

Denis Trenchev

Code:

```
require 'csv'

i = 0
arr1 = []
arr2 = []
arr3 = []

Dir.glob(ARGV[0]+"*.rb") do |first_folder|
  name = first_folder.split('/').last.split('.').first.split('_')

  if name.length == 3
    if name[1].to_s.length == 5
      arr1[i] = []
      arr[i][0] = name[0]
      arr[i][1] = name[1]
      i+=1
    end
  end
end
i = 0

Dir.glob(ARGV[1]+"*.rb") do |second_folder|
  name = second_folder.split('/').last.split('.').first.split('_')

  if name.length == 3
    if name[1].to_s.length == 5
      arr1[i] = []
      arr[i][0] = name_1[0]
      arr[i][1] = name_1[1]
      i+=1
    end
  end
end
i = 0

arr1.each do |compare1|
  arr2.each do |compare2|
    if compare2 == compare1
      arr3[i] = compare1
      i+=1
    end
  end
end

sort = arr3.sort_by{|asd| asd[1]}
CSV.open("students.csv", "w") do |csv|
  sort.each do |element|
    csv << element
  end
end
```

Problem:

Denis Trenchev is stupid idiot, who can not write ruby code!!!!!! TOTALY WRONG !
FK U ! Greetz, RAINBOW PATKA ! _(_

Fix:

DELETE the whole program and try again !

Rank:

Easy to understand: -100

Readable: 4

Working in most cases: 0

Overall: -32

Dimitar Nesterov

Code:

```
require 'csv'
def is_numeric(o)
  true if Integer(o) rescue false
end
array = []
count = 0
Dir.glob(ARGV[0] + "/*.rb") do |file|
  name = file.split("/").last.split(".").first.split("_")

  name[0] = name[0].to_s
  name[0] = name[0].capitalize
  name[1] = name[1].to_s
  name[1] = name[1].capitalize

  if name.size == 3 && is_numeric(name[2])
    if name[1].length == 10

      array[count] = []
      array[count][0] = name[0].to_s
      array[count][1] = " #{name[1].to_s}"
      count += 1

    end
  end
end
array = array.sort_by {|el| -el[1]}
CSV.open("result.csv", "w") do |csv|

  array.uniq.each do |e|

    csv << e

  end
end
```

Problem:

There is a problem with the sorting. The student should sort the result by Last Name DESC not by Last Name ASC and also remove the dash before "el[1]".

Fix:

change: `array = array.sort_by {|el| -el[1]}`
to: `array = array.sort_by {|el| el[1]}.reverse`

Rank:

Easy to understand: 4

Readable: 4

Working in most cases: 1

Overall: 3

Dimitar Terziev

Code:

```
require 'csv'
arr = []
Dir.glob("#{ARGV[0]}*.rb*"){|file|
  file_str = file.split('/').last
  if(file_str =~ /\A[a-zA-Z]+\_[a-zA-Z]+\_\d+2 \.rb\z/ && file_str.split('_')[1].size == 5)
    arr.push("#{file_str.split('_')[1]} #{file_str.split('_').first}")
  end
}
CSV.open('result.csv', 'w'){|csv|
  arr.uniq.sort.each{|el|
    csv << "#{el.split(' ').last} #{el.split(' ').first}".split(' ')
  }
}
```

Problem:

The student checks first and last name if they contain only letters, but in the fixtures names can also include numbers.

Fix:

change the if statement:

```
if(file_str =~ /\A[a-zA-Z]+\_[a-zA-Z]+\_\d+2 \.rb\z/ && file_str.split('_')[1].size == 5)
```

to:

```
if(file_str.split('_')[1].size == 5)
```

Rank:

Easy to understand: 4

Readable: 4

Working in most cases: 4

Overall: 4

Georgi Ivanov

Code:

```
require "csv"

arr = []
i = 0

Dir.glob(ARGV[0]+"*.rb") do |file|
  name = file.split('/').last.split('.').first.split('_')
  firstname = name[0]
  lastname = name[1]
  exercise = name[2]

  if firstname == '' || lastname == '' || exercise == ''
  elsif name.length == 3

    if lastname.length == 5
      arr[i] = []
      arr[i][0] = name[0]
      arr[i][1] = name[1]
      i+=1
    end
  end
end

daiba = arr.sort_by{|asd| asd[0]}.reverse!
CSV.open("result.csv", "w") do |csv|
  daiba.each do |element|
    csv << element
  end
end
```

Problem:

There is no problem. Program works fine!

