

Министерство науки и высшего образования РФ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Пермский государственный национальный исследовательский университет»

Институт компьютерных наук и  
технологий

**ОТЧЁТ**  
по лабораторной работе №1  
по дисциплине «Языки программирования»  
Вариант 1

Работу выполнил  
студент группы ФИТ-1,2-2023 2 курса  
Шатров В.С.  
«01» октября 2024 г.

Работу проверил  
Ракина В.Д.  
«\_\_» \_\_\_\_\_ 2024 г.

Пермь 2024

## СОДЕРЖАНИЕ

Задание 1 .....	6
Текст задания .....	6
Алгоритм решения.....	6
Тестирование .....	6
Код программы .....	7
Задание 2 .....	8
Текст задания .....	8
Алгоритм решения.....	8
Тестирование .....	8
Код программы .....	9
Задание 3 .....	10
Текст задания .....	10
Алгоритм решения.....	10
Тестирование .....	10
Код программы .....	11
Задание 4 .....	12
Текст задания .....	12
Алгоритм решения .....	12
Тестирование.....	12
Код программы .....	13
Задание 5 .....	14
Текст задания .....	14
Алгоритм решения .....	14
Тестирование.....	14
Код программы .....	15
Задание 6 .....	16
Текст задания .....	16

Алгоритм решения .....	16
Тестирование.....	16
Код программы .....	17
Задание 7 .....	18
Текст задания .....	18
Алгоритм решения .....	18
Тестирование.....	18
Код программы .....	19
Задание 8 .....	20
Текст задания .....	20
Алгоритм решения .....	20
Тестирование.....	20
Код программы .....	21
Задание 9 .....	22
Текст задания .....	22
Алгоритм решения .....	22
Тестирование.....	22
Код программы .....	23
Задание 10 .....	24
Текст задания .....	24
Алгоритм решения .....	24
Тестирование.....	25
Код программы .....	25
Задание 11 .....	26
Текст задания .....	26
Алгоритм решения .....	26
Тестирование.....	26
Код программы .....	27

<b>Задание 12 .....</b>	<b>28</b>
<b>Текст задания .....</b>	<b>28</b>
<b>Алгоритм решения .....</b>	<b>28</b>
<b>Тестирование.....</b>	<b>28</b>
<b>Код программы .....</b>	<b>29</b>
<b>Задание 13 .....</b>	<b>30</b>
<b>Текст задания .....</b>	<b>30</b>
<b>Алгоритм решения .....</b>	<b>30</b>
<b>Тестирование.....</b>	<b>30</b>
<b>Код программы .....</b>	<b>31</b>
<b>Задание 14 .....</b>	<b>32</b>
<b>Текст задания .....</b>	<b>32</b>
<b>Алгоритм решения .....</b>	<b>32</b>
<b>Тестирование.....</b>	<b>32</b>
<b>Код программы .....</b>	<b>33</b>
<b>Задание 15 .....</b>	<b>34</b>
<b>Текст задания .....</b>	<b>34</b>
<b>Алгоритм решения .....</b>	<b>34</b>
<b>Тестирование.....</b>	<b>34</b>
<b>Код программы .....</b>	<b>35</b>
<b>Задание 16 .....</b>	<b>36</b>
<b>Текст задания .....</b>	<b>36</b>
<b>Алгоритм решения .....</b>	<b>36</b>
<b>Тестирование.....</b>	<b>36</b>
<b>Код программы .....</b>	<b>37</b>
<b>Задание 17 .....</b>	<b>38</b>
<b>Текст задания .....</b>	<b>38</b>
<b>Алгоритм решения .....</b>	<b>38</b>

Тестирование.....	38
Код программы .....	40
Задание 18 .....	41
Текст задания .....	41
Алгоритм решения .....	41
Тестирование.....	41
Код программы .....	42
Задание 19 .....	43
Текст задания .....	43
Алгоритм решения .....	43
Тестирование.....	43
Код программы .....	44
Задание 20 .....	45
Текст задания .....	45
Алгоритм решения .....	45
Тестирование.....	45
Код программы .....	46

## Задание 1

### Текст задания

Сумма знаков.

Дана сигнатура метода: `public int sumLastNums (int x)`; Необходимо реализовать метод таким образом, чтобы он возвращал результат сложения двух последних знаков числа `x`, предполагая, что знаков в числе не менее двух.

Подсказки: `int x=123%10;` // `x` будет иметь значение 3 `int y=123/10;` // `y` будет иметь значение 12

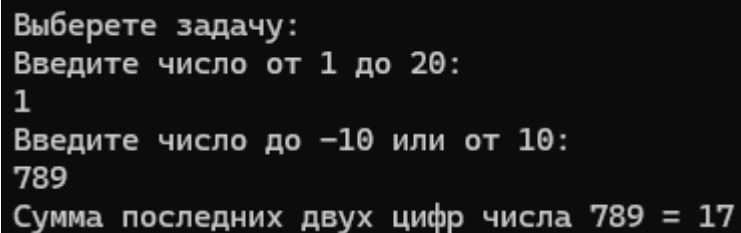
Пример: `x=4568` результат: 14

### Алгоритм решения

В метод передается число `x`. Последний знак заданного числа находится по формуле:  $(x \% 100) \% 10$ . Где `%` - остаток от деления числа `a` на число `b`. Предпоследний знак заданного числа находится по формуле:  $(x \% 100) / 10$ . Эти результаты складываются и если они меньше 0, то метод возвращает противоположное число от их суммы, иначе возвращает сумму последних двух цифр без изменений.

