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

|  |  |
| --- | --- |
| **Описание: Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Компьютерные системы и сети

**Отчет**

**по лабораторной работе № 7**

**Дисциплина:**Языки интеренет программирования.

Вариант 10.

Студент гр. ИУ6-33\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Гуськов А. Д..

(Подпись, дата) (И.О. Фамилия)

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

(Подпись, дата) (И.О. Фамилия)

Москва, 2017

Часть 1:

Задание: Сформировать программным путем символьный файл F, содержащий слова. Определить количество слов в файле F, имеющих длину 5 символов.

Код программы:

new2.rb

# coding: utf-8

require 'E:\lio\7\1\dop6.rb'

st = ''

puts 'введите строчку'

st = gets

f = File.new("F.txt", "w")

f.write(st)

f.close

puts 'количество слов с 5 буквами: ' + poi.to\_s

dop6.rb

# coding: utf-8

def poi

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

t = f.gets

f.close

m = t.split(' ')

t = m.select { |i| i.length == 5 }.size

return t

end

test.rb

require 'E:\lio\7\1\dop6.rb'

require 'minitest/autorun'

class Testlab < Minitest::Test

def setup

str = 'hrhaj eha 6'

@f = File.new

@f.write(str)

@f.close

end

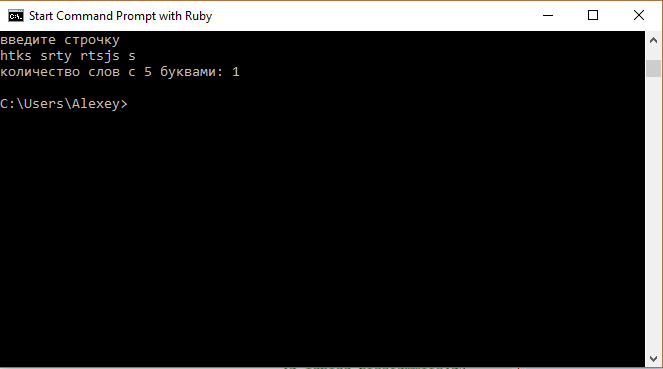
def test

assert\_equal(poi,1)

end

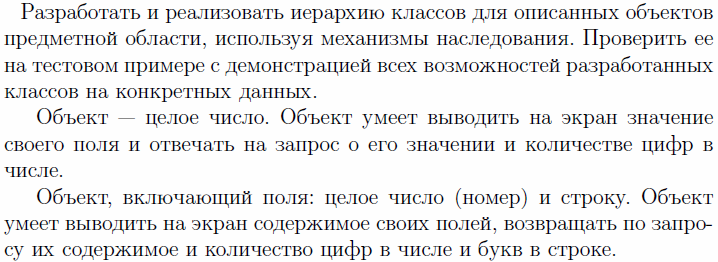
end

Тест:



Часть 2:

Задание:



Код программы:

di.rb

# coding: utf-8

require 'E:\lio\7\2\pik.rb'

s = 0

while s == 0

puts 'введите число'

s = gets.to\_i

end

puts 'введите строку'

str = gets.to\_s

chis = Chislo\_p.new(s)

stro = Strok\_p.new(s,str)

puts

puts 'элементы объекта число:'

chis.output

puts

puts 'элементы объекта строка:'

stro.output

pik.rb

# coding: utf-8

class Chislo\_p

attr\_accessor :s

def initialize(p)

@p = p

end

def col

@o = @p.to\_s.length

end

def output

puts 'число: '+ @p.to\_s

puts 'количество цифр в чисе: ' + col.to\_s

return @o

end

end

class Strok\_p < Chislo\_p

attr\_accessor :str

def initialize(p,st)

super p

@st = st

end

def output

super

puts 'строка: ' + @st

l = @st.length

puts 'число букв в строке:' + l.to\_s

return l.to\_i

end

end

test.rb

require 'E:\lio\7\2\pik.rb'

require 'minitest/autorun'

class TestProg < Minitest::Test # :nodoc:

def setup

@class1 = Chislo\_p.new(364)

@class2 = Strok\_p.new(364, 'rjsr')

end

def test\_1

assert\_equal(@class1.output, 3)

end

def test\_2

assert\_equal(@class2.output, 4)

end

end

Тест:

