

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

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3**

**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

<b>Обучающийся</b>	<u>24.10.2023</u> (дата)	<u>                    </u> (подпись)	<u>Зорина Я.С.</u> (Ф.И.О.)
<b>Руководитель</b>	<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
1.1.1 Задание 1 .....	4
1.1.2 Задание 2 .....	4
1.2 Упражнение 2 .....	7
1.2.1 Задание 1 .....	7
1.2.2 Задание 2 .....	8
1.2.3 Задание 3 .....	9

## **ВВЕДЕНИЕ**

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

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

Для выполнения лабораторной работы использовалась программа «Visual Studio».

## ГЛАВА 1. ХОД РАБОТЫ

### 1.1 Упражнение 1

#### 1.1.1 Задание 1

Эта программа представляет собой простое консольное приложение на языке программирования C#. Она принимает входные значения "x" и "y" от пользователя, затем в зависимости от условий, проверяет находятся ли эти значения внутри, вне или на границе заданной области.

- Если "x" и "y" находятся внутри окружности с радиусом 3 и "y" положителен, программа выводит сообщение "внутри".

- Если "x" и "y" находятся вне окружности с радиусом 3 или "y" отрицателен, программа выводит сообщение "вне".

- Если "x" и "y" находятся на границе окружности с радиусом 3, программа выводит сообщение "на границе".

Программа использует условные операторы if-else для определения расположения точки относительно заданной области и выводит соответствующий результат на экран консоли. Результат запуска представлен на рисунке 1.

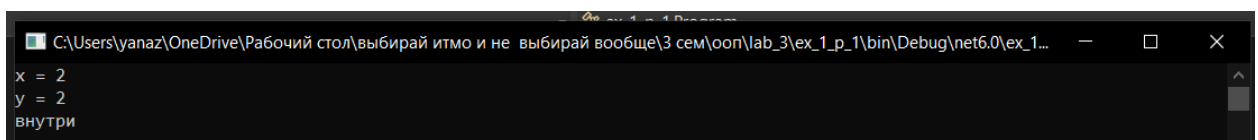


Рисунок 1. Задание 1 в упражнении 1

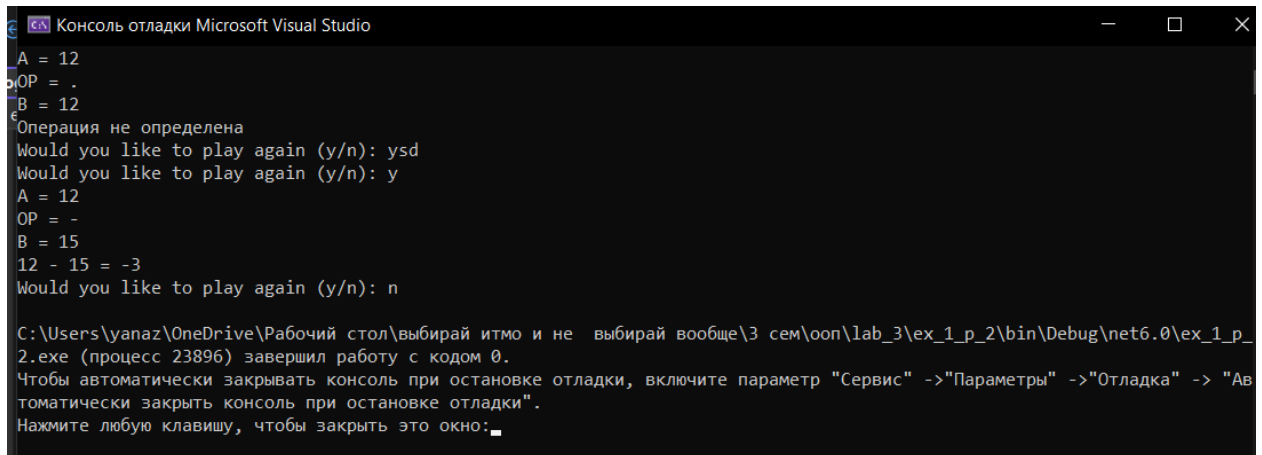
#### 1.1.2 Задание 2

Программа выполняет простые арифметические операции с двумя числами, вводимыми пользователем. Однако она не будет обрабатывать ситуации, когда происходит деление на ноль или вводится неправильный символ операции.

