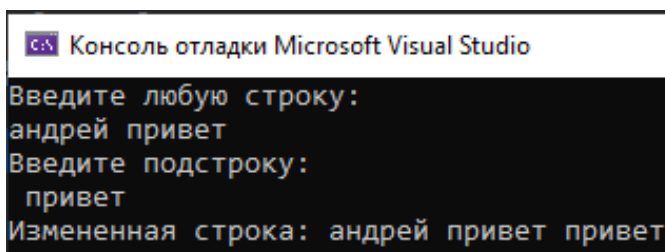
static void Main(string[] args)
{
    Console.WriteLine("Введите любую строку: ");
    StringBuilder user_str = new StringBuilder(Console.ReadLine());

    Console.WriteLine("Введите подстроку: ");
    string x = Console.ReadLine();

    user_str.Replace(x, x + x);

    Console.WriteLine($"Измененная строка: {user_str}");
}

```



```

Консоль отладки Microsoft Visual Studio
Введите любую строку:
андрей привет
Введите подстроку:
привет
Измененная строка: андрей привет привет

```

5) удаляет среднюю букву, если длина строки нечетная, и две средних, если длина строки четная;

```

static void Main(string[] args)
{
    Console.WriteLine("Введите любую строку: ");
    StringBuilder user_str = new StringBuilder(Console.ReadLine());
    StringBuilder edit_str = new StringBuilder();
    string tmp_user_str = Convert.ToString(user_str);

    string[] txtMass;
    txtMass = tmp_user_str.Split(' ');

    for (int i = 0; i <= txtMass.Length-1; i++)
    {
        char[] txtMassChar = txtMass[i].ToCharArray();
        Console.WriteLine(txtMassChar);
        for (int j = 0; j < txtMassChar.Length; j++)
        {
            int tmp_bool = txtMassChar.Length % 2;
            int middle = txtMassChar.Length / 2;
            if (tmp_bool == 0)
            {
                txtMass[i].ToString();
                txtMass[i].Remove(txtMassChar[middle]);
            }
            else
            {
                txtMass[i].ToString();
                txtMass[i].Remove(txtMassChar[middle]);
            }
        }
    }
}

```

```

        }
    }
    edit_str.Append(txtMass[i]);
}

Console.WriteLine($"Измененная строка: {edit_str}");
}

```

					ККОО.ОАXXXX.000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		203