



Team name:

CODE *PJMP\$*

Featuring :

Denis Stoinev, Iliyan Germanov, Stefan Iliev

Дата: 27.10.2014 г.



There were errors in 50% of the programs from class008_test (basically 1 or 2 errors in wrong program).

But there were rare cases like Nikola_Marinov_20_add57e.rb with 17 errors in 40 lines of code, too. We found 6 correctly working programs.

Common errors:

Error type “undefined method ” for nil:NilClass (NoMethodError) “: This was the most common error type in the wrong programs. It occurs when you try to use method to “nil” (for example “.length”, ‘+’ and etc).

Code example:

```
1.Dir.glob("#{ARGV[0]}/**/*.rb") do |current_file|
2.name = current_file.split('/').last.split(/_/)
3.if name[1].length == 5
4.  if not students_names.include?("#{name[1]}", "#{name[0]}") then
5.      students_names << (["#{name[1]}", "#{name[0]}"])
6.  end
7.end
```

If the directory “#{ARGV[0]}/**/” contains file.rb which not include ‘_’, on line 2 the array “name” will contain only nil values. Then on line 3 “name[1]” will be nil and when you try method “length” on “name[1]” it will throw error.

Solution:

```
1.Dir.glob("#{ARGV[0]}/**/*.rb") do |current_file|
2.name = current_file.split('/').last.split(/_/)
3.if name[1] != nil and name[1].length == 5
4.  if not students_names.include?("#{name[1]}", "#{name[0]}") then
5.      students_names << (["#{name[1]}", "#{name[0]}"])
6.  end
7.end
```

On line 3 add a check if name[1] != nil.

Error type “undefined local variable or method `name' for main:Object (NameError)”:

This type of error mostly occurs because the scope of variables in ruby or wrong name of variable.

Code example:

```
1.Dir.glob("#{ARGV[0]}/*") do |path|
2.  name = path.split(/\/\//).last
3.end
4.
5.for i in 0..2
6.  result = name
7.  puts result
8.end
```

In this case the variable “name” is only visible in 1-3 line. So when you try to pass the value of “name” to “result” it throws error.

Solution:

```
0.name = ""
1.Dir.glob("#{ARGV[0]}/*") do |path|
2.  name = path.split(/\/\//).last
3.end
4.
5.for i in 0..2
6.  result = name
7.  puts result
8.end
```

Use global variables. At line 0 we define name with value “”. The variable name is visible in the whole problem and results will be equal to the last path Dir.glob() returns.

Common problems:

The most common problem:

“No such file or directory - result.csv. No result.csv file was found.”

Its solution its simple just check how you name your output csv.

There were some sorting issues, too:

The solution here is to learn how to use sort_by or other sorting method.

Some students had syntax problems.

These errors could be avoided at first place with N simple steps:

- 1.Read task's condition carefully.
- 2.Use only functions and methods which you know how they work.
- 3.Expect invalid files (not in the correct format) and try to ignore them.

APPENDIXES:

David_Georgiev_12_1eea4f.rb

program code:

```
require 'csv'
students_names = []
Dir.glob("#{ARGV[0]}/**/*.*.rb") do |current_file|

  name = current_file.split('/').last.split(/_/)
  if name[1].length == 5
    if not students_names.include?("#{name[1]}", "#{name[0]}") then
      students_names << ["#{name[1]}", "#{name[0]}"]
    end
  end
end
end
CSV.open("result.csv", "w") do |csv|
  students_names.sort.each do |last, first|
    csv << ["#{first}", "#{last}"]
  end
end
end
```

Problems (Errors: 1):

Common error:

"David_Georgiev_12_1eea4f.rb:20:in `block in <main>': undefined method `length' for nil:NilClass (NoMethodError) from ./results/b/results/David_Georgiev_12_1eea4f.rb:17:in `glob' from ./results/b/results/David_Georgiev_12_1eea4f.rb:17:in `<main>'"

