Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

|  |  |
| --- | --- |
| **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**

**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 числовой файл F. И записать в файл H числа файла F которые делятся на 3, но не делятся на 7.

Часть 2

Разработать и реализовать иерархию классов для описанных объектов.

Объект ­– целое число. Умеет выводить значение своего поля и отвечать на запрос о его значении и количестве цифр в числе.

Объект, имеющий поля: целое и вещественное числа. Умеет выводить значение своих поле и отвечать на запрос о его значении и количестве цифр в целом числе.

# Часть 1

Код interface.rb

*require\_relative* **'./main.rb'**create\_files  
puts **'Files F.txt and H.txt has been created'**

Код main.rb

**def** *create\_files* ***File***.open(**'F.txt'**, **'w'**) **do** |*f*|  
 *str* = (0...rand(100)).map{(**'a'**..**'z'**).to\_a[rand(26)]}.join  
 *f*.write(*str*)  
 **end  
   
 *File***.open(**'F.txt'**) **do** |*f*|  
 *h* = ***File***.open(**'H.txt'**, **'w'**)  
 *str* = *f*.readline  
 *arr* = []  
 *str*.each\_char {|*c*| *arr* << *c*}  
 *arr*.each\_with\_index {|*val*, *index*| *h*.write(*val*) **if** *arr*[*index*-1] == **'a'**}  
 **end   
end**

Код test.rb

*require* **'minitest/autorun'***require* **'./main.rb'  
  
class *Test*** < ***MiniTest***::***Test* def** *test\_file  
 file\_mock* = ***MiniTest***::***Mock***.*new  
 file\_mock*.expect  
 **end  
end**

Скриншоты

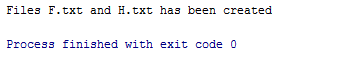


Рис.1. Выполнение программы

Содержимое файла F: nwupysslwxztcmvfshjstesonpqhmoyafd

qevtaynjjxkxcqnjkqyplbahobsalukzll

Содержимое файла H: fyhl

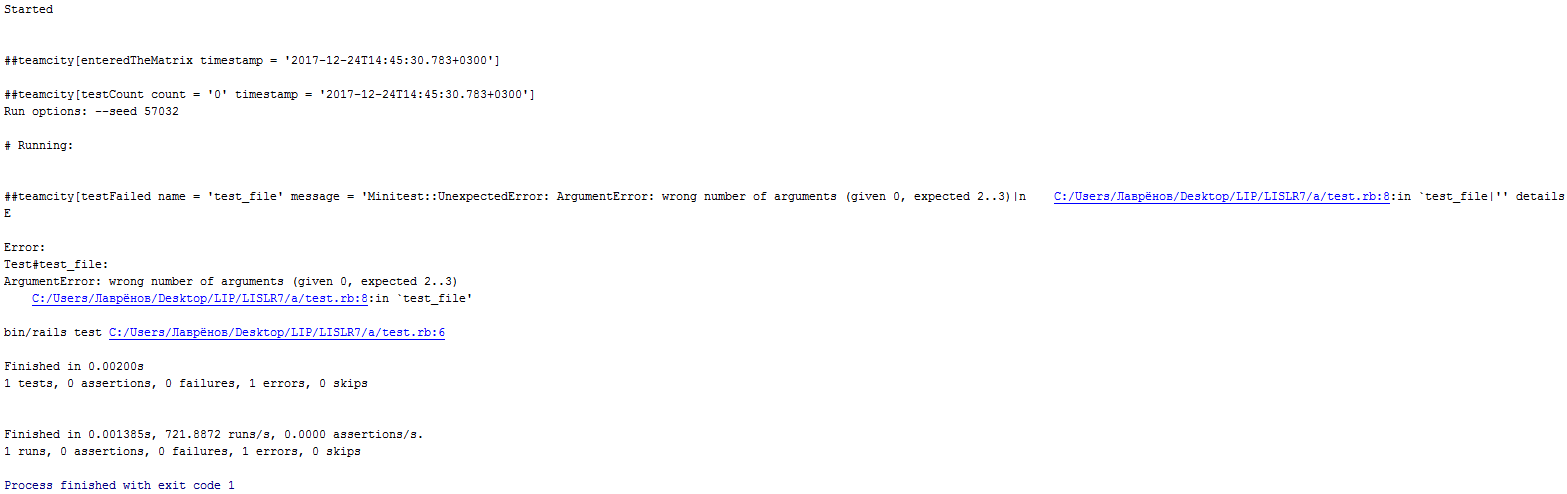


Рис.2. Выполнение теста

# Часть 2

Код interface.rb

*require\_relative* **'./main.rb'**puts **"Checking IntNum class"**print(**'Enter an integer number: '**)  
*inum* = gets  
*intnum* = ***IntNum***.*new*(*inum*)  
*intnum*.show\_num  
*intnum*.length\_of\_num  
  
puts **"Checking IntAndFloat class"**print(**'Enter an integer number: '**)  
*inum* = gets  
print(**'Enter a float number: '**)  
*fnum* = gets  
*intandfloat* = ***IntAndFloat***.*new*(*inum*,*fnum*)  
*intandfloat*.show\_num  
*intandfloat*.length\_of\_num