Проверка деления на ноль: Если пользователь вводит символ операции "/", программа выполнит деление первого числа (a) на второе число (b). Если второе число равно нулю ( $b = 0$ ), то вместо результата программа выведет вопросительный знак. Это происходит потому, что в математике деление на ноль не определено.

Неправильный символ операции: Если пользователь вводит символ операции, отличный от "+", "-", "\*", "/" или ":", программа присваивает переменной ok значение false, что означает, что операция не определена. Затем программа выводит сообщение "Операция не определена".

Результат представлен на рисунке 2.



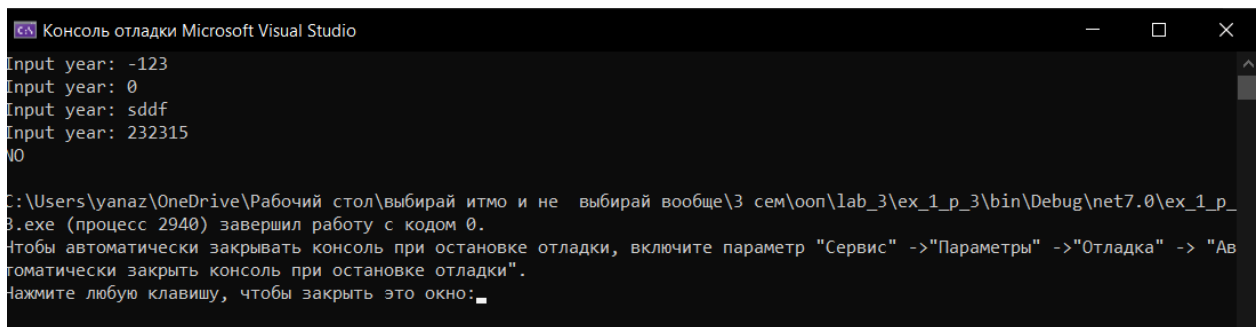
```
Консоль отладки Microsoft Visual Studio
A = 12
b,OP = .
B = 12
Операция не определена
Would you like to play again (y/n): ysd
Would you like to play again (y/n): y
A = 12
OP = -
B = 15
12 - 15 = -3
Would you like to play again (y/n): n

C:\Users\yanaz\OneDrive\Рабочий стол\выбирай итмо и не выбирай вообще\3 сем\oop\lab_3\ex_1_p_2\bin\Debug\net6.0\ex_1_p_2.exe (процесс 23896) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно: █
```

Рисунок 2. Результат выполнения операций

### 1.1.3 Задание 3

Программа простого консольного приложения, которое запрашивает у пользователя ввод года. Затем она проверяет введенное значение, чтобы убедиться, что оно является натуральным числом. Если ввод корректен, программа определяет, является ли год високосным, и выводит "YES" или "NO" в зависимости от результата. Результат выполнения программы представлен на рисунке 3.



```
Консоль отладки Microsoft Visual Studio
Input year: -123
Input year: 0
Input year: sddf
Input year: 232315
NO

C:\Users\yanaz\OneDrive\Рабочий стол\выбирай итмо и не выбирай вообще\3 сем\oop\lab_3\ex_1_p_3\bin\Debug\net7.0\ex_1_p_3.exe (процесс 2940) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно: _
```

Рисунок 3. Результат проверки года на високосность

## 1.2 Упражнение 2

### 1.2.1 Задание 1

Данный код предлагает пользователю выбрать одну из трех опций: "cycle thing", "sin thing" или "evklid thing".

