



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

О Т Ч Е Т

по лабораторной работе № 5


Дисциплина: Языки интернет-программирования

Вариант: 23

Студент

ИУ6-33Б

(Группа)


(Подпись, дата)

Д. И. Мироненко

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

В. Д. Шульман

(И.О. Фамилия)

Москва, 2023

Часть 1

Задание

Вычислить: $y = \frac{\sin(a)-b}{|b|+\cos(b^2)}$.

Код программы

Файл “main.rb”

```
# frozen_string_literal: true

require_relative 'utils'

print "Enter 'a': "

a = gets.chomp

print "Enter 'b': "

b = gets.chomp

puts "Result: #{f(a, b)}"
```

Файл “utils.rb”

```
# frozen_string_literal: true

def f(param_a, param_b)

  a = param_a.to_f

  b = param_b.to_f

  (Math.sin(a) - b) / (b.abs + Math.cos(b**2))

end
```

Файл “test_utils.rb”

```
# frozen_string_literal: true

require 'minitest/autorun'

require_relative 'utils'

# Test `utils.rb`

class TestUtils < Minitest::Test

  def test_f

    assert_in_delta(-1.0, f(5, 9), 0.1)

  end

end
```

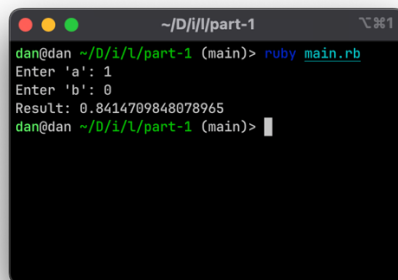
```
    assert_in_delta(-0.9, f(8, 3), 0.1)

    assert_in_delta(0.8, f(1, 0), 0.1)

end

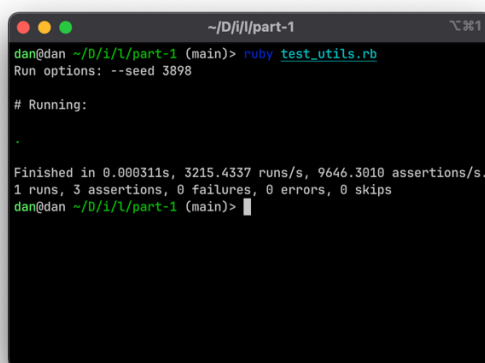
end
```

Работа программы



```
dan@dan ~/D/i/l/part-1 (main)> ruby main.rb
Enter 'a': 1
Enter 'b': 0
Result: 0.8414709848078965
dan@dan ~/D/i/l/part-1 (main)>
```

Рисунок 1 – работа программы



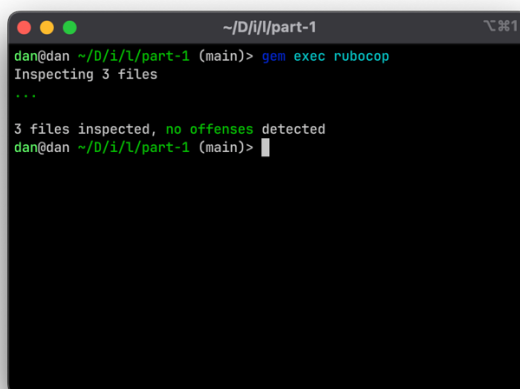
```
dan@dan ~/D/i/l/part-1 (main)> ruby test_utils.rb
Run options: --seed 3898

# Running:

.

Finished in 0.000311s, 3215.4337 runs/s, 9646.3010 assertions/s.
1 runs, 3 assertions, 0 failures, 0 errors, 0 skips
dan@dan ~/D/i/l/part-1 (main)>
```

Рисунок 2 – выполнение unit-тестов



```
dan@dan ~/D/i/l/part-1 (main)> gem exec rubocop
Inspecting 3 files
...

3 files inspected, no offenses detected
dan@dan ~/D/i/l/part-1 (main)>
```

Рисунок 3 – проверка кода rubocop'ом

Часть 2

Задание

Дана строка, состоящая из дат в ISO-формате, разделённых одним пробелом. Найти самую близкую дату к сегодняшнему дню. (2022-01-11 2022-01-24 . . .)