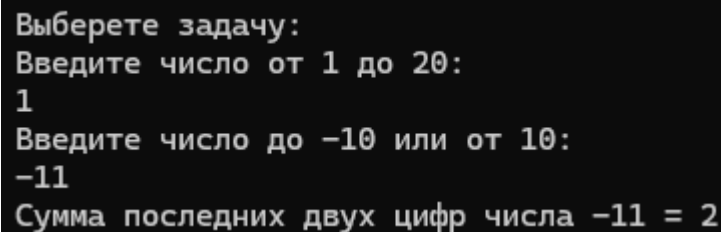
### Тестирование

Рис.1



```
Выберете задачу:  
Введите число от 1 до 20:  
1  
Введите число до -10 или от 10:  
789  
Сумма последних двух цифр числа 789 = 17
```

Рис.2



```
Выберете задачу:  
Введите число от 1 до 20:  
1  
Введите число до -10 или от 10:  
-11  
Сумма последних двух цифр числа -11 = 2
```

Рис.3

```
Выберете задачу:  
Введите число от 1 до 20:  
Таа  
неверный ввод  
введите число от 1 до 20 повторно:  
1  
Введите число до -10 или от 10:  
таа  
неверный ввод  
введите число до -10 или от 10 повторно:  
1234567  
Сумма последних двух цифр числа 1234567 = 13
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 2

### Текст задания

Есть ли позитив.

Дана сигнатура метода: `public bool isPositive (int x)`; Необходимо реализовать метод таким образом, чтобы он принимал число `x` и возвращал `true`, если оно положительное.

Пример 1: `x=3` результат: `true`

Пример 2: `x=-5` результат: `false`

### Алгоритм решения

В метод передается число `x`, возвращается результат выражения `x > 0`.

### Тестирование

Рис.4

```
Выберете задачу:  
Введите число от 1 до 20:  
2  
Введите число:  
-3  
Число -3 - не положительное
```

Рис.5

```
Выберете задачу:  
Введите число от 1 до 20:  
2  
Введите число:  
taa  
неверный ввод  
введите число повторно:  
0  
Число 0 - не положительное
```

Рис.6



```
Выберете задачу:  
Введите число от 1 до 20:  
2  
Введите число:  
52  
Число 52 – положительное.
```

### **Код программы**

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

### Задание 3

#### Текст задания

Большая буква.

Дана сигнатура метода: `public bool isUpperCase (char x)`; Необходимо реализовать метод таким образом, чтобы он принимал символ `x` и возвращал `true`, если это большая буква в диапазоне от 'A' до 'Z'.

Пример 1: `x='D'` результат: `true`

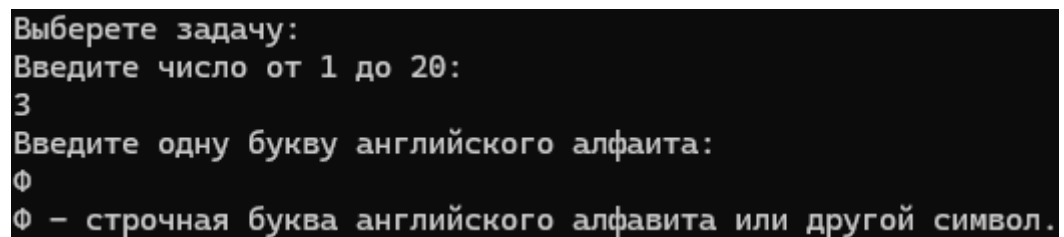
Пример 2: `x='q'` результат: `false`

#### Алгоритм решения

В метод передается переменная `x` символьного типа. Буквы от 'A' до 'Z' в таблице ASCII соответствуют числам в промежутке от 65 до 90, соответственно если `x` больше либо равен 65 и меньше, либо равен 90 метод возвращает `true`, иначе `false`.

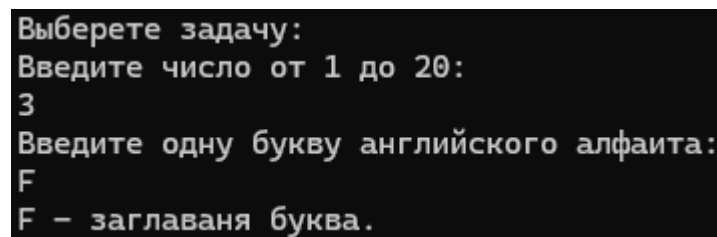
#### Тестирование

Рис.7



```
Выберете задачу:  
Введите число от 1 до 20:  
3  
Введите одну букву английского алфавита:  
Ф  
Ф – строчная буква английского алфавита или другой символ.
```

Рис.8



```
Выберете задачу:  
Введите число от 1 до 20:  
3  
Введите одну букву английского алфавита:  
F  
F – заглавная буква.
```

Рис.9

```
Выберете задачу:  
Введите число от 1 до 20:  
3  
Введите одну букву английского алфавита:  
таа  
неверный ввод, попробуйте ещё раз:  
t  
t – строчная буква английского алфавита или другой символ.
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 4

### Текст задания

Делитель.

Дана сигнатура метода: `public bool isDivisor (int a, int b)`; Необходимо реализовать метод таким образом, чтобы он возвращал `true`, если любое из принятых чисел делит другое нацело.

