

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ITMO University

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

По дисциплине Объектно-ориентированное программирование

Обучающийся Зорина Яна Сергеевна

Факультет Факультет инфокоммуникационных технологий

Группа K3222

Направление подготовки 11.03.02 Инфокоммуникационные технологии и
системы связи

Образовательная программа Программирование в
инфокоммуникационных системах

Обучающийся	<u>14.11.2023</u> (дата)	<u> </u> (подпись)	<u>Зорина Я.С.</u> (Ф.И.О.)
Руководитель	<u> </u> (дата)	<u> </u> (подпись)	<u>Васильев С.Ю.</u> (Ф.И.О.)

Санкт Петербург

2023

СОДЕРЖАНИЕ

Введение.....	3
ГЛАВА 1. Ход работы	4
1.1 Упражнение 1	4
1.2 Упражнение 2	4
1.3 Упражнение 3	5
1.4 Упражнение 4	6
1.5 Упражнение 5	7

ВВЕДЕНИЕ

Целью лабораторной работы является знакомство с языком программирования С# и программой «Visual Studio», изучение структуры программы на данном языке, освоение компиляции и отладки, изучение и приобретение навыков работы с методами класса.

Отчёт содержит одну главу, каждая подглава посвящена одному упражнению из лабораторной работы.

Для выполнения лабораторной работы использовалась программа «Visual Studio».

По этой [ссылке](#) доступен репозиторий, в котором каждой лабораторной работе отведена папка, в которых доступны программы для упражнений.

ГЛАВА 1. ХОД РАБОТЫ

1.1 Упражнение 1

Консольное приложение в пространстве имён "ex_1". Внутри этого пространства имён находится класс "Program" с методом "Main", который принимает от пользователя два целых числа, x и y. Затем он вызывает метод "Greater" из класса "Utils" для определения большего из двух введённых чисел и выводит результат. Класс "Utils" содержит статический метод под названием "Greater", который принимает два целых числа в качестве входных данных, сравнивает их и возвращает большее из двух. Результат выполнения программы представлен на рисунке 1.

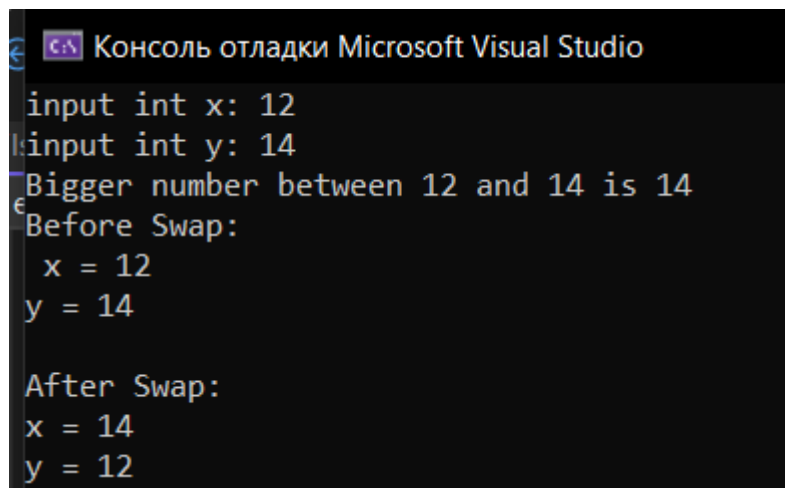


```
Консоль отладки Microsoft Visual Studio
input int x: 13
input int y: 17
Bigger number between 13 and 17 is 17
```

Рисунок 1. Упражнение 1

1.2 Упражнение 2

В этом упражнении в класс "Utils" добавлен метод "Swap", который меняет местами значения двух переменных. Результат выполнения программы представлен на рисунке 2.

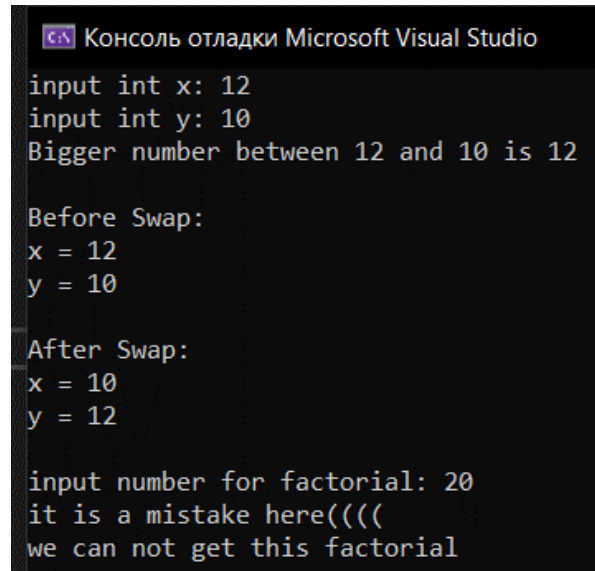


```
Консоль отладки Microsoft Visual Studio
input int x: 12
input int y: 14
Bigger number between 12 and 14 is 14
Before Swap:
  x = 12
  y = 14
After Swap:
  x = 14
  y = 12
```

Рисунок 2. Упражнение 2

1.3 Упражнение 3

В этом упражнении в класс “Utils” был добавлен метод “Factorial”, который считает факториал введённого числа. Результаты выполнения программы при двух сценариях (с ошибкой и без) представлены на рисунках 3 и 4.