Код программы

Файл “main.rb”

```
# frozen_string_literal: true

require 'date'
require_relative 'utils'

print 'Enter dates in ISO format: '

dates = gets.chomp.split

today = Date.today

puts "Closest date: #{get_closest_date(dates, today)}"
```

Файл “utils.rb”

```
# frozen_string_literal: true

require 'date'

def get_closest_date(dates, target_date)
  dates.min_by do |date|
    (Date.parse(date) - target_date).abs
  end
end
```

Файл “test_utils.rb”

```
# frozen_string_literal: true

require 'minitest/autorun'
require 'date'
require_relative 'utils'

# Test 'utils.rb'
```

```
class TestUtils < Minitest::Test

  def test_get_closest_date1

    dates = ['2022-5-14', '2023-2-7', '2023-6-29', '2022-3-28']

    target_date = Date.today

    assert_equal '2023-6-29', get_closest_date(dates, target_date)

  end

  def test_get_closest_date2

    dates = ['2022-5-14', '2023-11-4', '2023-6-29']

    target_date = Date.today

    assert_equal '2023-11-4', get_closest_date(dates, target_date)

  end

  def test_get_closest_date3

    dates = ['2023-10-3', '1998-3-5', '1965-2-19']

    target_date = Date.today

    assert_equal '2023-10-3', get_closest_date(dates, target_date)

  end

end
```

Работа программы

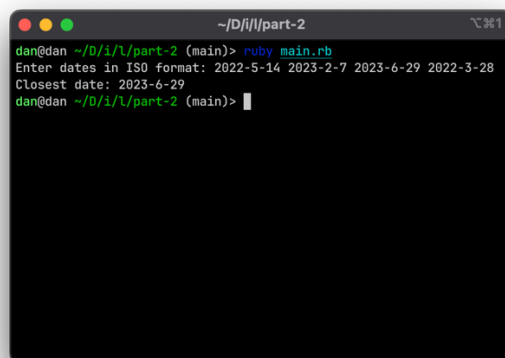
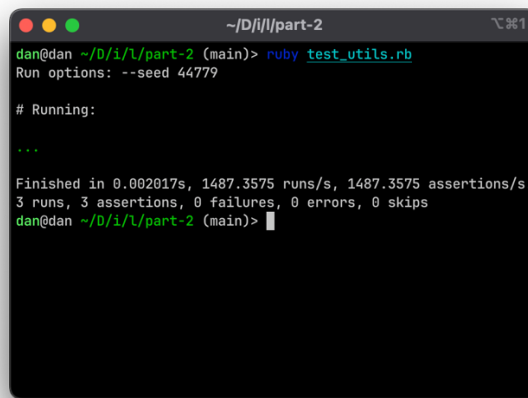


Рисунок 4 — работа программы



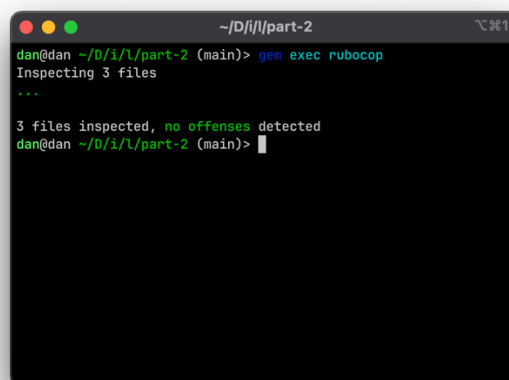
```
dan@dan ~/D/i/l/part-2 (main)> ruby test_utils.rb
Run options: --seed 44779

# Running:

...

Finished in 0.002017s, 1487.3575 runs/s, 1487.3575 assertions/s.
3 runs, 3 assertions, 0 failures, 0 errors, 0 skips
dan@dan ~/D/i/l/part-2 (main)>
```