Solution: if name[1] != nil && name[1].length == 5

Program rank: 4

Iliyan_Germanov_17_f8b0d9.rb

program code:

```
require 'csv'
results = Hash.new
results.compare_by_identity
def is_number(str)
  str[/[0-9]+/] == str
end
Dir.glob("#{ARGV[0]}/*.rb") do |path1|
```

```

filename1 = path1.split(/\/).last
if filename1.count("_") == 2
  firstname1 = filename1.split("_").first
  lastname1 = filename1.split("_")[1]
  digit1 = filename1.split("_")[2].split(".").first
  if is_number(digit1)
    flag = 0
    Dir.glob("#{ARGV[1]}/*.rb") do |path2|
      filename2 = path2.split(/\/).last
      if filename2.count("_") == 2
        digit2 = filename2.split("_")[2].split(".").first
        if is_number(digit2)
          name1 = firstname1 + lastname1
          name2 = filename2.split("_").first +
filename2.split("_")[1]
          if name1 == name2
            flag = 1
            break
          end
        end
      end
    end
  end
  if flag == 0
    results[lastname1] = firstname1
  end
end
end
end

CSV.open("result.csv", "w") do |csv|
  results.sort_by{|key, val| key}.each do |el|
    csv << el
  end
end

```

Problems (Errors: 0):

Program is working correctly!

Wrong expected results:

Expected: LastName4,FirstName4 L5le3_3,F5le3

Output csv: L5le3,F5le3

Program rank: 5

Stefan_Iliev_28_d77aee.rb

program code:

```
require 'csv'
first_folder = ARGV.shift
second_folder = ARGV.shift || "err"
names_hash = Hash.new

Dir.glob(first_folder+"/*.*").each do |text_file|
  text_file = text_file.split("/").last
  if (text_file.split("_").length == 3) then
    first_name = text_file.split("_")[0]
    second_name = text_file.split("_")[1]
    diggit = text_file.split("_")[2].split(/\./).first
    if (diggit.to_i.to_s != diggit) then names_hash[text_file] = text_file.length end
  end
end
```

```

        if (first_name =~ /\d/) then names_hash[text_file] = text_file.length end
        if (second_name =~ /\d/) then names_hash[text_file] = text_file.length end
    else
        names_hash[text_file] = text_file.length
    end
end
if second_folder != "err"
    Dir.glob(second_folder+"/*.*").each do |text_file|
        text_file = text_file.split("/").last
        if (text_file.split("_").length == 3) then
            first_name = text_file.split("_")[0]
            second_name = text_file.split("_")[1]
            diggit = text_file.split("_")[2].split(/\./).first
            if (diggit.to_i.to_s != diggit) then names_hash[text_file] = text_file.length
        end
        if (first_name =~ /\d/) then names_hash[text_file] = text_file.length end
        if (second_name =~ /\d/) then names_hash[text_file] = text_file.length
    end
    else
        names_hash[text_file] = text_file.length
    end
end
end
names_hash = Hash[names_hash.sort_by{ |k,v| k } ]
names_hash = Hash[names_hash.sort_by{ |k,v| v } ]
puts names_hash

CSV.open("results.csv","w") do |csv|
    names_hash.each do |element|
        csv << element
    end
end
end

```

Problem (Errors: 0):

Common problem: "No such file or directory - result.csv. No result.csv file was found."

Solution: "results.csv" => "result.csv"

Program rank: 5

Stanislav_Iliev_26_627d43.rb

program code:

```
require 'csv'
name_array = Array.new()
name_array2 = Array.new()
support_array = Array.new()
support_array2 = Array.new()
i = 0
dir1 = ARGV[0]
dir2 = ARGV[1]

Dir.glob("#{dir1}/*.*") do |file|
  name_array[i] = file.split(/\/).last
  i += 1
end
count = i
i = 0
Dir.glob("#{dir2}/*.*") do |file2|
  name_array2[i] = file2.split(/\/).last
  i += 1
end
```



```

end
i = 0
for check in i..count
  if name_array[check] != name_array2[check]
    support_array[i] = name_array[check]
    support_array2[i] = name_array2[check]
    i += 1
    puts support_array
    puts support_array2
    CSV.open("result.csv", "w") do |csv|
      support_array.each do |element|
        csv << [element]
      end
    end
    CSV.open("result.csv", "w") do |csv|
      support_array2.each do |element2|
        csv << [element2]
      end
    end
  end
end
end
end

```

Problems (Errors: 0):

Solution: There is no quick solution.