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 5

Overall: 5

Hristo Dachev

Code:

```
require 'csv'

hash = Hash.new

Dir.glob("#{ARGV[0]}*").each do |path|
  first_name = path.split("/").last.split("_").first
  last_name = path.split("/").last.split("_", 2).last.split("_").first
  digit = path.split("/").last.split("_",
2).last.split("_").last.split(".").first
  name = path.split("/").last

  if name.include? "_" then counter = name.count "_" end

  if (counter != 2) || (digit.to_i.to_s != digit)
    l = name.length
    hash[name] = l
  end
end
Dir.glob("#{ARGV[1]}*").each do |path|
  first_name = path.split("/").last.split("_").first
  last_name = path.split("/").last.split("_", 2).last.split("_").first
  digit = path.split("/").last.split("_",
2).last.split("_").last.split(".").first

  name = path.split("/").last
  if name.include? "_" then counter = name.count "_" end

  if (counter != 2) || (digit.to_i.to_s != digit)
    l = name.length
    hash[name] = l
  end
end

CSV.open("result.csv", "w") do |csv|
  hash.sort_by{ |k, v| v }.each do |name, length|
    csv << ["#{name}", "#{length}"]
  end
end
```

Problem:

There is no problem. Program works fine!

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 5

Overall: 5

Ivelin Slavchev

Code:

```
require 'csv'
result = Hash.new
Dir.glob(ARGV[0] + "*").each do |file1|
  short1 = file1.split("/").last
  ext1 = short1.split(".").last
  names1 = short1.split(".").first
  digit1 = file1.split("_").last
  if (ext1 != "rb") or (digit1.to_i.to_s != digit1) or (short1.scan("_").count !=
2)
    result[short1] = short1.length
  end
end
Dir.glob(ARGV[1] + "*").each do |file2|
  short2 = file2.split("/").last
  ext2 = short2.split(".").last
  names2 = short2.split(".").first
  digit2 = file2.split("_").last
  if (ext2 != "rb") or (digit2.to_i.to_s != digit2) or (short2.scan("_").count !=
2)
    result[short2] = short2.length
  end
end
result.sort_by{|k, v| v}
CSV.open("result.csv", "w") do |csv|
  result.each do |p|
    csv << p
  end
end
```

Problem:

There is a problem with a variable name and the sorting.

Fix:

```
change: (digit2.to_i.to_s != digit)
to: (digit2.to_i.to_s != digit2)
remove: result.sort_by{|k, v| v}
change: result.each do |p|
to: result.sort_by{|k, v| v}.each do |p|
```

Rank:

Easy to understand: 4

Readable: 4

Working in most cases: 1

Overall: 3

Ivo Valchev

Code:

```
hash_fold1={}
hash_fold2={}

Dir.glob("#{ARGV[0]}*.*") do |file|
  name = file.split("/").last.split(".").first.split("_")
  isNum = Integer(name[2]) rescue nil
  if name[0] and name[1] and name[0].length == 5 and !isNum!=nil
hash_fold1.include?(name[0])
    hash_fold1["#{name[1]}"] = "#{name[0]}"
  end
end
Dir.glob("#{ARGV[1]}*.*") do |file|
  name = file.split("/").last.split(".").first.split("_")
  isNum = Integer(name[2]) rescue nil
  if name[0] and name[1] and name[0].length == 5 and !isNum!=nil and!
hash_fold2.include?(name[0])
    hash_fold2["#{name[1]}"] = "#{name[0]}"
  end
end
File.open("result.csv", "w") do |csv|
  hash_fold1.sort.map do |key, value|
    if (hash_fold1[key]==hash_fold2[key])
      csv.puts("#{key},#{value}")
    end
  end
end
```

Problem:

There are two syntaxs errors. Also creating and writing in .csv file is wrong.