Пример 1: `a=3 b=6` результат: `true`

Пример 2: `a=2 b=15` результат: `false`

### Алгоритм решения

В метод передаются два числа `a` и `b`. В первую очередь проверяется, что оба числа равны 0, если это так, то возвращается `false`. Далее проверяется есть ли среди этих чисел хотя бы один 0, если есть, то возвращается `true`. Последним действием возвращается результат выражения  $(a \% b = 0)$  или  $(b \% a = 0)$ .

### Тестирование

Рис.10

```
Выберете задачу:  
Введите число от 1 до 20:  
4  
Введите число:  
3  
Введите число:  
6  
одно из двух чисел – делитель другого.
```

Рис.11

```
Выберете задачу:  
Введите число от 1 до 20:  
4  
Введите число:  
0  
Введите число:  
1  
одно из двух чисел – делитель другого.
```

Рис.12

```
Выберете задачу:  
Введите число от 1 до 20:  
4  
Введите число:  
7  
Введите число:  
8  
числа не деляться без остатка друг на друга
```

Рис.13

```
Выберете задачу:  
Введите число от 1 до 20:  
4  
Введите число:  
0  
Введите число:  
0  
числа не деляться без остатка друг на друга
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 5

### Текст задания

Многократный вызов.

Дана сигнатура метода: `public int lastNumSum(int a, int b)` Необходимо реализовать метод таким образом, чтобы он считал сумму цифр двух чисел из разряда единиц. Выполните с его помощью последовательное сложение пяти чисел и результат выведите на экран. Постарайтесь выполнить задачу, используя минимально возможное количество вспомогательных переменных.

Пример: 5+11 это 6 6+123 это 9 9+14 это 13 13+1 это 4 Итого 4

### Алгоритм решения

В метод передается два числа. Что бы найти суммы их последних цифр, нужно у каждого числа взять остаток от деления на 10 и после сложить по модулю.

Что бы последовательно сложить последние цифры пяти чисел, понадобится две переменные а и б. Считываем значения переменных после чего начинается цикл, который будет длиться четыре итерации. В теле цикла присваиваем переменной а результат работы метода `lastNumSum`, затем если сейчас не последняя итерация считываем новое значение переменной б. После выполнения четырех итераций в переменной а будет храниться результат последовательного сложения последних цифр пяти чисел.

### Тестирование

Рис.14

```
Выберете задачу:  
Введите число от 1 до 20:  
5  
Введите число:  
5  
Введите число:  
11  
Введите число:  
123  
Введите число:  
14  
Введите число:  
1  
ответ: 4
```

Рис.15

```
Выберете задачу:  
Введите число от 1 до 20:  
5  
Введите число:  
0  
Введите число:  
-1  
Введите число:  
-1  
Введите число:  
32  
Введите число:  
5  
ответ: 9
```

Рис.16

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 6

### Текст задания

Безопасное деление.

Дана сигнатура метода: `public double safeDiv (int x, int y)`; Необходимо реализовать метод таким образом, чтобы он возвращал деление  $x$  на  $y$ , и при этом гарантировал, что не будет выкинута ошибка деления на 0. При делении на 0 следует вернуть из метода число 0.

Подсказка: смотри ограничения на операции типов данных.

Пример 1:  $x=5$   $y=0$  результат: 0

Пример 2:  $x=8$   $y=2$  результат: 4

### Алгоритм решения

В метод передаются два числа  $x$  – делимое,  $y$  – делитель. Если делитель равен 0, то возвращается 0. Иначе возвращается результат деления  $x$  на  $y$  приведенный к типу данных `double`.

### Тестирование

Рис.17

```
Выберете задачу:  
Введите число от 1 до 20:  
6  
Введите число:  
4  
Введите число:  
3  
4 разделить на 3 = 1,3333333333333333
```

Рис.18

```
Выберете задачу:  
Введите число от 1 до 20:  
6  
Введите число:  
0  
Введите число:  
1  
0 разделить на 1 = 0
```



Рис.19

```
Выберете задачу:  
Введите число от 1 до 20:  
6  
Введите число:  
5  
Введите число:  
0  
деление на 0
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 7

### Текст задания

Строка сравнения.

Дана сигнатура метода: `public String makeDecision (int x, int y);` Необходимо реализовать метод таким образом, чтобы он возвращал строку, которая включает два принятых методом числа и корректно выставленный знак операции сравнения (больше, меньше, или равно).

Пример 1: `x=5 y=7` результат: `"5 < 7"`

Пример 2: `x=8 y=-1` результат: `"8 > -1"`

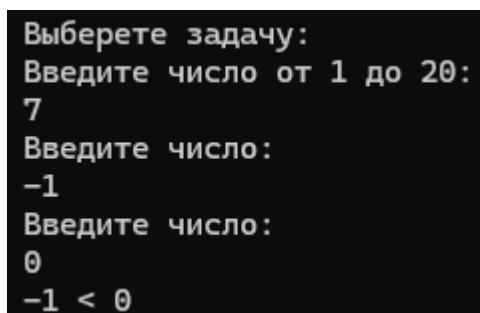
Пример 3: `x=4 y=4` результат: `"4 == 4"`

### Алгоритм решения

В метод передается два числа `x` и `y`. Если `x` больше `y` возвращается строка в формате: `x > y`. Если `x` меньше `y` возвращается строка в формате: `x < y`. Иначе возвращается строка в формате `x == y`.

