



**«Московский государственный технический университет  
имени Н.Э. Баумана»**

**(МГТУ им. Н.Э. Баумана)**

---

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Компьютерные системы и сети(ИУ6)

## **О т ч е т**

**по лабораторной работе № \_\_\_\_\_**

**Дисциплина:** Языки интернет-программирования

**Название лабораторной работы:** \_\_\_\_\_

\_\_\_\_\_

Студент гр. ИУ6-32Б \_\_\_\_\_ Д.В.Лабзунова  
(Подпись, дата) (И.О. Фамилия)

Преподаватель \_\_\_\_\_  
(Подпись, дата) (И.О. Фамилия)

# 1. Задание

## Часть 1

Вычислить:  $y = -\frac{\cos^{1/2}(\frac{5 \cdot x}{1-6 \cdot x})}{x^2}$ .

## Часть 2

Дана последовательность строк. Строки содержат зашифрованную информацию и состоят из слов, разделенных пробелом. Пробел записан без шифра. Написать программу, обеспечивающую ввод строк и их расшифровку. Для расшифровки каждая из букв слова заменяется следующей за ней буквой алфавита. Буква «я» заменяется буквой «а». Вывести на печать зашифрованную и подвергнутую дешифровке последовательности строк.

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

# 2. Код программы

## 1 часть:

main.rb:

```
# frozen_string_literal: true

require_relative 'math_func'

# client class
class Main
  puts 'enter x'
  arg = begin
    Integer(gets.chomp)
  rescue StandardError
    nil
  end
  abort 'NaN' unless arg.is_a? Numeric
  arg = arg.to_i
  math = MathFunc.new
  res = math.func(arg)
  puts 'Result of -sqrt(cos(5*x/(1-6*x)))/(x*x): ', res
end
```

math\_func.rb:

```
# frozen_string_literal: true

# Math logic
# :reek:UtilityFunction
class MathFunc
  def func(arg)
    fun = Math.cos(5 * arg / (1 - 6 * arg))
    fun = -Math.sqrt(fun)
    fun /= (arg * arg)
    fun.round(6)
  end
end
```

```
end
end
```

test.rb:

```
require 'minitest/autorun'
require_relative 'math_func'

# testing math logic
class TestMath < Minitest::Test
  def setup; end

  def test_math
    assert_equal(-0.195969, MathFunc.func(2))
    assert_equal(-0.735053, MathFunc.func(1))
  end
end
```

## 2 часть:

client.rb:

```
# frozen_string_literal: true

require_relative 'decoder'

# client class
class Client
  sequence = gets('0').chomp('0')
  decoder = Decoder.new
  puts '---'
  puts 'initial sequence: ', sequence
  puts '---'
  puts 'final sequence: '
  strings = sequence.split(10.chr)
  (0..strings.length - 1).each do |iteration|
    puts decoder.decode(strings[iteration])
  end
end
```

decoder.rb:

```
# frozen_string_literal: true

# string decoding class
# :reek:UtilityFunction
# :reek:TooManyStatements
class Decoder
  def decode(str)
    decoded_string = []
    arr = str.split('//')
    (0..arr.length - 2).each do |iteration|
      current_word = arr[iteration]
      decoded_string[iteration] = case current_word
      when ' '
        ' '
      when 'я'
        'а'
      else
        (current_word.ord + 1).chr
      end
    end
  end
end
```

```
    decoded_string.join + ' '
  end
end
```

test.rb:

```
# frozen_string_literal: true

require 'minitest/autorun'

require_relative 'decoder'

# :reek:TooManyStatements
# :reek:UtilityFunction
# decode logic testing
class Test < Minitest::Test
  def setup; end

  def create_string
    rnd = Random.new
    initial_string = []
    final_string = []
    20.times do |iteration|
      symbol_code = rnd.rand(65...122)
      initial_string[iteration] = symbol_code.chr
      final_string[iteration] = case symbol_code[iteration]
                                when ' '
                                  ' '
                                when 'z'
                                  'a'
                                else
                                  (symbol_code + 1).chr
                                end
    end
    decoder = Decoder.new
    [decoder.decode(initial_string), final_string]
  end

  def test_decoding
    assert_equal(create_string[1], create_string[2])
  end
end
```

### 3. Результаты выполнения:

#### 1 часть:

Основная

программа:

```
C:\Ruby26-x64\bin\ruby.exe "C:/Users/1/Desktop/Laboratory works/LR5/5.1/main.rb"
enter x
2
Result of -sqrt(cos(5*x/(1-6*x)))/(x*x):
-0.195969

Process finished with exit code 0
|
```

Тесты:

```
Testing started at 23:27 ...
C:\Ruby26-x64\bin\ruby.exe "C:\Program Files\JetBrains\RubyMine 2020.2.2\plugins\ruby\rb\testing\runner\minitest_runner.rb"
Started

Process finished with exit code 0
```

## 2 часть:

Основная программа:

```
C:\Ruby26-x64\bin\ruby.exe "C:/Users/1/Desktop/Laboratory works/LR5/5.2/client.rb"
djfsfdjfs fs fsfsf zzzzzsdfjknsfd
dfdsgdsfdsdf
0
--|
initial sequence:
djfsfdjfs fs fsfsf zzzzzsdfjknsfd
dfdsgdsfdsdf
---
final sequence:
ekgtgekgt gt gtgtg aaaaategklotge
egethetgeteg

Process finished with exit code 0
```

Тесты:

```
✓ Tests passed: 1 of 1 test - 0 ms
Testing started at 23:30 ...
C:\Ruby26-x64\bin\ruby.exe "C:\Program Files\JetBrains\RubyMine 2020.2.2\plugins\ruby\rb\testing\runner\minitest_runner.rb"
Started

Process finished with exit code 0
```

## 4. Проверка rubocop и reek:

```
Inspecting 6 files
.....

6 files inspected, no offenses detected
```

Rubocop:

```
C:\Users\1>_
```

```
C:\Users\1>reek "C:\Users\1\Desktop\Laboratory works\LR5"
Inspecting 6 file(s):
.....
```

Reek: 0 total warnings