Код main.rb

**class *IntNum*** *attr\_accessor* **:inum  
   
 def** *initialize*(*input\_inum*)  
 **@inum** = *input\_inum*.to\_i  
 **end  
   
 def** *show\_num* puts **"Int number:** #{**@inum**}**"  
 end  
   
 def** *length\_of\_num* puts **"Number of numbers:** #{**@inum**.to\_s.chars.map(&**:to\_i**).length}**"  
 end  
end  
  
class *IntAndFloat*** < ***IntNum*** *attr\_accessor* **:fnum  
   
 def** *initialize*(*input\_inum*, *input\_fnum*)  
 **super**(*input\_inum*)  
 **@fnum** = *input\_fnum*.to\_f  
 **end  
   
 def** *show\_num* **super** puts **"Float number:** #{**@fnum**}**"  
 end  
end**

Код test.rb

*require* **'minitest/autorun'***require* **'./main.rb'  
  
class *Test*** < ***MiniTest***::***Test* def** *test\_relatives* assert\_kind\_of ***IntNum***, ***IntAndFloat***.*new*(45676, 33.35)  
 **end  
   
 def** *test\_length* assert\_equal 5, ***IntAndFloat***.*new*(45676, 33.35).length\_of\_num  
 **end  
   
end**

Скриншоты

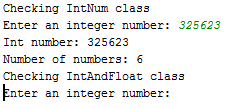


Рис.3. Выполнение программы

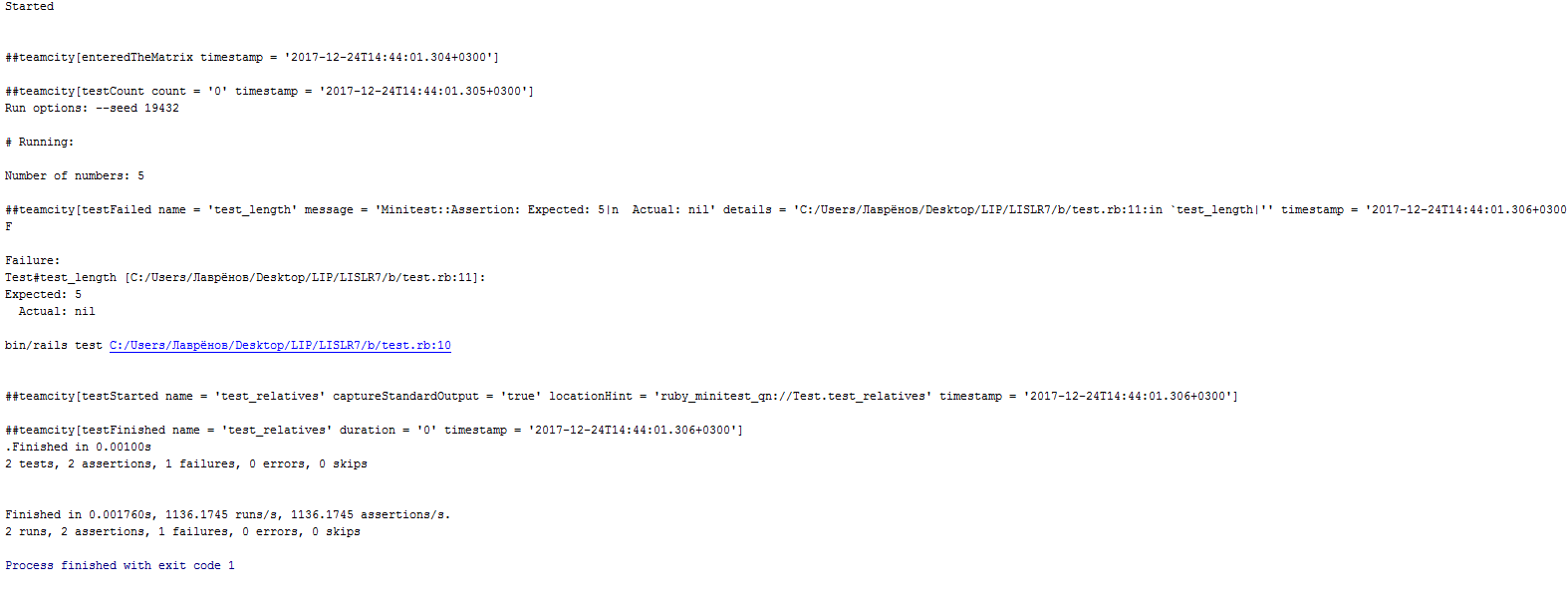


Рис.4. Выполнение теста

# Вывод

Было создано 2 консольных ruby приложений по выданному условию. Обе программы имеют тесты. Приложения протестированы и работают верно.