При выборе опции "cycle thing", программа запрашивает у пользователя ввод числа `n`, а затем использует цикл `while`, `do while` и `for` для вывода на консоль нечетных чисел от 1 до `n`. После завершения вывода программа возвращает пользователя в основное меню.

При выборе опции "sin thing", программа запрашивает у пользователя ввод чисел `x1` и `x2`. Затем с помощью цикла `do while` программа вычисляет значения синуса для значений `x` от `x1` до `x2` с шагом 0,01 и выводит их на консоль. Результаты округляются до двух знаков после запятой. После завершения вывода программа возвращает пользователя в основное меню.

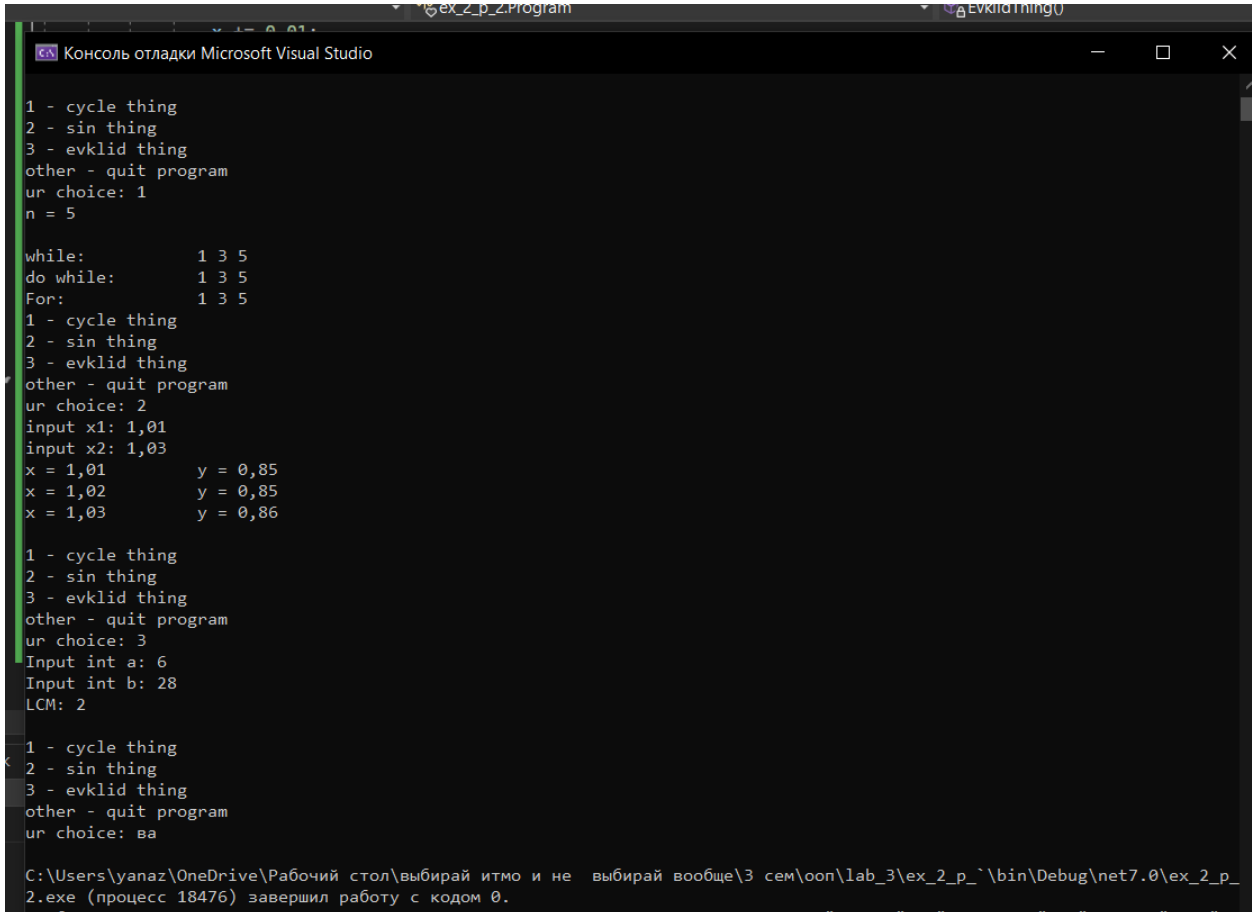
При выборе опции "evklid thing", программа запрашивает у пользователя ввод двух целых чисел `a` и `b`. Затем с помощью цикла `while` программа находит наименьшее общее кратное (НОК) этих чисел с помощью алгоритма Евклида. Результат выводится на консоль. После вывода результата программа возвращает пользователя в основное меню.

