

### Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

# «Московский государственный технический университет имени Н.Э. Баумана

(национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)

#### ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

направление подготовки 09.03.01 Информатика и вычислительная техника

## ОТЧЕТ по лабораторной работе № 5

**Название:** Создание консольного приложения Ruby.

Дисциплина: Языки интернет-программирования

Студент	ИУ6		
	(Группа)	(Подпись, дата)	(И.О. Фамилия)
Преподаватель			
		(Подпись, дата)	(И.О. Фамилия)

#### Лабораторная работа №4

#### Часть 1:

end end

```
Вычислить: a = \frac{2 \cdot \cos(x - \frac{\pi}{6})}{\frac{1}{2} + \sin^2(y)}
```

```
Код в console.rb:
```

```
# frozen string literal: true
require relative 'main'
print 'x: '
x num = gets.to f
print 'y: '
y num = gets.to f
puts "Result: #{maths x num, y num}"
Код в main.rb:
# frozen_string_literal: true
def maths(x num, y num)
 (2 * Math.cos(x num - Math::PI / 6.0)) / (1 / 2.0 + Math.sin(y num)**2)
end
Код в test.rb:
# frozen string literal: true
require 'minitest/autorun'
require 'minitest/assertions'
require_relative 'main'
# class with test
class Testing < Minitest::Test</pre>
  def setup
   super
   @del = 0.001
   @fail = 'Error'
  end
  def test 1
    x num = rand(-10..10)
    y \ num = rand(-10..10)
    assert in delta((2 * Math.cos(x num - Math::PI / 6.0)) / (1 / 2.0 +
Math.\sin(y_num)**2), maths(x_num, y_num), @del, @fail + '2')
  end
  def test_2
   x_num = 1
   y_num = 1
   res = 1.471187100994271
    assert_in_delta(res, maths(x_num, y_num),
                     @del, @fail + ' 1')
```

Результат выполнения и результаты проверки анализаторами rubocop и reek:

#### Часть 2:

Дана последовательность строк. Каждая строка состоит из слов, разделенных пробелами. Написать программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в обмене местами слов максимальной и минимальной длины в каждой строке. Если среди слов строки есть несколько таких слов, то в обмене участвуют только первые обнаруженные слова. Вывести на печать исходную и скорректированную последовательности строк.

Автоматический тест программы обязательно должен генерировать случайные строки в соответствии с правилами, перечисленными в задании.

#### Код в console.rb:

```
# frozen_string_literal: true
require_relative 'main'
ARGF.each do |line|
  puts output(line.chomp)
end
```

#### Код в main.rb:

```
# frozen_string_literal: true

def output(str)
    arr = str.split
    minmax = arr.each_with_index.minmax_by { |word, _index| word.length }
    arr[minmax[0][1]] = minmax[1][0]
    arr[minmax[1][1]] = minmax[0][0]
    arr.join ' '
end
```

#### Код в test.rb:

```
# frozen_string_literal: true

require 'minitest/autorun'
require 'minitest/assertions'
require_relative 'main'

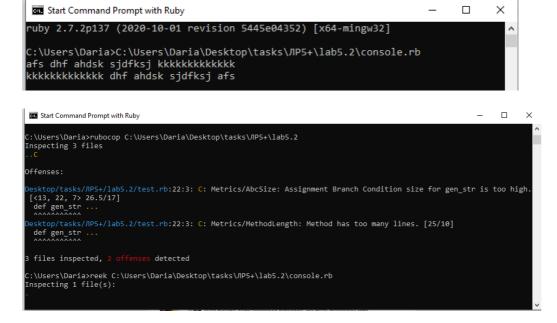
# class with test
class Testing < Minitest::Test
    CHARS = ('a'..'\(\frac{1}{3}\)).to_a

def test_1
    10.times do
        str = gen_str
        assert_equal str[1], output(str[0])
    end
end

def gen word(len word)</pre>
```

```
(CHARS.sample len word).join ''
  end
  def gen str
    max \ size = rand(7...10)
    min size = rand(2...6)
    max word = gen word max size
    min word = gen word min size
    str\ length = rand(3..10)
    bool1 = false
    bool2 = false
    arr input = []
    arr result = []
    str length.times do |i|
      if i == rand(i...str length - 1) && !bool2
        arr input << [min word]
        arr result << [max word]</pre>
        bool2 = true
      elsif i == rand(i...str length) && !bool1
        arr input << [max word]</pre>
        arr result << [min word]</pre>
        bool1 = true
      else
        word = gen word rand((min size + 1)...max size)
        arr input << word
        arr result << word
      end
    end
    [arr_input.join(' '), arr result.join(' ')]
  end
end
```

Результат выполнения и результаты проверки анализаторами rubocop и reek:



**Вывод:** в ходе данной лабораторной работы я изучила основные принципы программирования на языке Ruby.