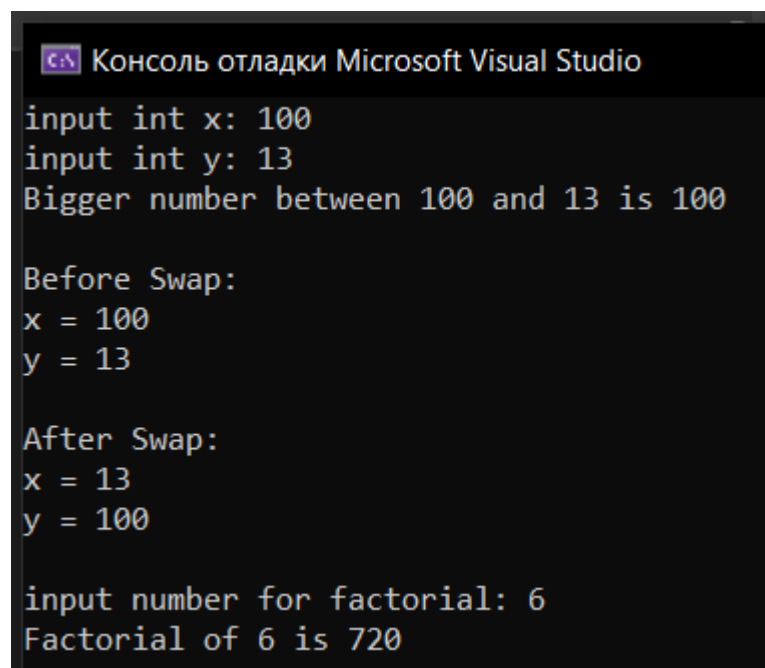
```
Консоль отладки Microsoft Visual Studio
input int x: 12
input int y: 10
Bigger number between 12 and 10 is 12

Before Swap:
x = 12
y = 10

After Swap:
x = 10
y = 12

input number for factorial: 20
it is a mistake here(((
we can not get this factorial
```

Рисунок 3. Упражнение 3 с ошибкой



```
Консоль отладки Microsoft Visual Studio
input int x: 100
input int y: 13
Bigger number between 100 and 13 is 100

Before Swap:
x = 100
y = 13

After Swap:
x = 13
y = 100

input number for factorial: 6
Factorial of 6 is 720
```

Рисунок 4. Упражнение 3 без ошибки

1.4 Упражнение 4

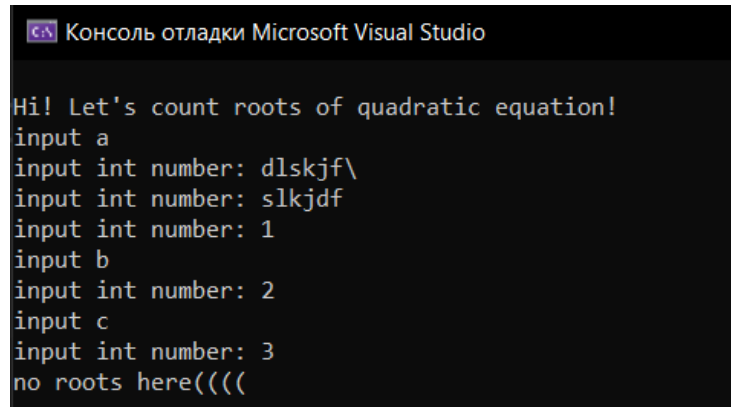
В этом упражнении требуется написать класс “Operation”, который считает площадь треугольника (с обработкой всех ошибок). Требуется выбрать вид треугольника (равносторонний или разносторонний), а далее ввести сторону (стороны). Результат работы программы представлен на рисунке 5.

```
Hi! Let's count triangle's square!
choose triangle type:
1 - equilateral
2 - versatile
your choice: 2
Cool! Let's get sides of triangle!
side 1
input natural number: -1
input natural number: k1
input natural number: 2
side 2
input natural number: 2
side 3
input natural number: 4
this triangle does not not exist!
side 1
input natural number: 2
side 2
input natural number: 3
side 3
input natural number: 3
Great! Count square)
Square of this triangle is 2,8284271247461903
```

Рисунок 5. Упражнение 4

1.5 Упражнение 5

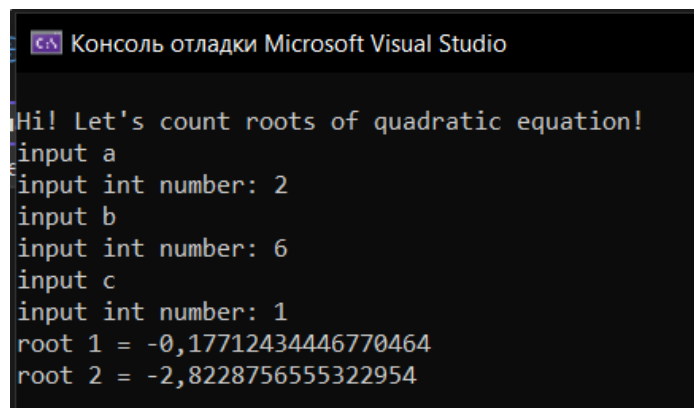
В данном упражнении был добавлен класс “SquareRoots”, который позволяет вычислить корни квадратного уравнения. Пример выполнения программы представлен на рисунках 6 и 7.



```
Консоль отладки Microsoft Visual Studio

Hi! Let's count roots of quadratic equation!
input a
input int number: dlskjf\
input int number: slkjdf
input int number: 1
input b
input int number: 2
input c
input int number: 3
no roots here(((
```

Рисунок 6. Упражнение 5. Пример без корней



```
Консоль отладки Microsoft Visual Studio

Hi! Let's count roots of quadratic equation!
input a
input int number: 2
input b
input int number: 6
input c
input int number: 1
root 1 = -0,17712434446770464
root 2 = -2,8228756555322954
```

Рисунок 7. Упражнение 5. Пример с корнями