Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

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

(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Компьютерные системы и сети

О т ч е т

по Лабораторной работе № 6

Вариант № 16

Дисциплина: Языки интернет программирования

Название домашней работы: Анализ линейной электрической цепи постоянного тока

Студент гр. ИУ6-34 _____ Лавренов И.С.
(Подпись, дата) (И.О. Фамилия)

Преподаватель _____ _____
(Подпись, дата) (И.О. Фамилия)

Москва, 2017

Задание

Часть 1

Организовать итерацинный цикл. Вычислить длину окружности с точностью 0.001, 0.0001 как предел последовательности периметров вписанных правильных многоугольников с удваивающимся числом сторон (начинасть с 6). Использовать формулу удвоения стороны $a_{2n}=sqrt(2R^2 - 2R*sqrt(R^2 - a_n^2 /4))$

Часть 2

Решить предыдущее задание с помощью enumerator.

Часть 3

Найти минимальный положительный корень уравнения с точностью до 0.0001

Уравнения:

x^2+sin(x/2)=0

Arctg(x)+x=1

Часть 1

Код interface.rb

```
require_relative './main.rb'

print 'Input precision 1e-3 or 1e-4: '
precision_s = gets
precision = precision_s.to_f
print 'Input radius: '
radius_s = gets
radius = radius_s.to_f
if [1e-3, 1e-4].include? precision
 print calc(precision,radius)
else
 print 'Choose one of given numbers'
end
```

Код main.rb

```
def side(previous_side,r)
 Math.sqrt(2*r**2 - 2*r*Math.sqrt(r**2 - previous_side**2/4))
end

def calc(precision,rad)
 current_side = rad
 sides_num = 6
 while ((sides_num*2)*side(current_side,rad) - sides_num*current_side) > precision
/ 2
  current_side = side(current_side,rad)
  sides_num *= 2
 end
 current_side*sides_num
end
```

Код test.rb

```ruby
require 'minitest/autorun'
require './main.rb'

class Test < MiniTest::Test

 def setup
 end

 def teardown
 end

 def test_ok
  (3..100).each do |r|
   assert_in_delta 2*r*Math::PI, calc(1e-3, r.to_f), 1e-3
   assert_in_delta 2*r*Math::PI, calc(1e-4, r.to_f), 1e-4
  end
 end

end
```

Скриншоты

```
Input precision 1e-3 or 1e-4: 1e-3
Input radius: 25
157.07919460741846
Process finished with exit code 0
```

Рис.1. Выполнение программы

```
Started

##teamcity[enteredTheMatrix timestamp = '2017-12-24T14:33:59.039+0300']

##teamcity[testCount count = '0' timestamp = '2017-12-24T14:33:59.039+0300']
Run options: --seed 36334

# Running:

##teamcity[testStarted name = 'test_ok' captureStandardOutput = 'true' locationHint = 'ruby_minitest_qn://Test.test_ok' timestamp = '2017-12-24T14:33:59.045+0300']

##teamcity[testFinished name = 'test_ok' duration = '4' timestamp = '2017-12-24T14:33:59.045+0300']
.Finished in 0.00600s
1 tests, 196 assertions, 0 failures, 0 errors, 0 skips


Finished in 0.005737s, 174.3110 runs/s, 34164.9596 assertions/s.
1 runs, 196 assertions, 0 failures, 0 errors, 0 skips

Process finished with exit code 0
```

Рис.2. Выполнение теста

Часть 2

Код interface.rb

```ruby
require_relative './main.rb'

print 'Input precision 1e-3 or 1e-4: '
precision_s = gets
precision = precision_s.to_f
print 'Input radius: '
radius_s = gets
radius = radius_s.to_f
if [1e-3, 1e-4].include? precision
 print calc(precision,radius)
else
 print 'Choose one of given numbers'
end
```

Код main.rb

```ruby
def side(previous_side,r)
 Math.sqrt(2*r**2 - 2*r*Math.sqrt(r**2 - previous_side**2/4))
end

def calc(precision,rad)
  sides_num = 6.0
  current_side = rad
 list = Enumerator.new do |calculating|
  loop do
   calculating.yield current_side
   current_side = side(current_side,rad)
   sides_num *= 2
  end
 end
 list.take_while {|current_side| ((sides_num*2)*side(current_side,rad) -
sides_num*current_side) > precision/2 }
 current_side*sides_num
end
```