### Тестирование

Рис.20



```
Выберете задачу:  
Введите число от 1 до 20:  
7  
Введите число:  
-1  
Введите число:  
0  
-1 < 0
```

Рис.21

```
Выберете задачу:  
Введите число от 1 до 20:  
7  
Введите число:  
8  
Введите число:  
4  
8 > 4
```

Рис.22

```
Выберете задачу:  
Введите число от 1 до 20:  
7  
Введите число:  
100  
Введите число:  
100  
100 == 100
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 8

### Текст задания

Тройная сумма.

Дана сигнатура метода: `public bool sum3 (int x, int y, int z);` Необходимо реализовать метод таким образом, чтобы он возвращал `true`, если два любых числа (из трех принятых) можно сложить так, чтобы получить третье.

Пример 1: `x=5 y=7 z=2` результат: `true`

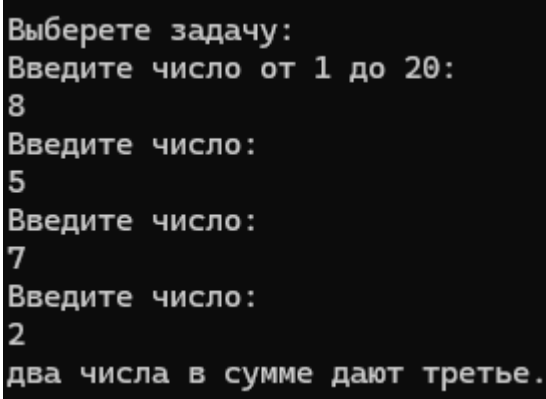
Пример 2: `x=8 y=-1 z=4` результат: `false`

### Алгоритм решения

В метод передается три числа `x`, `y` и `z`. Возвращается результат выражения: сумма `x` и `y` равна `z` или сумма `x` и `z` равна `y` или сумма `y` и `z` равна `x`.

### Тестирование

Рис.23



```
Выберете задачу:  
Введите число от 1 до 20:  
8  
Введите число:  
5  
Введите число:  
7  
Введите число:  
2  
два числа в сумме дают третье.
```

Рис.24

```
Выберете задачу:  
Введите число от 1 до 20:  
8  
Введите число:  
8  
Введите число:  
-1  
Введите число:  
4  
нет пары чисел сумма которых дает третье.
```

Рис.25

```
Выберете задачу:  
Введите число от 1 до 20:  
8  
Введите число:  
2  
Введите число:  
3  
Введите число:  
5  
два числа в сумме дают третье.
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 9

### Текст задания

Возраст.

Дана сигнатура метода: `public String age (int x)`; Необходимо реализовать метод таким образом, чтобы он возвращал строку, в которой сначала будет число `x`, а затем одно из слов: год года лет Слово “год” добавляется, если число `x` заканчивается на 1, кроме числа 11. Слово “года” добавляется, если число `x` заканчивается на 2, 3 или 4, кроме чисел 12, 13, 14. Слово “лет” добавляется во всех остальных случаях.

Подсказка: оператор `%` позволяет получить остаток от деления.

Пример 1: `x=5` результат: “5 лет”

Пример 2: `x=31` результат: “31 год”

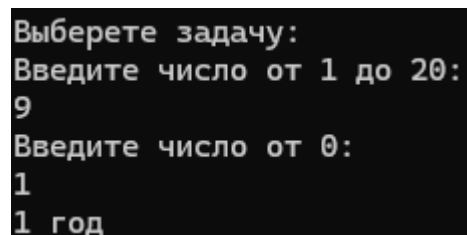
Пример 3: `x=44` результат: “44 года”

### Алгоритм решения

В метод передается число `x`. Инициализируется переменная `s` типа `string`. Далее проверяется если остаток от деления `x` на 10 равен 1 и `x` не равно 11, то переменной `s` присваивается значение “год”. Если остаток от деления `x` на 10 больше 1 и меньше 5, и `x` больше 15 или меньше 11, то переменной `s` присваивается значение “года”. Иначе переменной `s` присваивается значение “лет”. Возвращается строка в формате: `x s`.

### Тестирование

Рис.26



```
Выберете задачу:  
Введите число от 1 до 20:  
9  
Введите число от 0:  
1  
1 год
```

Рис.27

```
Выберете задачу:  
Введите число от 1 до 20:  
9  
Введите число от 0:  
11  
11 лет
```

Рис.28

```
Выберете задачу:  
Введите число от 1 до 20:  
9  
Введите число от 0:  
52  
52 года
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 10

### Текст задания

Вывод дней недели.

Дана сигнатура метода: `public void printDays (String x)`; В качестве параметра метод принимает строку, в которой записано название дня недели. Необходимо реализовать метод таким образом, чтобы он выводил на экран название переданного в него дня и всех последующих до конца недели дней. Если в качестве строки передан не день, то выводится текст “это не день недели”. Первый день понедельник, последний – воскресенье. Вместо `if` в данной задаче используйте `switch`.

Пример 1: `x="четверг"` результат: четверг пятница суббота воскресенье

Пример 2: `x="чг"` результат: это не день недели

### Алгоритм решения

В метод передается переменная `x`. Инициализируется массив строк `days` из семи элементов, каждый из которых равен соответствующему дню недели.

Если значение `x` = понедельник, то в консоль выводятся элементы массива, начиная со второго.

Если значение `x` = вторник, то в консоль выводятся элементы массива, начиная с третьего.

Если значение `x` = среда, то в консоль выводятся элементы массива, начиная с четвертого.

Если значение `x` = четверг, то в консоль выводятся элементы массива, начиная с пятого.

Если значение `x` = пятница, то в консоль выводятся элементы массива, начиная с шестого.

Если значение `x` = суббота, то в консоль выводятся седьмой элемент массива.

Если значение `x` = воскресенье, то в консоль ничего не выводится.

Иначе выводится сообщение об ошибке.



## Тестирование

Рис.29

