



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

## ОТЧЕТ

по лабораторной работе № 8

**Название:**

Создание веб-приложения с помощью RoR

**Дисциплина:** Языки интернет-программирования

Студент

ИУ6-35Б

(Группа)

\_\_\_\_\_  
(Подпись, дата)

К. А. Спасскова

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

\_\_\_\_\_  
(И.О. Фамилия)

Москва, 2022

## Вариант 19

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.

Дано натуральное число  $n$  и целые числа  $a_1, a_2, \dots, a_n$ . Написать программу, определяющую, есть ли среди заданной последовательности чисел отрезки (непрерывные подпоследовательности), состоящие степеней пятерки. Если есть, то подсчитать их количество и определить самый длинный из отрезков. Вывести на печать все отрезки, самый длинный из них и количество искоемых отрезков. Если таких отрезков нет, то соответствующее сообщение. При составлении программы использовать функции.

Контроллер:

```
# frozen_string_literal: true

# class of controller
class ElementsController < ApplicationController
  def index; end

  def self.power_of_5?(num)
```

```

    tmp = 0
    return false if num.zero?

    while tmp.zero?
      tmp = num % 5
      num /= 5
    end
    if tmp.eql?(1) && num.zero? then true
    else
      false
    end
  end
end

def self.segments_of_powers(array)
  res = []
  buf = []
  array.each do |elem|
    if power_of_5?(elem) then buf << elem
    elsif !buf.length.zero?
      res << buf
      buf = []
    end
  end
  res << buf unless buf.length.zero?
  res
end

def self.largest_segment(arr)
  unless arr.length.zero?
    max_subarr = arr[0]
    arr.each do |elem|
      max_subarr = elem if max_subarr.length < elem.length
    end
  end
  max_subarr
end

def result
  flash[:alert] = ElementsController.data_check(params[:length].to_i,
params[:str_elem])
  unless flash[:alert]
    @length_of_arr = params[:length].to_i
    @array = params[:str_elem].split.map!(&:to_i)
  end
end

```

```

    @array_of_pow = ElementsController.segments_of_powers(@array)
    @max_subarr = ElementsController.largest_segment(@array_of_pow)
  end
end

def self.data_check(len, str)
  if /^[^d^s]/.match(str) then "Sequence isn't correct"
  else
    arr = str.split.map!(&:to_i)
    "Length isn't correct" if arr.length != len
  end
end
end
end

```

Представления:

Index.html.erb:

```

<h1>Adding length of sequence and sequence</h1>
<p>
  Input sequence of positive integers separated by space
</p>
<%= form_with url: '/elements/result', method: :get do |form| %>
  <div class = 'form-group' style = 'width: 60%;'>
    <%= form.label :length %>
    <%= form.number_field(:length, {:class => 'form-control'}) %>
  </div>

  <div class = 'form-group' style = 'width: 60%;'>
    <%= form.label :sequence %>
    <%= form.text_field(:str_elem, {:class => 'form-control'}) %>
  </div>

  <div class = 'form-group' style = 'width: 60%; margin-top: 10px;'>
    <%= form.submit "calculate", class: 'btn btn-primary' %>
  </div>
<%end%>

```

Result.html.erb:

```

<head>
</head>
<body>
  <h1>Result of completing task</h1>

```

```

<%if flash[:alert]%>
  <div class = 'alert alert-danger' style = 'width:
60%;'><%=flash[:alert]%></div>
<%else%>
  <table class='table table-bordered' style = 'width: 60%;'>
    <tr>
      <th>Task</th>
      <th>Result of task</th>
    </tr>
    <tr>
      <td>Length of sequence</td>
      <td><%=@length_of_arr%></td>
    </tr>
    <tr>
      <td>Sequence</td>
      <td><%=@array%></td>
    </tr>
    <tr>
      <%if !@array_of_pow.length.zero?%>
        <td>Sequences of powers of 5</td>
        <td><%=@array_of_pow.each do |elem|%>
          <p><%=elem%></p>
        <%end%>
      </td>
    </tr>
    <tr>
      <td>Sequence of powers of 5 with max length</td>
      <td><%=@max_subarr%></td>
    </tr>
    <tr>
      <td>Number of sequences of powers of 5</td>
      <td><%=@array_of_pow.length%></td>
    </tr>
    <%else%>
      <p>There are no sequences of powers of 5</p>
    <%end%>
  </table>
<%end%>
<div><%=link_to "return to home page", "/"%></div>
</body>

