



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»

О Т Ч Е Т

по лабораторной работе № 9

Название: Создание простейших веб-приложений Ruby on Rails.
AJAX

Дисциплина: Языки Интернет-программирования

Студент

ИУ6-34Б

(Группа)

20.11.2022

(Подпись, дата)

С. А. Рахманов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

Д. В. Малахов

(И.О. Фамилия)

Цель работы: углубление теоретических сведений о принципах работы асинхронного веб-интерфейса и получение практических навыков создания веб-приложения с использованием средств Ruby on Rails и технологии AJAX.

Задание:

Все консольные приложения Ruby следует реализовывать в виде трех отдельных файлов:

1. основная программа;
2. программа для взаимодействия с пользователем через консоль;
3. программа для автоматического тестирования на основе MiniTest::Unit или RSpec. Везде, где это возможно, данные для проверки должны формироваться автоматически по правилам, указанным в задании.

При реализации программ везде, где это возможно, следует избегать использования циклов `for`, `do`, `while`. Вместо них используйте методы из примеси `Enumerable`.

Все тексты программ должны быть проверены на соответствие стилю программирования Ruby при помощи `rubocop.ru` или `reek`.

Разработать веб-приложение, имеющее HTML-страницу с формой ввода данных и HTML-страницу для представления результатов. Результат расчёта должен быть представлен в форме таблицы, оформленной с помощью элемента `table` или отдельными ячейками `div` и имеющей не менее двух колонок. Если по условию задания результат может быть представлен только в виде одной строки таблицы, необходимо реализовать вывод промежуточных результатов расчёта в качестве дополнительных строк. В этом случае первой колонкой таблицы будет порядковый номер итерации.

Под вводом с клавиатуры в тексте заданий следует понимать ввод в поле ввода данных формы на HTML-странице.

Текст задания:

Написать программу, которая вводит цепочку целых чисел (количество чисел не менее 10) и определяет наиболее длинную монотонно возрастающую их последовательность. Вывести на печать введенную цепочку, все найденные последовательности и наиболее длинную из них. При программировании использовать функцию.

Тексты измененных файлов

seq_controller.rb

```
def correct(arr)
  found = []
  tmp_found = []

  arr.take(arr.size - 1).each_index do |i|
    tmp_found << arr[i]
    unless arr[i] < arr[i + 1]
      found << tmp_found
      tmp_found = []
    end
  end

  tmp_found << arr[-1]
  found << tmp_found

  sizes = found.map(&:size)

  @max_found = found[sizes.index(sizes.max)].join(' ')
  @found = found.map { |x| x.join(' ') }.join(' | ')
end

# Top-level class
class SeqController < ApplicationController
  def input; end

  def view
    @str = !params[:str] || params[:str] == '' ? 'Error' : params[:str]
    @max_found = @str
    @found = @str
    arr = @str&.split&.map(&:to_i)
    arr && !arr&.empty? ? correct(arr) : nil

    respond_to do |format|
      format.html
      format.js
    end
  end
end
```

input.html.erb

```
<h1>Seq#input</h1>
<p>Find me in app/views/seq/input.html.erb</p>

<%= form_with url: '/seq/view.js', method: :get, local: false do |form| %>
  <%= form.label("Input string:") %> <br/>
  <%= form.text_field(:str, value: "0 1 5 5 5 10 20 25 -5 1 5 25 125 625") %>
  <br/><br/>
  <%= form.submit("Correct and print")%> <br/>
<% end %>

<div id="result"></div>
```

view.js.erb

```
document.getElementById('result').innerHTML = `
  <h3>Результаты:</h3>

  <%= ' <p>НЕТ ПОСЛЕДОВАТЕЛЬНОСТЕЙ</p>' if !@found %>
  <%= alert if @str&.match(/\s\d/).nil? %>

  <p id="str">str: <%= @str %></p>
  <p id="max found">max found: <%= @max_found %></p>
  <p id="found">found: <%= @found %></p>
```

```
`;  
routes.rb  
Rails.application.routes.draw do  
  get 'seq/input'  
  get 'seq/view'  
  root 'seq#input'  
  # Define your application routes per the DSL in  
  https://guides.rubyonrails.org/routing.html  
  
  # Defines the root path route ("/")  
  # root "articles#index"  
end
```

```
seq_controller_test.rb  
# frozen_string_literal: true  
  
require 'test_helper'  
  
class SeqControllerTest < ActionDispatch::IntegrationTest  
  test 'should get input' do  
    get seq_input_url  
    assert_response :success  
  end  
  
  test 'should get view' do  
    get seq_view_url  
    assert_response :success  
  end  
end
```

Результаты выполнения:



Lab9 x +

localhost:3000

Google Перев... Gmail – элект... > Нова

Seq#input

Find me in app/views/seq/input.html.erb

Input string:

0 1 5 5 5 10 20 25 -5 1 5 25 1:

Correct and print

Результаты:

str: 0 1 5 5 5 10 20 25 -5 1 5 25 125 625

max found: -5 1 5 25 125 625

found: 0 1 5 | 5 | 5 10 20 25 | -5 1 5 25 125 625

Результаты проверки анализатором rubocop:

```
Терминал
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab9/lab9/app$ rubocop controllers
Inspecting 2 files
..

2 files inspected, no offenses detected
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab9/lab9/app$
```

Тестирование с помощью Selenium IDE:

Selenium IDE - lab9*

Project: lab9*

Tests ▾ +

Search tests... 🔍

- ✓ test_1*
- ✓ test_2*

⏮ ⏪ ⏩ ⏭ ⌛ ⌚

http://localhost:3000 ▾

| | Command | Target | Value |
|----|-------------------|--------------------|--------------|
| 1 | ✓ open | / | |
| 2 | ✓ set window size | 908x1033 | |
| 3 | ✓ click | css=html | |
| 4 | ✓ click | css=html | |
| 5 | ✓ type | id=str | |
| 6 | ✓ click | name=commit | |
| 7 | ✓ mouse over | name=commit | |
| 8 | ✓ mouse out | name=commit | |
| 9 | ✓ assert text | css=h3 | Результаты: |
| 10 | ✓ assert text | css=#result > #str | str: Error |
| 11 | ✓ assert text | id=max found | max found: 0 |
| 12 | ✓ assert text | id=found | found: 0 |

Selenium IDE - lab9*

Project: lab9*

Tests +

Search tests...

http://localhost:3000

| | Command | Target | Value |
|---|-------------------|--------------------|---------------------------------------------------|
| 1 | ✓ open | / | |
| 2 | ✓ set window size | 908x1033 | |
| 3 | ✓ click | name=commit | |
| 4 | ✓ assert text | css=h3 | Результаты: |
| 5 | ✓ assert text | css=#result > #str | str: 0 1 5 5 10 20 25 -5 1 5 25 125 62 5 |
| 6 | ✓ assert text | id=max found | max found: -5 1 5 25 125 625 |
| 7 | ✓ assert text | id=found | found: 0 1 5 5 5 10 20 25 -5 1 5 25 125 625 |

Вывод: в результате выполнения лабораторной работы были получены практические навыки создания асинхронного веб-интерфейса, интеграционного тестирования с помощью среды Selenium IDE.