*Федеральное государственное бюджетное образовательное учреждение высшего образования*

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

***(национальный исследовательский университет)***

### ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

**О т ч е т**

# по лабораторной работе №6

**вариант 2.17**

**Дисциплина:**

### Языки интернет-программирования

**Название лабораторной работы:**

[Особенности использования языка Ruby](http://e-learning.bmstu.ru/moodle/mod/resource/view.php?id=1005)

Студент гр. ИУ6-32 18.02.2018 **Петров Олег Алексеевич**

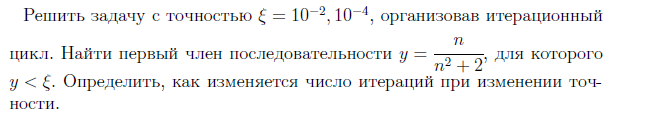
(Подпись, дата) (И.О. Фамилия)

Преподаватель

(Подпись, дата) (И.О. Фамилия)

Москва, 2018

# Задание 1:



**Код front.rb**

*require\_relative 'back'*

*[10\*\*-2, 10\*\*-4].each do |ce|*

*puts f(ce)*

*end*

**Код test.rb**

*require 'minitest/autorun'*

*require\_relative 'back'*

*# Minitest*

*class Test < Minitest::Test*

*def test\_delta*

*assert\_in\_delta(f(10\*\*-2), 100, 10\*\*-2)*

*assert\_in\_delta(f(10\*\*-4), 10000, 10\*\*-4)*

*end*

*end*

**Код back.rb**

*require 'mathn'*

*def f(ce)*

*i = 1*

*cr = i/(i\*i+2)*

*until cr <= ce do*

*i=i+1*

*cr = i/(i\*i+2)*

*end*

*i*

*end*

**Результаты работы программы**

****

Рис.1 результат запуска test.rb

# Задание 2:

Решить предыдущее задание с помощью Enumerator

**Код front.rb**

*require\_relative* **'back'**[10\*\*-2, 10\*\*-4].each **do** |*ce*|  
 puts f(*ce*).to\_f  
**end**

**Код test.rb**

*require* **'minitest/autorun'***require\_relative* **'back'***# Minitest***class *Test*** < ***Minitest***::***Test* def** *test\_delta* assert\_in\_delta(f(10\*\*-2), 0.009998000399920015, 10\*\*-2)  
 assert\_in\_delta(f(10\*\*-4), 0.000099999980000000, 10\*\*-4)  
 **end  
end**

**Код back.rbw**

*require* **'mathn'  
  
def** *f*(*ce*)  
 *e* = ***Enumerator***.*new* **do** |*y*|  
 *i* = 1  
 *cr* = *i*/(*i*\**i*+2)  
 loop **do** *i*=*i*+1  
 *cr* = *i*/(*i*\**i*+2)  
 *y* << *cr* **end  
 end** *e*.lazy.find{ |*cr*| *cr* <= *ce* }  
**end**

**Результаты работы программы**

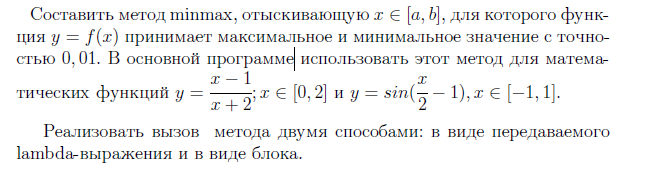
****

Рис.2 результат запуска test.rb



Рис.3 результат запуска front.rb

# Задание 3:



**Код front.rb**

*require\_relative* **'back'**[10\*\*-2, 10\*\*-4].each **do** |*ce*|  
 puts f(*ce*).to\_f  
**end**

**Код test.rb**

*require* **'minitest/autorun'***require\_relative* **'back'***# Minitest***class *Test*** < ***Minitest***::***Test* def** *test\_delta* assert\_in\_delta(f(10\*\*-2), 0.009998000399920015, 10\*\*-2)  
 assert\_in\_delta(f(10\*\*-4), 0.000099999980000000, 10\*\*-4)  
 **end  
end**

**Код back.rbw**

*require* **'mathn'  
  
def** *f*(*ce*)  
 *e* = ***Enumerator***.*new* **do** |*y*|  
 *i* = 1  
 *cr* = *i*/(*i*\**i*+2)  
 loop **do** *i*=*i*+1  
 *cr* = *i*/(*i*\**i*+2)  
 *y* << *cr* **end  
 end** *e*.lazy.find{ |*cr*| *cr* <= *ce* }  
**end**

**Результаты работы программы**

****

Рис.4 результат запуска test.rb

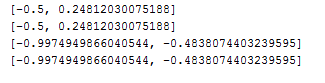


Рис.5 результат запуска front.rb

# Вывод:

1. Создан код приложения Ruby, реализованный в виде трех отдельных файлов.
2. Проведено тестирование.
3. Тестирование показало корректность работы.