Program rank: 3

Nikolay_Mihailov_25_f70059.rb

program code:

```

require 'csv'
hash = Hash.new
count = 0
Dir.glob(ARGV[0] + "/*.rb") do |file|

  first = file.split(/\/).last
  puts first

  #for (i = 0;i < first.length;i+=1)
  size = first.length
  i = 0
  first.each do |element|

    c = first[i].chr
    if element == 0 || element == 1 || element == 2 || element == 3
      || element == 4 || element == 5 || element == 6 || element == 7 || element == 8 ||
      element == 9

      count += 1
    end
  end
end

```

```

        end
      end
      puts count
    end

    Dir.glob(ARGV[1] + "/*.rb") do |secFile|
      sec = secFile.split(/\/).last
      #puts sec

    end

    CSV.open("result.csv", "w") do |csv|
      hash.sort_by{|key,val| key}.each do |element|
        csv << element
      end
    end
  end
end

```

Problems (Errors: 1):

Error:

```

"Nikolay_Mihailov_25_f70059.rb:26:in `block in <main>': undefined method `each' for
"F10letter3_L10letter4.rb":String (NoMethodError) from
../results/b/results/Nikolay_Mihailov_25_f70059.rb:18:in `glob' from
../results/b/results/Nikolay_Mihailov_25_f70059.rb:18:in `<main>' "

```

Solution: There is no quick solution.

Program rank: 3

Borislav_Stratev_2_a65be5.rb

program code:

```

require 'csv'
a = Array.new
h = Hash.new
Dir.glob("#{ARGV[0]}/*.rb") do |dir_file_name_1|
  Dir.glob("#{ARGV[1]}/*.rb") do |dir_file_name_2|

    file_name_1 = dir_file_name_1.split(/\/).last.to_s
    file_name_2 = dir_file_name_2.split(/\/).last.to_s

    if(file_name_1 != file_name_2)
      file_name = file_name_1
      digit = file_name.split(/_/).last.split(/\.).first.to_s
      first_name = file_name.split(/_/).first.to_s
      full_first_name = first_name + digit
      full_first_name = full_first_name.to_s
      tmp = file_name.split("#{first_name}_")
      full_last_name = tmp.last.split(/_/).first.to_s + digit
      full_last_name = full_last_name.to_s
    end
  end
end

```

```

            h[full_last_name] = full_first_name
        end
    end
end

CSV.open("results.csv", "w") do |csv|
    a = h.sort
    a.each do |element|
        csv << element
    end
end

```

Problems (Errors: 0):

Common problem: "No such file or directory - result.csv. No result.csv file was found."

Solution: There is no quick solution.

"results.csv" => "result.csv" (only for the common problem)

Program rank: 3

Lili_Kokalova_22_e0ea9c.rb

program code:

```

require 'csv'
student = Array.new
student1 = Array.new

Dir.glob(ARGV[0]+"/**/*.*.rb").each do |file_name1|
    file_name = file_name1.split("/").last
    first_name = file_name.split("/").last.split("_").first
    p first_name
    last_name = file_name.split("/").last.split("_",2).last.split("_").first
    #task = file_name.split("_").last.split(".").first
    student << ["#{first_name}", "#{last_name}"]
end

Dir.glob(ARGV[1]+"/**/*.*.rb").each do |file_name1|
    file_name = file_name1.split("/").last
    first_name = file_name.split("/").last.split("_").first
    p first_name
    last_name = file_name.split("/").last.split("_",2).last.split("_").first

```

```

        #task = file_name.split("_").last.split(".").first
        student1 << ["#{first_name}", "#{last_name}"]
    end

    CSV.open("result.csv", "w") do |csv|
        student.each do |fn, ln|
            student1.each do |fn1, ln1|
                if fn != fn1
                    if ln != ln1
                        csv << ["#{fn1}", "#{ln1}"]
                    end
                end
            end
        end
    end
end
end
end

```

Problems (Errors: 0):

Solution: There is no quick solution.