Цикл `Main` повторяется бесконечно, позволяя пользователю выбирать и выполнять определенные опции. Если пользователь вводит значение, отличное от 1, 2 или 3, программа завершается.

Выбор типа цикла зависит от конкретной задачи и требований к решению. В данном случае, использование разных типов циклов было обусловлено диапазоном значений, которые должны были быть обработаны, и удобством в использовании. Цикл `while` был выбран для обработки диапазона значений, когда заранее нельзя определить точное количество итераций. Цикл `do while` был использован для выполнения хотя бы одной

итерации и продолжения выполнения в зависимости от условия. Цикл for был выбран для обработки диапазона значений с явными условиями начала, окончания и шага выполнения.

Результат выполнения программы представлен на рисунке 4.



```
1 - cycle thing
2 - sin thing
3 - evklid thing
other - quit program
ur choice: 1
n = 5

while:      1 3 5
do while:   1 3 5
For:        1 3 5
1 - cycle thing
2 - sin thing
3 - evklid thing
other - quit program
ur choice: 2
input x1: 1,01
input x2: 1,03
x = 1,01      y = 0,85
x = 1,02      y = 0,85
x = 1,03      y = 0,86

1 - cycle thing
2 - sin thing
3 - evklid thing
other - quit program
ur choice: 3
Input int a: 6
Input int b: 28
LCM: 2

1 - cycle thing
2 - sin thing
3 - evklid thing
other - quit program
ur choice: va

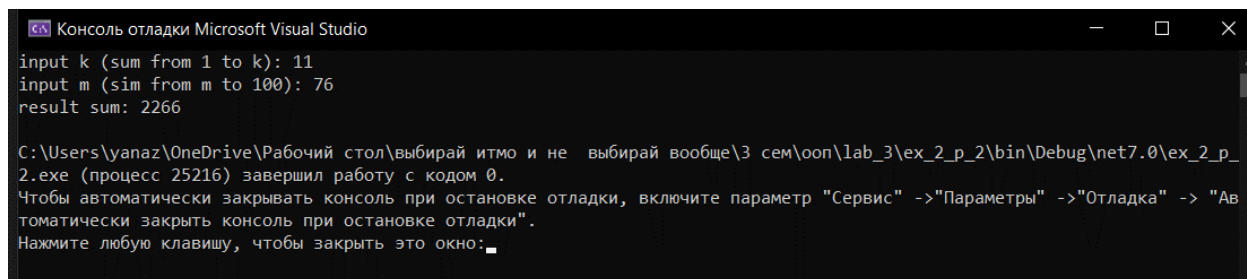
C:\Users\yanaz\OneDrive\Рабочий стол\выбирай итмо и не выбирай вообще\3 сем\oop\lab_3\ex_2_p_`bin\Debug\net7.0\ex_2_p_
2.exe (процесс 18476) завершил работу с кодом 0.
Иногда автоматически закрываются консоли при отладке. Чтобы избежать этого, выберите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрывать консоли при отладке" и отключите его.
```