```
Выберете задачу:  
Введите число от 1 до 20:  
10  
Введите день недели:  
среда  
четверг  
пятница  
суббота  
воскресенье
```

Рис.30

```
Выберете задачу:  
Введите число от 1 до 20:  
10  
Введите день недели:  
пятница  
суббота  
воскресенье
```

Рис.31

```
Выберете задачу:  
Введите число от 1 до 20:  
10  
Введите день недели:  
Сыграем в гвинт?  
это не день недели
```

## Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 11

### Текст задания

Числа наоборот.

Дана сигнатура метода: `public String reverseListNums (int x);` Необходимо реализовать метод таким образом, чтобы он возвращал строку, в которой будут записаны все числа от  $x$  до 0 (включительно).

Пример:  $x=5$  результат: "5 4 3 2 1 0"

### Алгоритм решения

В метод передается число  $x$ . Инициализируется переменная  $s$  типа `string` со значением "значение  $x$ ".

Если  $x$  больше 0, то в цикле начиная с  $x-1$  до 0 с шагом -1 в  $s$  добавляется пробел и строка со значением счетчика цикла.

Если  $x$  меньше 0, то в цикле начиная с  $x+1$  до 0 с шагом 1 в  $s$  добавляется пробел и строка со значением счетчика цикла.

Возвращается переменная  $s$ .

### Тестирование

Рис.32

```
Выберете задачу:  
Введите число от 1 до 20:  
11  
Введите число:  
-8  
Результат: -8 -7 -6 -5 -4 -3 -2 -1 0
```

Рис.33

```
Выберете задачу:  
Введите число от 1 до 20:  
11  
Введите число:  
6  
Результат: 6 5 4 3 2 1 0
```

Рис.34

```
Выберете задачу:  
Введите число от 1 до 20:  
11  
Введите число:  
0  
Результат: 0
```

### **Код программы**

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 12

### Текст задания

Степень числа.

Дана сигнатура метода: `public int pow (int x, int y)`; Необходимо реализовать метод таким образом, чтобы он возвращал результат возведения  $x$  в степень  $y$ .

Подсказка: для получения степени необходимо умножить единицу на число  $x$ , и сделать это  $y$  раз, т.е. два в третьей степени это  $1*2*2*2$

Пример:  $x=2$   $y=5$  результат: 32

### Алгоритм решения

В метод передаются два числа  $x$  и  $y$ . Инициализируется переменная `answer` типа `int` со значением 1. В цикле от 0 до  $y$  не включительно переменная `answer` умножается на  $x$ . Возвращается переменная `answer`.

### Тестирование

Рис.35

```
Выберете задачу:  
Введите число от 1 до 20:  
12  
Введите число:  
3  
Введите число от 0:  
4  
3 в степени 4 = 81.
```

Рис.36

```
Выберете задачу:  
Введите число от 1 до 20:  
12  
Введите число:  
1  
Введите число от 0:  
543  
1 в степени 543 = 1.
```

Рис.37

```
Выберете задачу:  
Введите число от 1 до 20:  
12  
Введите число:  
7  
Введите число от 0:  
0  
7 в степени 0 = 1.
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 13

### Текст задания

Одинаковость.

Дана сигнатура метода: `public bool equalNum (int x)`; Необходимо реализовать метод таким образом, чтобы он возвращал `true`, если все знаки числа одинаковы, и `false` в ином случае.

Подсказки: `int x=123%10;` // x будет иметь значение 3 `int y=123/10;` // y будет иметь значение 12

Пример 1: `x=1111` результат: `true`

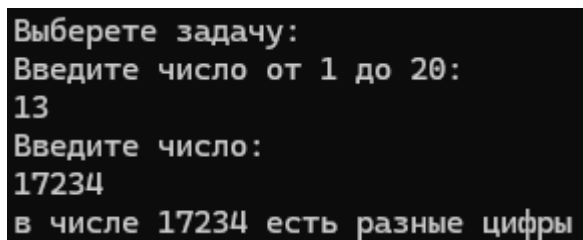
Пример 2: `x=1211` результат: `false`

### Алгоритм решения

В метод передается число `x`. Инициализируется новая переменная `s` типа `string` со значением `x`. Инициализируется переменная `digit` типа `char` со значением последней цифры числа `x`. Возвращается результат выражения `длина строки s равна количеству символов digit в строке s`.

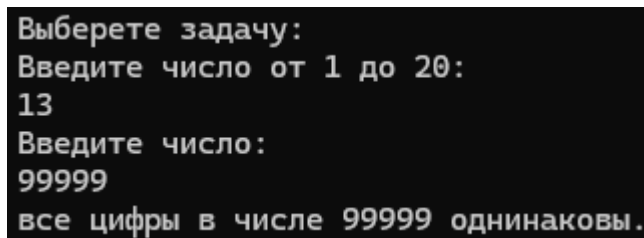
### Тестирование

Рис. 38



```
Выберете задачу:  
Введите число от 1 до 20:  
13  
Введите число:  
17234  
в числе 17234 есть разные цифры
```

Рис. 39



```
Выберете задачу:  
Введите число от 1 до 20:  
13  
Введите число:  
99999  
все цифры в числе 99999 одинаковы.
```

## **Код программы**

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 14

### Текст задания

Левый треугольник.

Дана сигнатура метода: `public void leftTriangle (int x);` Необходимо реализовать метод таким образом, чтобы он выводил на экран треугольник из символов '\*' у которого `x` символов в высоту, а количество символов в ряду совпадает с номером строки.