Program rank: 3

Valentin_Varbanov_4_041472.rb

program code:

```

students_first_dir = Array.new
students_second_dir = Array.new

for i in 0..1
    directory = ARGV[i]
    if ARGV[i].split(/\/).last(1).to_s == "/"
        directory += "**/*.rb"
    else
        directory += "/*/*/*.rb"
    end
    Dir.glob(directory).each do |dir|
        student = dir.split(/\/)
        if i == 0
            students_first_dir.push(student)
        else
            students_second_dir.push(student)
        end
    end
end

```

```

end

studentscsv = Array.new

students_first_dir.each do |std|
  match = 0
  students_second_dir.each do |std2|
    name = std.last.split(/_/)

    name2 = std2.last.split(/_/)
    for i in 0..1
      if name[i] == name2[i]
        match = 1
      end
    end
  end
end

end
studentscsv.push(name[1], name[2])
end

CSV.open("result.csv", "w") do |csv|
  studentscsv.each do |string|
    csv << string
  end
end
end

```

Problems (Errors: 3):

Errors:

```

"/results/b/results/Valentin_Varbanov_4_041472.rb:57:in `block in <main>': undefined local
variable or method `name' for main:Object (NameError) from
../results/b/results/Valentin_Varbanov_4_041472.rb:44:in `each' from
../results/b/results/Valentin_Varbanov_4_041472.rb:44:in `<main>' "
"Valentin_Varbanov_4_041472.rb:60:in `<main>': uninitialized constant CSV (NameError) "
"/usr/lib/ruby/1.9.1/csv.rb:1729:in `<<': undefined method `map' for "Lastname":String
(NoMethodError)
  from Valentin_Varbanov_4_041472.rb:63:in `block (2 levels) in <main>'
  from Valentin_Varbanov_4_041472.rb:62:in `each'
  from Valentin_Varbanov_4_041472.rb:62:in `block in <main>'
  from /usr/lib/ruby/1.9.1/csv.rb:1354:in `open'
  from Valentin_Varbanov_4_041472.rb:61:in `<main>'
"

```

Solution: There is no quick solution.

Program rank: 1

Vladimir_Yordanov_9_4bbed0.rb

program code:

```
names = Hash.new
Dir.glob (ARGV[0] + "*.rb") do |file|
  if (ARGV[1] == true)
    ARGV[1] == false
  end

  slice = file.split("/").last
  first_name = slice.split('_')[0]
  second_name = slice.split('_')[1]
  if (second_name.length == 5)
    #print first_name
    #puts second_name
    names[first_name] = second_name
  end
end

end

names = names.sort
```

puts names

```
require 'csv'
CSV.open("results.csv", "w") do |csv|
  names.to_a.each do |element|
    csv << element
  end
end
```

Problems (Errors: 1):

Common error:

```
"Vladimir_Yordanov_9_4bbed0.rb:26:in `block in <main>': undefined method `length' for
nil:NilClass (NoMethodError) from ../results/b/results/Vladimir_Yordanov_9_4bbed0.rb:18:in
`glob' from ../results/b/results/Vladimir_Yordanov_9_4bbed0.rb:18:in `<main>' "
```

Common problem: "No such file or directory - result.csv. No result.csv file was found."

Solution:

Common error:

```
if (second_name.length == 5) => if second_name != nil and second_name.length
== 5
```

Common problem:

```
"results.csv" => "result.csv"
```

Dir.glob problem:

```
"*.rb" => "/*.*rb"
```

Sort problem:

```
names = names.sort_by{|key, val| [val, key]}
```

Program rank: 4

Veselina_Kolova_8_65630e.rb

program code:

```
require 'csv'
```

```
people = Hash.new
```

```
Dir.glob("#{ARGV[0]}/**/*.*").each do |text_file|
```

```
  if File.extname(text_file) text_file.include?(".rb") &&
text_file.split(/_/).last.split(/\.\/).first.to_i.is_a Integer then
    if (text_file.split("/").last.split("_").length == 3) then
      text_file = text_file.split("/").last
      if (text_file.split("_")[1].length == 5) then
        people[text_file.split("_")[1]] = text_file.split("_")[0]
      end
    end
  end
end
```

```
people = Hash[people.sort_by{|k,v| k}.reverse]
```

```

CSV.open("result.csv","w") do |csv|
  people.each do |element|
    csv << element
  end
end