```

elements\_controller\_spec.rb:

```

# frozen_string_literal: true

require 'spec_helper'
require 'rails_helper'

# require
'/home/ksusha/uni/labs_web/laba8/app/controllers/elements_controller'

# testing ElementsController
RSpec.describe ElementsController, type: :controller do
  it 'returns true if value is a power of 5' do
    expect(ElementsController.power_of_5?(0)).to eq(false)
    expect(ElementsController.power_of_5?(1)).to eq(true)
    expect(ElementsController.power_of_5?(5)).to eq(true)
  end

  it 'returns zero length array when there are no segments of powers of
5' do
    arr = [0, 6, 7, 24, 26, 90]
    expect(ElementsController.segments_of_powers(arr)).to eq([])
  end

  describe 'GET index' do
    it 'has a 200 status code' do
      get :index
      expect(response.status).to eq(200)
    end
  end
end

RSpec.describe ElementsController, type: 'request' do
  it 'expects to see sequences of powers of 5' do
    get '/elements/result?length=7&str_elem=1+2+3+4+5+6+7'
    expect(assigns[:length_of_arr]).to eq(7)
  end

  it 'alert message' do
    get '/elements/result?length=7&str_elem=1+2+3+4+5+6+q'
    expect(flash[:alert]).to eq("Sequence isn't correct")

    get '/elements/result?length=4&str_elem=1+2+3+4+5+6'
    expect(flash[:alert]).to eq("Length isn't correct")
  end
end

```

end

Выполнение тестов:

```
9cc193c174b36883 main -> main
• ksusha@ksusha-notebook:~/uni/labs_web/lab8$ rspec spec
.....

Finished in 0.13924 seconds (files took 1.08 seconds to load)
5 examples, 0 failures

o ksusha@ksusha-notebook:~/uni/labs_web/lab8$
```

Вид веб-приложения:

ActionDispatch:Flash × Laba8 × sp20ks/lab8 × style for flash rails - 1 × Rails Flash Messages × Alerts - Bootstrap × Новый документ - G... × Языки ИПТ: ЛР 8 × + × - ×

← → ↻ localhost:3000

## Adding length of sequence and sequence

Input sequence of positive integers separated by space

Length

Sequence

calculate

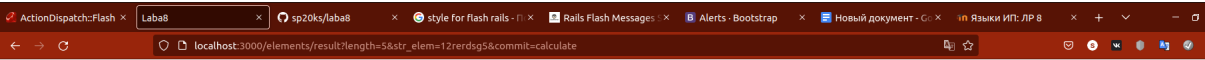
ActionDispatch:Flash × Laba8 × sp20ks/lab8 × style for flash rails - 1 × Rails Flash Messages × Alerts - Bootstrap × Новый документ - G... × Языки ИПТ: ЛР 8 × + × - ×

← → ↻ localhost:3000/elements/result?length=6&str\_elem=1+1+2+0+25+3&commit=calculate

## Result of completing task

Task	Result of task
Length of sequence	6
Sequence	[1, 1, 2, 0, 25, 3]
Sequences of powers of 5	[1, 1] [25]
Sequence of powers of 5 with max length	[1, 1]
Number of sequences of powers of 5	2

[return to home page](#)



## Result of completing task

Sequence isn't correct

[return to home page](#)



## Result of completing task

Length isn't correct

[return to home page](#)