Пример 1: `x=2` результат:

```
*  
**
```

Пример 2: `x=4` результат:

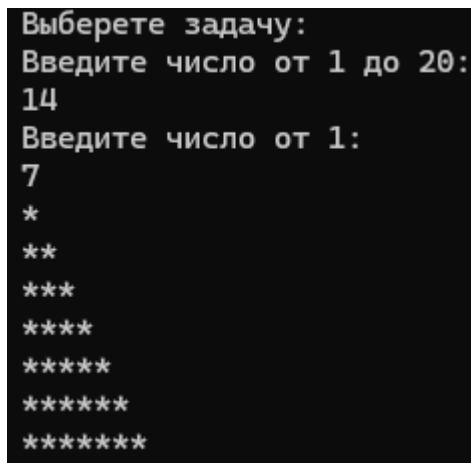
```
*  
**  
***  
****
```

### Алгоритм решения

В метод передается число `x`. В цикле со счетчиком `i` от 0 до `x` не включительно вложен второй цикл от 0 до `i`, в теле которого выводится на консоль "\*", после выполнения вложенного цикла происходит перенос на новую строку.

### Тестирование

Рис.40



```
Выберете задачу:  
Введите число от 1 до 20:  
14  
Введите число от 1:  
7  
*  
**  
***  
****  
*****  
*****  
*****
```



Рис.41

```
Выберете задачу:  
Введите число от 1 до 20:  
14  
Введите число от 1:  
5  
*  
**  
***  
****  
*****
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 15

### Текст задания

Угадайка.

Дана сигнатура метода: `public void guessGame()` Необходимо реализовать метод таким образом, чтобы он генерировал случайное число от 0 до 9, далее считывал с консоли введенное пользователем число и выводил, угадал ли пользователь то, что было загадано, или нет. Метод запускается до тех пор, пока пользователь не угадает число. После этого выведите на экран количество попыток, которое потребовалось пользователю, чтобы угадать число.

Пример: Введите число от 0 до 9: 5 Вы не угадали, введите число от 0 до 9: 9 Вы угадали! Вы отгадали число за 2 попытки

### Алгоритм решения

Создается объект класса `Random`, переменной `guessNum` присваивается случайное значение от 0 до 10 не включительно, с помощью метода класса `Random Next`. Инициализируется переменная `userNum` со значением -1, так как оно не может быть в переменной `guessNum`. Инициализируется переменная счетчик `i`. Пока значения переменных `guessNum` и `userNum` не совпадут: считываем с клавиатуры число от 0 до 9 и записываем в переменную `userNum`, увеличиваем счетчик на 1, если числа совпали, выводим информацию об этом в консоль, иначе выводим информацию что пользователь не угадал. После выполнения цикла выводим за сколько попыток пользователь угадал число.

### Тестирование

Рис. 42

```
Выберете задачу:  
Введите число от 1 до 20:  
15  
Введите число от 0 до 9:  
5  
Вы не угадали.  
Введите число от 0 до 9:  
4  
Вы не угадали.  
Введите число от 0 до 9:  
8  
Вы не угадали.  
Введите число от 0 до 9:  
3  
Вы угадали!  
Вы угадали за 4 попытки
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>

## Задание 16

### Текст задания

Поиск последнего значения.

Дана сигнатура метода: `public int findLast (int[] arr, int x)`; Необходимо реализовать метод таким образом, чтобы он возвращал индекс последнего вхождения числа `x` в массив `arr`. Если число не входит в массив – возвращается `-1`.

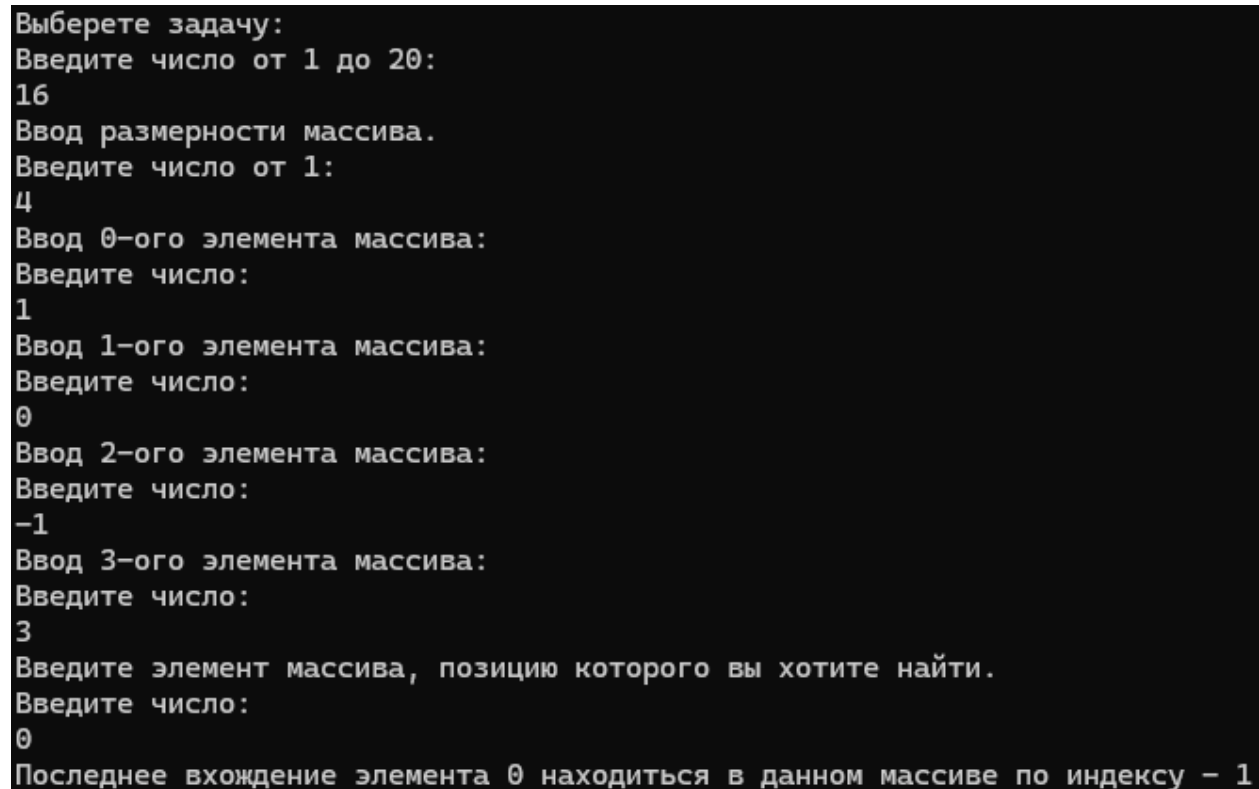
Пример: `arr=[1,2,3,4,2,2,5]` `x=2` результат: `5`