```

Problems (Errors: 1):

Error:

```

"Veselina_Kolova_8_65630e.rb:23: syntax error, unexpected tIDENTIFIER, expecting
keyword_then or ';' or '\n' ...le.exname(text_file) text_file.include?(".rb") && text_file... ... ^
../results/b/results/Veselina_Kolova_8_65630e.rb:23: syntax error, unexpected tCONSTANT,
expecting keyword_end ...t(/\.\/).first.to_i.is_a Integer then ... ^
../results/b/results/Veselina_Kolova_8_65630e.rb:31: syntax error, unexpected keyword_end,
expecting $end "

```

Solution:

```

      if File.exname(text_file) text_file.include?(".rb") &&
text_file.split(/_\/).last.split(/\.\/).first.to_i.is_a Integer then ==>
if File.exname(text_file) && text_file.include?(".rb") && text_file.split(/_\/).last.split(/\.\/).first.to_i.to_s
== text_file.split(/_\/).last.split(/\.\/).first then

```

Program rank: 4

Denis_Trenchev_4_b4c3f5.rb

program 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

```



```

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] #ZIBIT NOT APPROVED there is no arr
      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
end

sort = arr3.sort_by{|asd| asd[1]}
CSV.open("students.csv", "w") do |csv| #WTF IS STUDENTS.CSV (result) NIGGA
  sort.each do |element|
    csv << element
  end
end
Problems (Errors: 1):
Error:
"Denis_Trenchev_4_b4c3f5.rb:31:in `block in <main>': undefined local variable or method
`arr' for main:Object (NameError) from ../results/a/results/Denis_Trenchev_4_b4c3f5.rb:25:in
`glob' from ../results/a/results/Denis_Trenchev_4_b4c3f5.rb:25:in `<main>'"
```

Solution:

```

arr[i][0] = name[0] => arr1[i][0] = name[0]
arr[i][1] = name[1] => arr1[i][1] = name[1]

arr[i][0] = name_1[0] => arr2[i][0] = name[0]
arr[i][1] = name_1[1] => arr2[i][1] = name[1]

students.csv => result.csv
```

Program rank: 3

Marian_Belchev_17_ad26e0.rb

program 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
```

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
end
```

Problems (Errors: 0):

Solution:

```
"#{ARGV[0]}*_*_*.rb" => "#{ARGV[0]}/*_*_*.rb"
"#{ARGV[1]}*_*_*.rb" => "#{ARGV[1]}/*_*_*.rb"
"results.csv" => "result.csv"
```

Suggestions:

use hash.compare_by_identity so duplicate keys won't be discarded

Program rank: 4

Ivelin_Slavchev_10_835552.rb

program 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
```

```

        end
    end
    result.sort_by{|k, v| v}
    CSV.open("result.csv", "w") do |csv|
        result.each do |p|
            csv << p
        end
    end
end

```

Problems (Errors: 1):

Error:

```
(digit2.to_i.to_s != digit) => (digit2.to_i.to_s != digit2)
```

Sort problem:

```
result.sort_by{|k, v| v} => result.sort_by{|k, v| [v, k]}
```

Program rank: 4

Stanislav_Valkanov_25_4482c1.rb

program 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

```

Problems (Errors: 1):

```
"Stanislav_Valkanov_25_4482c1.rb:23:in `block in <main>': undefined method `length' for nil:NilClass (NoMethodError) from ./results/a/results/Stanislav_Valkanov_25_4482c1.rb:18:in `glob' from ./results/a/results/Stanislav_Valkanov_25_4482c1.rb:18:in `<main>' "
```

Soluton:

```
if (last.length == 5)&&(short_name.split("_").size == 3) =>
  if last != nil && (last.length == 5)&&(short_name.split("_").size == 3)
```

Program rank: 4

Lubomir_Yankov_16_650c0b.rb

program 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
end
```

```

        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

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

    array.each do |element|

        csv << element

    end

end
end

```

Problems (Errors: 0):