Код test.rb

```ruby
require 'minitest/autorun'
require './main.rb'

class Test < MiniTest::Test

 def setup
 end

 def teardown
 end

 def test_ok
  (3..100).each do |r|
   assert_in_delta 2*r*Math::PI, calc(1e-3, r.to_f), 1e-3
   assert_in_delta 2*r*Math::PI, calc(1e-4, r.to_f), 1e-4
  end
 end
end
```

Скриншоты

```
Input precision 1e-3 or 1e-4: 1e-3
Input radius: 4
25.132460863292977
Process finished with exit code 0
```

Рис.3. Выполнение программы

```
Started

##teamcity[enteredTheMatrix timestamp = '2017-12-24T14:31:55.642+0300']

##teamcity[testCount count = '0' timestamp = '2017-12-24T14:31:55.642+0300']
Run options: --seed 47063

# Running:

##teamcity[testStarted name = 'test_ok' captureStandardOutput = 'true' locationHint = 'ruby_minitest_qn://Test.test_ok' timestamp = '2017-12-24T14:31:55.649+0300']

##teamcity[testFinished name = 'test_ok' duration = '6' timestamp = '2017-12-24T14:31:55.650+0300']
.Finished in 0.00700s
1 tests, 196 assertions, 0 failures, 0 errors, 0 skips


Finished in 0.007429s, 134.6133 runs/s, 26384.2120 assertions/s.
1 runs, 196 assertions, 0 failures, 0 errors, 0 skips

Process finished with exit code 0
```

Рис.4. Выполнение теста

Часть 3

Код interface.rb

```ruby
require_relative './main.rb'

puts "Lambda: "
p root(0, 300, 0, ->(x) {x**2 + Math.sin(x/2)})
p root(0, 300, 1, ->(x) {Math.atan(x) + x})
puts "Block: "
p root(0, 300, 0) {|x| x**2 + Math.sin(x/2)}
p root(0, 300, 1) {|x| Math.atan(x) + x}
```

Код main.rb

```ruby
def root(a, b, answer=0, equation = nil)
 while (b-a > 0.0001)
  c = (a+b)/2.0
  if block_given?
   if (yield(b)*yield(c) < answer )
    a = c
   else
    b = c
   end
  else
   if (equation.call(b) * equation.call(c) < answer)
    a = c
   else
    b = c
   end
  end
 end
 (a+b)/2
end
```

Код test.rb

```ruby
require 'minitest/autorun'
require_relative './main.rb'

class Test < MiniTest::Test

 def setup
 end

 def teardown
 end

 def test_ok
  assert_in_delta 3.576e-05, root(0, 300, 0, ->(x) {x**2 + Math.sin(x/2)}), 1e-3
  assert_in_delta 0.513, root(0, 300, 1) {|x| Math.atan(x) + x}, 1e-3
 end

end
```

Скриншоты

```
Lambda:
3.5762786865234375e-05
0.5126595497131348
Block:
3.5762786865234375e-05
0.5126595497131348
```

Рис.5. Выполнение программы

```
Started

##teamcity[enteredTheMatrix timestamp = '2017-12-24T14:29:23.589+0300']

##teamcity[testCount count = '0' timestamp = '2017-12-24T14:29:23.589+0300']
Run options: --seed 51299

# Running:

##teamcity[testStarted name = 'test_ok' captureStandardOutput = 'true' locationHint = 'ruby_minitest_qn://Test.test_ok' timestamp = '2017-12-24T14:29:23.590+0300']

##teamcity[testFinished name = 'test_ok' duration = '0' timestamp = '2017-12-24T14:29:23.590+0300']
.Finished in 0.00100s
1 tests, 2 assertions, 0 failures, 0 errors, 0 skips


Finished in 0.001126s, 887.7975 runs/s, 1775.5950 assertions/s.
1 runs, 2 assertions, 0 failures, 0 errors, 0 skips

Process finished with exit code 0
```

Рис.6. Выполнение теста

Вывод

Было создано 3 консольных ruby приложений по выданному условию. Все программы имеют тесты. Приложения протестированы и работают верно.