### Алгоритм решения

В метод передается массив целых чисел `arr` и число `x`. Инициализируем переменную `index` со значением `-1`, в цикле с конца массива сравниваем его элементы с `x`, если элемент массива совпал с числом `x`, то переменной `index` присваиваем значение счетчика и выходим из цикла, после чего возвращаем переменную `index`.

### Тестирование

Рис. 43



```
Выберете задачу:  
Введите число от 1 до 20:  
16  
Ввод размерности массива.  
Введите число от 1:  
4  
Ввод 0-ого элемента массива:  
Введите число:  
1  
Ввод 1-ого элемента массива:  
Введите число:  
0  
Ввод 2-ого элемента массива:  
Введите число:  
-1  
Ввод 3-ого элемента массива:  
Введите число:  
3  
Введите элемент массива, позицию которого вы хотите найти.  
Введите число:  
0  
Последнее вхождение элемента 0 находится в данном массиве по индексу - 1
```

Рис.44

```
Выберете задачу:
Введите число от 1 до 20:
16
Ввод размерности массива.
Введите число от 1:
3
Ввод 0-ого элемента массива:
Введите число:
1
Ввод 1-ого элемента массива:
Введите число:
2
Ввод 2-ого элемента массива:
Введите число:
1
Введите элемент массива, позицию которого вы хотите найти.
Введите число:
1
Последнее вхождение элемента 1 находится в данном массиве по индексу - 2
```

Рис. 45

```
Выберете задачу:
Введите число от 1 до 20:
16
Ввод размерности массива.
Введите число от 1:
4
Ввод 0-ого элемента массива:
Введите число:
1
Ввод 1-ого элемента массива:
Введите число:
1
Ввод 2-ого элемента массива:
Введите число:
1
Ввод 3-ого элемента массива:
Введите число:
1
Введите элемент массива, позицию которого вы хотите найти.
Введите число:
-1
Элемента -1 нет в массиве.
```

## Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 17

### Текст задания

Добавление в массив.

Дана сигнатура метода: `public int[] add (int[] arr, int x, int pos);` Необходимо реализовать метод таким образом, чтобы он возвращал новый массив, который будет содержать все элементы массива `arr`, однако в позицию `pos` будет вставлено значение `x`.  
Пример: `arr=[1,2,3,4,5]` `x=9` `pos=3` результат: `[1,2,3,9,4,5]`

### Алгоритм решения

В метод передается массив целых чисел `arr`, и числа `x` и `pos`. Инициализируется новый массив `newArr` размерность которого на 1 больше размерности `arr`. Инициализируется переменная `isAdd` со значением 0. В цикле от 0 до размерности массива `arr` если счетчик равен `pos`, то в эту позицию в массиве `newArr` записывается `x`, а переменная `isAdd` увеличивается на 1. В каждой итерации цикла в значение из массива `arr` записывается в массив `newArr` по соответствующему индексу со сдвигом на `isAdd`. После цикла возвращается массив `newArr`.

### Тестирование

Рис.46

```
Выберете задачу:
Введите число от 1 до 20:
17
Ввод размерности массива.
Введите число от 1:
4
Ввод 0-ого элемента массива:
Введите число:
1
Ввод 1-ого элемента массива:
Введите число:
2
Ввод 2-ого элемента массива:
Введите число:
3
Ввод 3-ого элемента массива:
Введите число:
4
Ввод элемента для вставки в массив.
Введите число:
-1
Ввод индекса вставки в массив.
Введите число от 0 до 3:
2
1 2 -1 3 4
```

Рис.47

```
Выберете задачу:
Введите число от 1 до 20:
17
Ввод размерности массива.
Введите число от 1:
3
Ввод 0-ого элемента массива:
Введите число:
0
Ввод 1-ого элемента массива:
Введите число:
0
Ввод 2-ого элемента массива:
Введите число:
0
Ввод элемента для вставки в массив.
Введите число:
4
Ввод индекса вставки в массив.
Введите число от 0 до 2:
0
4 0 0 0
```

## **Код программы**

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>



## Задание 18

### Текст задания

Реверс.