Рисунок 5 — выполнение unit-тестов



```
dan@dan ~/D/i/l/part-2 (main)> gem exec rubocop
Inspecting 3 files
...

3 files inspected, no offenses detected
dan@dan ~/D/i/l/part-2 (main)>
```

Рисунок 6 — проверка кода rubocop'ом

Часть 3

Задание

Дана последовательность строк. Каждая строка состоит из слов, разделенных пробелами. Написать программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в следующем. Если слово содержит символы, отличные от букв латинского алфавита и цифр, то удалить его. Если слово состоит из букв латинского алфавита и цифр и начинается с цифры, заменить эту цифру символом «_» подчеркивание. Вести подсчет количества корректировок. Вывести на печать исходные и скорректированные последовательности строк. Автоматический тест программы обязательно должен генерировать случайные строки в соответствии с правилами, перечисленными в задании.

Код программы

Файл “main.rb”

```
# frozen_string_literal: true

require_relative 'utils'

SEPARATOR = '-' * 30

print 'Strings count: '

strings_count = gets.to_i

strings = input(strings_count)

puts SEPARATOR

puts 'Source text:'

show strings

corrections_count = fix_text(strings)

puts SEPARATOR

puts "Fixed text (corrections: #{corrections_count})."

show strings
```

Файл “utils.rb”

```
# frozen_string_literal: true

def alnum?(str)

  /^[[:alnum:]]+$/ .match?(str)

end

def numeric?(str)

  /^[[:digit:]]+$/ .match?(str)

end

def fix_text(strings)

  corrections = 0

  strings.size.times do |i|

    new_string = []
```

```

strings[i].size.times do |j|

  if alnum? strings[i][j]

    if numeric? strings[i][j][0]

      new_string.push("_#{strings[i][j][1..]}")

      corrections += 1

    else

      new_string.push(strings[i][j])

    end

  else

    corrections += 1

  end

end

strings[i] = new_string

end

corrections

end

def input(count)

  raise StandardError, "'count' cannot be a negative number!" if count.negative?

  strings = []

  count.times do |i|

    print "#{i + 1} "

    strings.push(gets.split)

  end

  strings

end

def show(strings)

  strings.size.times do |i|

    puts "#{i + 1} #{strings[i].join(' ')}"

  end

end

```

Файл “test_utils.rb”

```
# frozen_string_literal: true
```



```

require 'minitest/autorun'

require_relative 'utils'

# Test 'fix_text' function

class TestFixText < Minitest::Test

  def test_some_text

    strings = [

      %w[j4hk jh$4f4 5owbe3],

      %w[3qn43 ie&1df]

    ]

    assert_equal 4, fix_text(strings)

    assert_equal [%w[j4hk _owbe3], ['_qn43']], strings

  end

  def test_empty_text

    strings = [[]]

    assert_equal 0, fix_text(strings)

    assert_equal [[]], strings

  end

  def generate_random_word

    chars = ('a'..'z').to_a + ('A'..'Z').to_a + (0..9).to_a + ['_', '!', '@', '#', '$', '%', '^', '&', '*']

    len = rand(15)

    word = ""

    len.times do

      word += chars.sample.to_s

    end

    word

  end

  def generate_random_strings

    strings = []

    rand(15).times do

      new_strings = []

      rand(15).times do