Рисунок 4. Результат выполнения всех трёх опций

### 1.2.2 Задание 2

Данный код запрашивает у пользователя ввод двух чисел `k` и `m`. Затем программа выполняет цикл от 1 до 100, пропуская числа, которые находятся между `k` и `m`. Каждое число, не попадающее в этот диапазон, добавляется к переменной `result`. По завершению цикла, программа выводит сумму всех чисел, которые не находятся между `k` и `m`, на консоль. Результат выполнения программы представлен на рисунке 5.





```
Консоль отладки Microsoft Visual Studio
input k (sum from 1 to k): 11
input m (sim from m to 100): 76
result sum: 2266

C:\Users\yanaz\OneDrive\Рабочий стол\выбирай итмо и не выбирай вообще\3 сем\oop\lab_3\ex_2_p_2\bin\Debug\net7.0\ex_2_p_2.exe (процесс 25216) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно: █
```

Рисунок 5. Результат суммирования по заданным условиям

### 1.2.3 Задание 3

Программа, представленная в задании, имитирует стрельбу по мишени. Она позволяет пользователю вводить координаты выстрела (x и y) несколько раз. После каждого выстрела выводится информация о сумме набранных очков.

Функциональность программы включает:

- Повторный ввод данных о выстреле в цикле.
- Использование случайного значения для определения центра мишени, чтобы стрелок не знал точного местоположения мишени.
- Введение случайной помехи при выстреле, чтобы усложнить задачу для стрелка.

Программа также задает некоторые константы, определяющие диапазон координат и значения очков за попадание.

Количество попыток стрельбы ограничено числом 5. После окончания всех попыток выводятся полученные очки, а затем игроку предлагается продолжить или завершить игру.

Игрок получает очки в зависимости от расстояния между точкой выстрела и центром мишени. Если расстояние меньше определенного значения, игрок получает соответствующее количество очков.

Наконец, программа отслеживает общее количество очков игрока и выводит его после каждого раунда.

Результат работы программы представлен на рисунке 6.

```
Hi, the best shooter in the world!
You have 5 attempts to hit the target!
Min x and y: -3
Max x and y: 3
The gun is a little old, so you have a little hindrance when shooting (-0.5, +0, +0.5)
After all of them u see points and can finish game or play again!
input option (1 or 2): 2
Attempt 1/5
input x: 5
input x: ысно
input x: 2y
input x: 2
input y: -2
Attempt 2/5
input x: 1
input y: -1
Attempt 3/5
input x: 3
input y: 2
Attempt 4/5
input x: 1
input y: -2
Attempt 5/5
input x: -2
input y: 2
You are out of attempts(( Here are your points:
Attempt 1: 5 points
Attempt 2: 10 points
Attempt 3: 0 points
Attempt 4: 5 points
Attempt 5: 0 points
sum of your points: 20
Now you have 20 points

Would you like to continue game (y/n): ыпр
Would you like to continue game (y/n): 32
Would you like to continue game (y/n): y
Would you like to continue game (y/n): н
Would you like to continue game (y/n): y
Hi, the best shooter in the world!
You have 5 attempts to hit the target!
Min x and y: -3
Max x and y: 3
The gun is a little old, so you have a little hindrance when shooting (-0.5, +0, +0.5)
After all of them u see points and can finish game or play again!
input option (1 or 2): 2
Attempt 1/5
input x: 1
input y: 1
Attempt 2/5
input x: 3
input y: 3
Attempt 3/5
input x: -1
input y: 2
Attempt 4/5
input x: -2
input y: 3
Attempt 5/5
input x: 2
input y: -3
You are out of attempts(( Here are your points:
Attempt 1: 1 points
Attempt 2: 0 points
Attempt 3: 0 points
Attempt 4: 0 points
Attempt 5: 1 points
sum of your points: 2
Now you have 22 points

Would you like to continue game (y/n): n
```

Рисунок 6. Результат игры со стрельбой