Fix:

```
change: !isNum!=nil hash_fold1.include?(name[0])
to:      !isNum!=nil and hash_fold1.include?(name[0])
```

Rank:

Easy to understand: 2

Readable: 2

Working in most cases: 1

Overall: 1.6

Kalin Marinov

Code:

```
require 'csv'

hash = Hash.new

Dir.glob("#{ ARGV[0] }/*") do |name|
  name = name.split("/").last
  short_name = name.split('_')[1]
  if short_name.length == 5
    hash[name] = short_name
  end
end

CSV.open("result.csv", "w") do |csv|
  hash = hash.sort_by { |key, value| value }.reverse
  hash.each |key| do
    csv << key
  end
end
```

Problem:

Wrong file name and wrong each loop.

Fix:

Change file name to: `Kalin_Marinov_12_bce70c.rb`

Change: `hash.each |key| do`

To: `hash.each do |key|`

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 1

Overall: 3.6

Kamena Dacheva

Code:

```
student = Hash.new { |name, programs| name[programs] = [] }
directory = ARGV[0]
require "csv"

class String
  def is_number?
    Float(self) != nil rescue false
  end
end

Dir.glob("#{directory}/*.*)" do |my_repository|

  name_dir = my_repository.split("/").last

  name = name_dir.split("_").first.capitalize
  sir_name = name_dir.split("_", 2).last.split("_").first.capitalize
  program = name_dir.split("_").last.split(".").first
  ex = name_dir.split("_").last.split(".").last

  if name_dir.include? "_" then counter = name_dir.count "_" end
  student["#{name}"] << sir_name if ((counter == 2) && (sir_name.length == 5) &&
(program.is_number?) && (ex == "rb"))
end

CSV.open("result.csv", "w") do |csv|
  student.sort_by{|k, v| v}.reverse.each do |f_name, l_name|
    csv << [f_name, l_name].flatten
  end
end
```

Problem:

Wrong sorting.

Fix:

Change: `student.sort_by{|k, v| v}.reverse.each`
To: `student.sort_by{|k, v| k}.reverse.each`

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 4

Overall: 4.6

Kristina Pironkova

Code:

```
require 'csv'
results=Hash.new
Directory = ARGV[0]
Dir.glob("#{Directory}/*.rb") do |file_name|

  first_name = file_name.split("/").last.split("_").first.capitalize
  last_name=file_name.split("/").last.split("_",2).last.split("_").first.capitalize

  if first_name.length == 10

    results["#{last_name}"] = "#{first_name}"
  end
end

CSV.open("results.csv", "w") do |csv|
  results.sort.each do |first,last|

    csv << [last,first]

  end
end
```

Problem:

Wrong result file name and sorting.

Fix:

Rename: results.csv
To: result.csv

Change: result.sort.each
To: result.sort.reverse.each

Rank:

Easy to understand: 5

Readable: 4

Working in most cases: 4

Overall: 4.3

Lubomir Yankov

Code:

```
require 'csv'
def is_numeric(o)
  true if Integer(o) rescue false
end

array = []
count = 0

Dir.glob(ARGV[0] + "**").each do |file|
  ch_count = 0
  file_name = file.split("/").last.split("")

  file_name.each do |ch|

    if is_numeric(ch)

      ch_count += 1

    end

  end
  if ch_count == 9
    len = file_name.length
    array[count] = []
    array[count][0] = file_name
    array[count][1] = len/2.round
    count += 1
  end
end

array = array.sort_by {|e1| e1[0]}
CSV.open("results.csv", "w") do |csv|

  array.each do |element|

    csv << element

  end

end
```

Problem:

Totally wrong!

