|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 **09.03.03 Прикладная информатика**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | **7** |

**Название:**

Работа с файлами. ООП

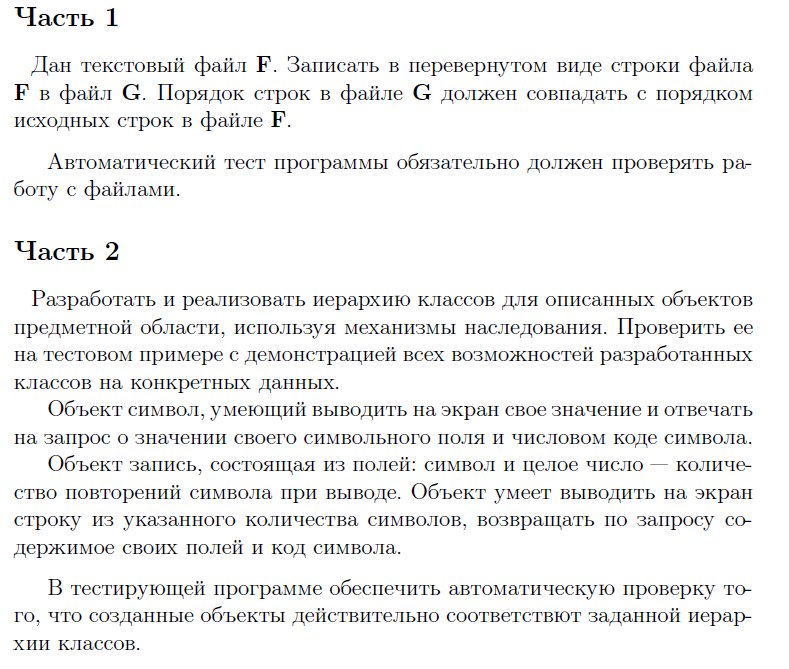
**Дисциплина:** Языки Интернет-программирования



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-35Б |  | 27.10.2023 | В. И. Мамыкин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Е.Ю. Гаврилова |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Задание:**

****

**Цель:** реализовать консольные приложения на Ruby и написать для них тесты. Научиться работать с файлами и классами.

Часть 1:

**1.rb**

# frozen\_string\_literal: true

require\_relative 'functions'

fill\_the\_files

**functions.rb**

require 'securerandom'

def generate\_random\_string(length)

SecureRandom.alphanumeric(length)

end

def fill\_the\_files

f\_file = File.open('F.txt', 'w')

rand(1..10).times do

random\_string = generate\_random\_string(rand(5..10))

f\_file.puts random\_string

end

f\_file.close

f\_file = File.open('F.txt', 'r')

g\_file = File.open('G.txt', 'w')

f\_file.each\_line do |line|

reversed\_line = line.chomp.reverse

g\_file.puts reversed\_line

end

f\_file.close

g\_file.close

true

end

def proverka

f\_file = File.open('F.txt', 'r')

g\_file = File.open('G.txt', 'r')

f\_file.each\_line do |line|

reserved\_line = line.chomp.reverse

line\_from\_g\_file = g\_file.gets.chomp

return false if reserved\_line != line\_from\_g\_file

end

true

end

**tests.rb**

# frozen\_string\_literal: true

require 'minitest/autorun'

require\_relative '1'

class TestFunc < Minitest::Test

def test\_func1

assert\_equal(true, File.exist?('F.txt'))

assert\_equal(proverka, fill\_the\_files)

end

def test\_func2

assert\_equal(true, File.exist?('F.txt'))

assert\_equal(proverka, fill\_the\_files)

end

def test\_func3

assert\_equal(true, File.exist?('F.txt'))

assert\_equal(proverka, fill\_the\_files)

end

end

Часть 2:

**1.rb**

# frozen\_string\_literal: true

require\_relative 'functions'

symbol\_a = Symbols.new('A')

record\_b = Record.new('B', 4)

# Вызов методов

symbol\_a.display\_value

puts symbol\_a.symbol\_field

puts symbol\_a.symbol\_code

record\_b.display\_value

puts record\_b.symbol\_field

puts record\_b.symbol\_code

puts record\_b.repeat\_count

record\_b.display\_string

**functions.rb**

class Symbols

def initialize(value)

@value = value

end

def display\_value

puts @value

end

def symbol\_field

@value

end

def symbol\_code

@value.ord

end

end

class Record < Symbols

def initialize(value, repeat\_count)

super(value)