Дана сигнатура метода: `public void reverse (int[] arr)`; Необходимо реализовать метод таким образом, чтобы он изменял массив `arr`. После проведенных изменений массив должен быть записан задом-наперед.

Пример: `arr=[1,2,3,4,5]` результат: `arr=[5,4,3,2,1]`

### Алгоритм решения

В метод передается массив `arr`. В цикле с конца массива до его середины в переменную `buffer` записывается значение элемента, расположенного по индексу: индекс последнего минус индекс текущего. Элементу, расположенному по этому индексу, присваивается значение элемента, расположенного по текущему индексу. Элементу с текущим индексом присваивается значение `buffer`.

### Тестирование

Рис.48

```
Выберете задачу:
Введите число от 1 до 20:
18
Ввод размерности массива.
Введите число от 1:
5
Ввод 0-ого элемента массива:
Введите число:
1
Ввод 1-ого элемента массива:
Введите число:
2
Ввод 2-ого элемента массива:
Введите число:
3
Ввод 3-ого элемента массива:
Введите число:
4
Ввод 4-ого элемента массива:
Введите число:
5
5 4 3 2 1
```

Рис.49

```
Выберете задачу:
Введите число от 1 до 20:
18
Ввод размерности массива.
Введите число от 1:
4
Ввод 0-ого элемента массива:
Введите число:
1
Ввод 1-ого элемента массива:
Введите число:
2
Ввод 2-ого элемента массива:
Введите число:
3
Ввод 3-ого элемента массива:
Введите число:
4
4 3 2 1
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 19

### Текст задания

Объединение.

Дана сигнатура метода: `public int[] concat (int[] arr1,int[] arr2);` Необходимо реализовать метод таким образом, чтобы он возвращал новый массив, в котором сначала идут элементы первого массива (`arr1`), а затем второго (`arr2`).

Пример: `arr1=[1,2,3]` `arr2=[7,8,9]` результат: `[1,2,3,7,8,9]`

### Алгоритм решения

В метод передается два массива `arr1` и `arr2`. Создается новый массив `newArr` размерность которого равна сумме размерности `arr1` и `arr2`. В двух последовательных циклах элементам `newArr` присваиваются значения соответствующих элементов `arr1`, а затем `arr2`, со сдвигом на размерность `arr1`. Возвращается `newArr`.

### Тестирование

Рис.50

```
Выберете задачу:  
Введите число от 1 до 20:  
19  
Ввод размерности массива.  
Введите число от 1:  
3  
Ввод 0-ого элемента массива:  
Введите число:  
1  
Ввод 1-ого элемента массива:  
Введите число:  
2  
Ввод 2-ого элемента массива:  
Введите число:  
3  
Ввод размерности массива.  
Введите число от 1:  
2  
Ввод 0-ого элемента массива:  
Введите число:  
4  
Ввод 1-ого элемента массива:  
Введите число:  
5  
1 2 3 4 5
```

### Код программы

<https://github.com/VladimirShatrov/Labs/blob/lab1/Laba1/Program.cs>

## Задание 20

### Текст задания

Удалить негатив.

Дана сигнатура метода: `public int[] deleteNegative (int[] arr);` Необходимо реализовать метод таким образом, чтобы он возвращал новый массив, в котором записаны все элементы массива `arr` кроме отрицательных.

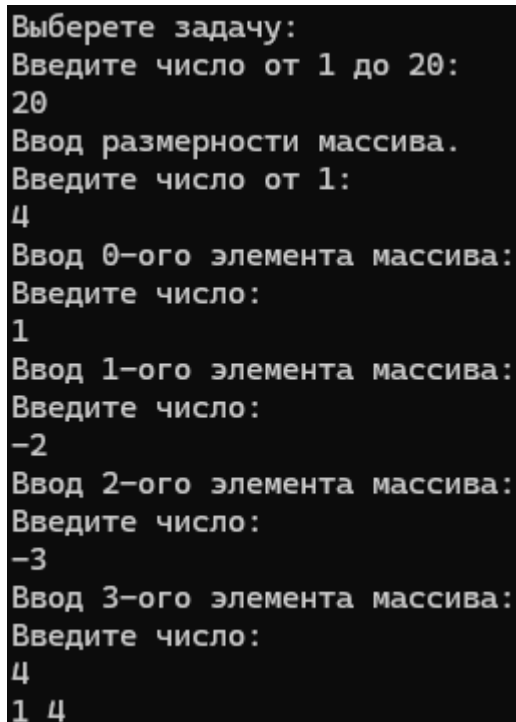
Пример: `arr=[1,2,-3,4,-2,2,-5]` результат: `[1,2,4,2]`

### Алгоритм решения

В метод передается массив `arr`. Сначала идет подсчет не отрицательных чисел в этом массиве. Затем создается новый массив `newArr` с размерностью равной количеству не отрицательных чисел в массиве `arr`. В цикле с начала массива `arr`, если очередной элемент больше либо равен 0, то он записывается в новый массив.

### Тестирование

Рис.51



```
Выберете задачу:  
Введите число от 1 до 20:  
20  
Ввод размерности массива.  
Введите число от 1:  
4  
Ввод 0-ого элемента массива:  
Введите число:  
1  
Ввод 1-ого элемента массива:  
Введите число:  
-2  
Ввод 2-ого элемента массива:  
Введите число:  
-3  
Ввод 3-ого элемента массива:  
Введите число:  
4  
1 4
```

## **Код программы**

<https://github.com/VladimirShatrov/Labs/blob/lab1/Lab1/Program.cs>