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 4

Overall: 4.6

Marian Belchev

Code:

```
require 'csv'

hash1 = Hash.new
hash2 = Hash.new

Dir.glob("#{ARGV[0]}*_**.rb") do |file1|
  Dir.glob("#{ARGV[1]}*_**.rb") do |file2|
    firstName1 = file1.split("/").last.split("_").first
    lastName1 = file1.split("/").last.split("_", 2).last.split("_").first
    number1 = file1.split("_").last.split(".").first

    firstName2 = file2.split("/").last.split("_").first
    lastName2 = file2.split("/").last.split("_", 2).last.split("_").first
    number2 = file2.split("_").last.split(".").first

    hash1[firstName1] = lastName1 + "." + number1
    hash2[firstName2] = lastName2 + "." + number2
  end
end

CSV.open("results.csv", "w") do |csv|
  hash2.sort.each do |key, value|
    if !hash1.has_key?(key) && !hash1.has_value?(value.split(".").first) && !
hash1.has_value?(value.split(".").last.to_i)
      csv << [key, value.gsub('.', '')]
    end
    if hash1.has_key?(key) && !hash1.has_value?(value.split(".").first) && !
hash1.has_value?(value.split(".").last.to_i)
      csv << [key, value.gsub('.', '')]
    end
  end
end
```

Problem:

No problems !!! Marian Belchev is sooo smart, cool, beautiful and is the best Ruby programmer!!!!
Deserves excellent mark !

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 5

Overall: 5

Momchil Angelov

Code:

```
require 'csv'

arr1=Array.new
arr2=Array.new
arr3=Array.new
a = ARGV[0]
b = ARGV[1]
i=0
Dir.glob(a + "/*.rb") do |my_text_file1|
  short= my_text_file1.split('/').last
  length1 = short.length
  shorter= short.split('.').first.split('_')
  first_name=shorter[0]
  last_name=shorter[1]
  digits=shorter[2].to_i

  if !first_name || !last_name || digits=0
    next
  else
    arr1 << ["#{short}" "#{length1}"]
  end
end
Dir.glob(b + "/*.rb") do |my_text_file2|

  short2= my_text_file2.split('/').last
  length2 = short2.length
  shorter2= short2.split('.').first.split('_')
  first_name2=shorter2[0]
  last_name2=shorter2[1]
  digits2=shorter2[2].to_i

  if !first_name2 || !last_name2 || digits2=0
    next
  else
    arr2 << ["#{short2}", "#{length2}"]
  end
end

arr3 = arr1 & arr2

arr3 = arr3.sort_by {|e1|
  e1[1]
}

CSV.open("result.csv", "w") do |csv|

  arr3.each do |element|
    csv << element
  end
end
```

Problem:

Unfinished, wrong variable and dont know how to compare.

Rank:

Easy to understand: 2

Readable: 2

Working in most cases: 1

Overall: 1.3

Moretti Georgiev

Code:

```
require 'csv'

student = Hash.new

Dir.glob("#{ARGV[0]}*_**.rb") do |file|
  firstName = file.split("/").last.split("_").first
  lastName = file.split("/").last.split("_", 2).last.split("_").first
  digit = file.split("/").last.split("_").last.split(".").first
  if lastName.length == 10
    student[firstName] = lastName
  end
end

CSV.open("result.csv", "w") do |csv_file|
  student.sort.each do |key, value|
    csv_file << ["#{key}, #{value}"]
  end
end
```

Problem:

There is only one problem. There is space after the coma.

His brain is too high above the ground and can not works properly!

Fix:

Change: `csv_file << ["#{key}, #{value}"]`
To: `csv_file << ["#{key},#{value}"]`

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 5

Overall: 5

Nikola Marinov

Code:

```
require 'csv'
def is_numeric(o)
  true if Integer(o) rescue false
end
array=[]
count=0

Dir.glob(ARGV[0] + "**/*.").each do |file|
  full_name=file.split("/").last
  name = file.split("/").last.split(".").first_split("_")
  if name.lenght != 3 && !is_numeric(name[2])
    array(count) = []
    array(count) [0]=full_name
    array(count)[1]= full_name.to_s.lenght
    count += 1
  end
end

Dir.glob(ARGV[0] + "**/*.").each do |file|
  full_name=file.split("/").last
  name = file.split("/").last.split(".").first_split("_")
  if name.lenght != 3 && !is_numeric(name[2])
    array(count) = []
    array(count) [0]=full_name
    array(count)[1]= full_name.to_s.lenght
    count += 1
  end
end

array = array.sort_by{|el| el[0]}
CSV.open("task.csv",w) do |csv|
  array=uniq.each do |element|
    csv << element
  end
end
```

Problem:

Wrong result file name. Strange defining of array

Fix:

Can not be fixed! And always late for class!