```

```
new_strings.push(generate_random_word)

end

strings.push(new_strings)

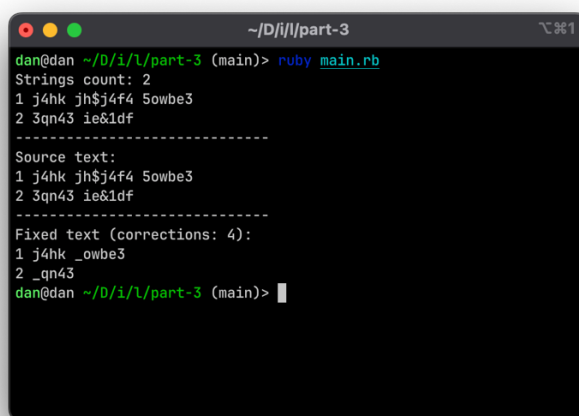
end

strings

end

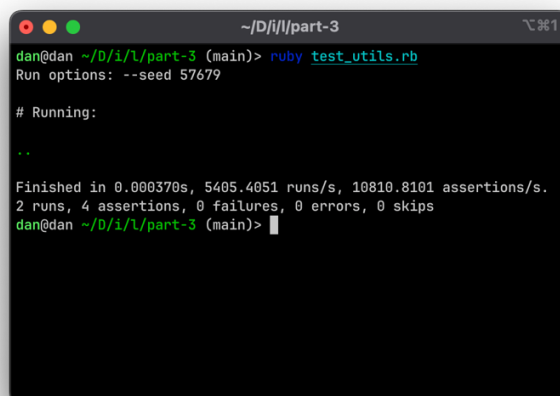
end
```

Работа программы



```
dan@dan ~/D/i/l/part-3 (main)> ruby main.rb
Strings count: 2
1 j4hk jh$j4f4 5owbe3
2 3qn43 ie&ldf
-----
Source text:
1 j4hk jh$j4f4 5owbe3
2 3qn43 ie&ldf
-----
Fixed text (corrections: 4):
1 j4hk _owbe3
2 _qn43
dan@dan ~/D/i/l/part-3 (main)>
```

Рисунок 7 – работа программы



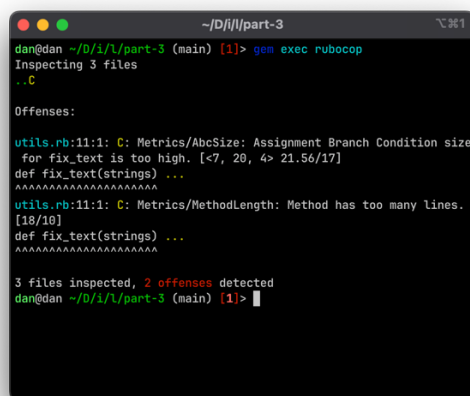
```
dan@dan ~/D/i/l/part-3 (main)> ruby test_utils.rb
Run options: --seed 57679

# Running:

..

Finished in 0.000370s, 5405.4051 runs/s, 10810.8101 assertions/s.
2 runs, 4 assertions, 0 failures, 0 errors, 0 skips
dan@dan ~/D/i/l/part-3 (main)>
```

Рисунок 8 – выполнение unit-тестов

A terminal window with a dark background and light-colored text. The window title is "~/D/i/l/part-3". The prompt is "dan@dan ~/D/i/l/part-3 (main) [1]". The user has entered "gem exec rubocop". The output shows "Inspecting 3 files", followed by a separator line "C". Under the heading "Offenses:", two errors are listed. The first error is "utils.rb:11:1: C: Metrics/AbcSize: Assignment Branch Condition size for fix_text is too high. [<7, 20, 4> 21.56/17]" and the second is "utils.rb:11:1: C: Metrics/MethodLength: Method has too many lines. [18/10]". Both errors are followed by a separator line "C". At the bottom, it says "3 files inspected, 2 offenses detected" and the prompt "dan@dan ~/D/i/l/part-3 (main) [1]" is shown again with a cursor.

```
~/D/i/l/part-3
dan@dan ~/D/i/l/part-3 (main) [1]> gem exec rubocop
Inspecting 3 files
..C

Offenses:

utils.rb:11:1: C: Metrics/AbcSize: Assignment Branch Condition size
for fix_text is too high. [<7, 20, 4> 21.56/17]
def fix_text(strings) ...
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
utils.rb:11:1: C: Metrics/MethodLength: Method has too many lines.
[18/10]
def fix_text(strings) ...
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

3 files inspected, 2 offenses detected
dan@dan ~/D/i/l/part-3 (main) [1]>
```

Рисунок 9 – проверка кода rubocop’ом

Вывод

Изучил основы языка программирования ruby, написав несколько программ; познакомился с unit-тестированием; проверил код на наличие ошибок с помощью линтера rubocop.