@repeat\_count = repeat\_count

end

def display\_string

puts @value \* @repeat\_count

end

attr\_reader :repeat\_count

end

def proverka

symbol\_a = Symbols.new('A')

record\_b = Record.new('B', 4)

symbol\_a.is\_a?(Symbols) && record\_b.is\_a?(Symbols) && record\_b.is\_a?(Record)

end

**tests.rb**

# frozen\_string\_literal: true

require 'minitest/autorun'

require\_relative '1'

class TestFunc < Minitest::Test

def test\_func1

assert\_equal(proverka, true)

end

end

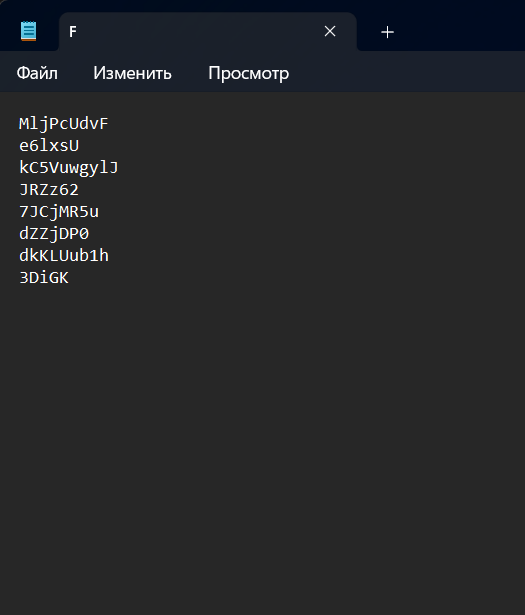


Рисунок 1 – Файл F

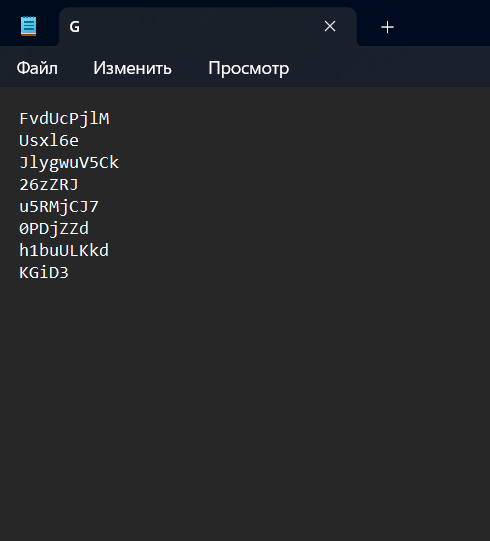


Рисунок 2 – Файл G

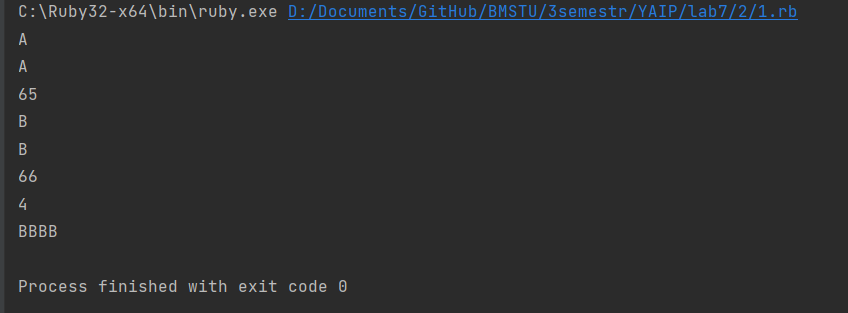


Рисунок 3 – Результат выполнения 2 программы

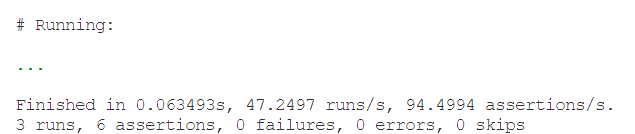


Рисунок 4 – Результат выполнения тестов 1 части

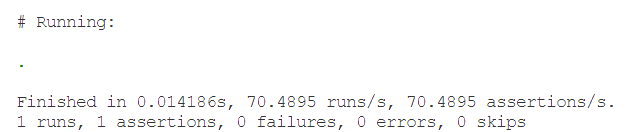


Рисунок 5 – Результат выполнения тестов 2 части

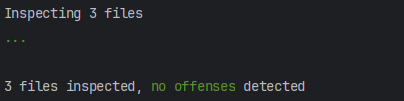


Рисунок 6 – Результат работы robocop (*rubocop --config rubocop.yaml*) (аналогично для всех частей задания)



Рисунок 7 – Результат работы reek (аналогично для всех частей задания)

**Вывод:** были сделаны консольные приложения на Ruby и написаны для них тесты. Было изучено взаимодействие с файлами и классами.