Rank:

Easy to understand: 3

Readable: 2

Working in most cases: 1

Overall: 1.6

Petko Bozhinov

Code:

```
require 'csv'

class String
  def numeric?
    Float(self) != nil rescue false
  end
end

output = Array.new
i = 0
Dir.glob(ARGV[0] + "/*") do |file|
  file = file.split('/').last.split('.').first.split('_')
  Dir.glob(ARGV[1] + "/*") do |file2|
    file2 = file2.split('/').last.split('.').first.split('_')
    if "#{file[0]} #{file[1]}" == "#{file2[0]} #{file2[1]}"
      if file[2].numeric?
        if file[0].to_s.length == 5
          output[i] = Array.new
          output[i][0] = file[0]
          output[i][1] = file[1]
          i+=1
        end
      end
    end
  end
end

output = output.sort_by{ |element| element[1]}
CSV.open("result.csv", "w") do |csv|
  output.each do |pusher|
    csv << pusher
  end
end
```

Problem:

Wrong calling defined class and its method.

Rank:

Easy to understand: 4

Readable: 5

Working in most cases: 1

Overall: 3.3

Radoslav Kostadinov

Code:

```
require 'csv'
file1 = Hash.new
file2 = Hash.new
path1 = ARGV[0]
path2 = ARGV[1]

Dir.glob("#{path1}*.rb") do |my_text_file|
  s = my_text_file.split(/\//).last.capitalize
  first_name = my_text_file.split("/").last.split("_").first
  last_name = my_text_file.split("/").last.split("_",2).last.split("_").first

  if s.count('_') == 2 and !((first_name == "" || first_name == " ") ||
(last_name == "" || last_name == " "))
    file1[first_name] = last_name
  end
end

Dir.glob("#{path2}*.rb") do |my_text_file|
  s = my_text_file.split(/\//).last.capitalize
  first_name = my_text_file.split("/").last.split("_").first
  last_name = my_text_file.split("/").last.split("_",2).last.split("_").first

  if s.count('_') == 2 and !((first_name == "" || first_name == " ") ||
(last_name == "" || last_name == " "))
    file2[first_name] = last_name
  end
end

CSV.open("result.csv", "w") do |csv|
  file1.sort.each do |first_name, last_name|
    file2.sort.each do |first_name1, last_name1|
      if first_name1 == first_name and last_name1 == last_name
        begin
          end
        else
          csv << [last_name1, first_name1]
        end
      end
    end
  end
end
end
```

Problem:

Wrong loops while writing in result file and spamming in the official_results_a.csv file.

Rank:

Easy to understand: 1

Readable: 1

Working in most cases: 1

Overall: 0

Simeon Shopkin

Code:

```
require 'csv'

arr = Array.new
Dir.glob(ARGV[0]+"/*.rb") do |first_files|
  Dir.glob(ARGV[1]+"/*.rb") do |second_files|
    first_files = first_files.split("/").last.split(".").first.split("_")
    if first_files.size != 3
      if first_files != second_files
        print_count = first_files.split("/").last.split(".").first
        p = print_count.size.to_s
        print =
first_files[0].capitalize+"_"+first_files[1].capitalize+"_"+first_files[2]+", "+p
        arr.push(print)
      end
    end
  end
end

CSV.open("result.csv", "w") do |csv|
  arr.sort.each do |element|
    csv << [element]
  end
end
```

Problem:

The whole program is wrong! He did not understood the task.

Rank:

Easy to understand: 3

Readable: 4

Working in most cases: 1

Overall: 2.6

Stanimir Bogdanov

Code:

```
require 'csv'

directory = ARGV[0]
students = Hash.new

Dir.glob("#{directory}*") do |filename|
  unless (filename.split('/').last =~ /^[a-zA-Z0-9]+_[a-zA-Z0-9]+_[0-9]+.rb$/).nil?
    first_name = filename.split('/').last.split('_')[0]
    second_name = filename.split('/').last.split('_')[1]
    students[first_name] = second_name if first_name.length == 10
  end
end

CSV.open("result.csv", "w") do |csv|
  Hash[students.sort_by { |first, last| last }.reverse].each do |first, last|
    csv << [ first, last ]
    # puts "#{first},#{last}"
  end
end
```

Problem:

Stanimir is TOO SMART for this EASY program!

