***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования***

|  |  |
| --- | --- |
|  | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Отчет**

**по лабораторной работе № \_\_4\_\_**

**Дисциплина: \_\_Языки Интернет-программирования\_\_\_\_\_**

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

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студент гр. \_ИУ6-33Б**\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Яныгин Д.С.\_\_\_**

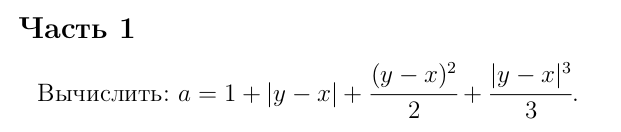
(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2020

**Задание 1:**

Часть 2

Дана последовательность строк. Каждая строка состоит из слов, раз-

деленных пробелами. Написать программу, обеспечивающую ввод строк

и их корректировку. Корректировка заключается в удалении или замене

слов. Если слово стоит на нечетном месте и начинается на букву «л» –

оно удаляется; если слово стоит на четном месте и начинается на бук-

ву «н», оно замещается на слово, введенное с клавиатуры. Вывести на

печать исходную и скорректированную последовательности строк.

Автоматический тест программы обязательно должен генерировать

случайные строки в соответствии с правилами, перечисленными в зада-

нии.

**Часть 1**

**LR5\_1\_func.rb**

def count(x, y)

a = 1 + (y-x).abs + ((y-x)\*\*2)/2.0 + (((y-x).abs)\*\*3)/3.0

end

**LR5\_1\_test.rb**

require 'minitest/autorun'

require\_relative 'LR5\_1\_func.rb'

class TestString < Minitest::Test

def test\_change\_example

assert\_equal 1.0, count(2, 2)

end

end

**LR5\_1\_user.rb**

load "LR5\_1\_func.rb"

p ('Введите x и y')

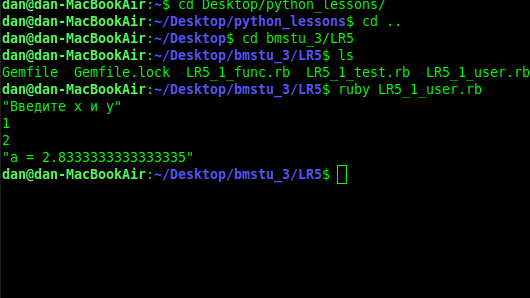
x = gets.to\_i

y = gets.to\_i

p ('a = ' + count(x, y).to\_s)

**Результат:**

Рис. 1



**Часть 2**

**LR5\_2\_func.rb**

def change(s, replace)

s.split

.each\_with\_index

.map do |x, idx|

if idx.even? && x.start\_with?('л')

nil

elsif idx.odd? && x.start\_with?('н') then

replace

else

x

end

end

.compact

.join(' ')

end

def output(s, r)

temp = change(s, r)

puts('\nOriginal string: ' + s)

puts('Changed string: ' + temp)

end

def input()

s = ""

ARGF.each\_line do |line|

s += line

end

s

end

**LR5\_2\_test.rb**

require 'minitest/autorun'

require\_relative 'LR5\_2\_func.rb'

class TestString < Minitest::Test

def test\_change\_example

assert\_equal 'кек ненене лолол', change('лолол ненен ненене лолол', 'кек')

end

end

**LR5\_2\_user.rb**

require\_relative "LR5\_2\_func.rb"

puts("Enter string for replace: ")

rep = gets.chomp

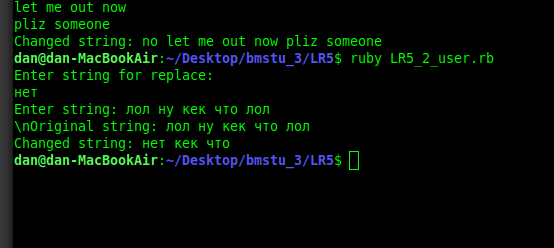
print("Enter string: ")

s = input()

output(s, rep)

**Результат:**

Рис. 2



**Вывод:** в ходе работы были реализованы простые программы на ruby, чтобы научиться пользоваться его самым базовым функционалом.