Rank:

Easy to understand: 1

Readable: 1

Working in most cases: 1

Overall: 1

Stanislav Gospodinov

Code:

```
require 'csv'
hash = Hash.new

Dir.glob("#{ARGV[0]}*.rb") do |file|
  filename = file.split('/').last.split('.').first;
  if filename.split('_').length == 3
    if filename.split('_')[1].length == 5
      hash[filename.split('_')[0]] = filename.split('_')[1]
    end
  end
end

hash = Hash[hash.sort_by{|k, v| v}]

CSV.open("results.csv", "w") do |csv|
  hash.each do |key, value|
    csv << [key, value].flatten
  end
end
```

Problem:

Wrong result file name.

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 5

Overall: 5

Stanislav Valkanov

Code:

```
require 'csv'
a = Hash.new
path = ARGV[0]
Dir.glob(path + "**/*.rb") do |my_text_file|
  short_name = my_text_file.split('/').last.split('.').first
  name = short_name.split("_")[0]
  last = short_name.split("_")[1]
  last.to_s
  if (last.length == 5)&&(short_name.split("_").size == 3)
    a["#{name}"] = last
  end
end
CSV.open("result.csv", "w") do |csv|
  Hash[a.sort.reverse].each do |element|
    csv << element
  end
end
```

Problem:

Works fine!

Rank:

Easy to understand: 4

Readable: 0

Working in most cases: 5

Overall: 3

Tihomir Lidanski

Code:

```
require 'csv'

Dir.glob(ARGV[0] + "/*.") do |file|
  name = file.split("/").last.split(".")

  Dir.glob(ARGV[1] + "/*.") do |file|

    puts name.length % 2.round()

  end
end

CSV.open("result.csv", "w") do |csv|

end
```

Problem:

Unfinished program.

Rank:

Easy to understand: 3

Readable: 1

Working in most cases: 10000000 ← points for trying

Overall: 3333334.3

Veselin Dechev

Code:

```
require 'csv'
result = Hash.new
Dir.glob(ARGV[0] + "*.rb").each do |first|
  name1 = first.split("/").last.capitalize
  first_name = name1.split("_").first.capitalize
  last_name = name1.split("_", 2).last.split('_').first.capitalize
  Dir.glob(ARGV[1] + "*.rb").each do |second|
    name2 = second.split("/").last.capitalize
    if (name1 == name2)
      result.compare_by_identity
      result[first_name] = last_name
    end
  end
end
end
CSV.open("result.csv", "w") do |csv|
  result.sort_by{|k, v| k}.each do |element|
    csv << element
  end
end
```

Problem:

Wrong file name. **НЕ Е ТЪП ДА СИ ЛЕСНО!!!!**

Fix:

Rename file name to: **Veselin_Dechev_2_5f1c22.rb**

Rank:

Easy to understand: 0

Readable: 0

Working in most cases: 0

Overall: 0

Borislav Rusinov

Code:

```
a=ARGV[0]
require 'csv'
array=[]
Dir.glob("#{a}*.*)" do |my_text_file|
  name = my_text_file.split("/").last.split(".").first.split("_")
  if name[1]!=nil && name[0].length==10
    array << name[0] + "," + name[1]
  end
end
array.sort!
array.reverse!
File.open("results.csv", "w") do |csv|
  array.each do |arg|
    csv.puts(arg)
  end
end
```

Problem:

Bobo is really really stupid, ugly, fat and the biggest G@Y EVER!!!!

Fix:

Chnge file name to: **Borislav_Rusinov_1_6fb3ad.rb**

Change result file name to: **result.csv**

Rank:

Easy to understand: 5

Readable: 5

Working in most cases: 5

Overall: 10 <3<3<3