Common error: "results.csv" => "result.csv"

file_name = file.split("/").last.split("") => file_name = file.split("/").last

file_name.each => file_name.each_char

Program rank: 4

Petko_Bozhinov_21_954dc6.rb

program 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
        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
end

```

Problems (Errors: 0)

Sort problem:

```

output = output.sort_by{ |element| element[1]} =>
output = output.sort_by{ |k, v| [v, k]}

```

Suggestions:

use hash.compare_by_identity so duplicate keys won't be discarded

Program rank: 5

Nikola_Marinov_20_add57e.rb

program 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

```

```
end
end
```

```
Dir.glob(ARGV[0] + "**/*.rb").each do |file|
```

```
  full_name=file.split("/").last
  name = file.split("/").last.split(".").first_split("_")
```

```
  if name.length != 3 && !is_numeric(name[2])
    array(count) = []
    array(count)[0]=full_name
    array(count)[1]= full_name.to_s.length
    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
```

Problems (Errors: 17):

Solution:

There is no quick solution.

Suggestions:

require 'ruby syntax'

Program rank: 1

Georgi_Ivanov_3_871529.rb:

program 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
end

```

```

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

```

Problems (Errors: 0):
 Programs is working...

Program rank: 5

Radoslav_Kostadinov_22_772118.rb

program 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
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

```

Problems (Errors: 0):
 Problem: wrong file checking algorithm

Solution:

Before:

```

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

```

After PIMPING:

```

CSV.open("result.csv", "w") do |csv|
  file1.sort.each do |first_name, last_name|
    flag = 0
    file2.sort.each do |first_name1, last_name1|
      if first_name1 == first_name and last_name1 == last_name
        flag = 1
      end
    end
  end
end
end

```

```

        break
      end
    end
  if flag == 0
    csv << [first_name, last_name]
  end
end
end
end

```

Program rank: 3

Simeon_Shopkin_23_56a835.rb

program 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
end
end

```

```

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

```

Problems (Errors: 1):

Problem: Program doesn't match with the task's condition!

Solution: There is no quick solution.

Program rank: 1

Ivo_Valchev_11_6c8bd9.rb

program 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

```

```

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
end

```

Programs (Errors: 0):

Problem with if conditions!

Solution:

```

!isNum!=nil hash_fold1.include?(name[0]) => "
!isNum!=nil and!hash_fold2.include?(name[0]) => "

```

Suggestions:

Do not use "!" for no reason!

Program rank: 2

Kristina_Pironkova_15_890ba0.rb

program 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

end

```

```

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

```

```

        csv << [last,first]

    end
end

```

Problems (Errors: 0):

Common problem: "results.csv" => "result.csv"

Sorting problem:

```
results.sort.each => results.sort_by{ |key, val| val}.reverse.each
```

Program rank: 5

Hristo_Dachev_29_4a196f.rb

program 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
end

```

Problems (Erros: 0):

Program is working correctly!

Program rank: 5

Dimitar_Nestorov_7_0d5526.rb

program 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 { |e| -e[1]}
CSV.open("result.csv", "w") do |csv|

    array.uniq.each do |e|

        csv << e
    end
end
end

```

Problems (Errors: 1):

Error:

```

"Dimitar_Nestorov_7_0d5526.rb:42:in `block in <main>': undefined method `-@' for "
L10letter2":String (NoMethodError) from ./results/a/results/Dimitar_Nestorov_7_0d5526.rb:42:in
`each' from ./results/a/results/Dimitar_Nestorov_7_0d5526.rb:42:in `sort_by' from
./results/a/results/Dimitar_Nestorov_7_0d5526.rb:42:in `<main>'

```

Solution:

```

" #{name[1].to_s}" => " #{name[1].to_s}"
array = array.sort_by { |e| -e[1]} => array = array.sort_by { |e| e[1]} .reverse

```

Program rank: 4

Tihomir_Lidanski_27_dafd44.rb

program 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
```

Problems (Errors: undefined):

Problem: unfinished program!

Solution: There is no quick solution!

Program rank: 1

Kalin_Marinov_14_12_bce70c.rb

program 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
```

Problems (Errors: 2):

Error:

```
"Kalin_Marinov_14_12_bce70c.rb:31: syntax error, unexpected keyword_do_block
Kalin_Marinov_14_12_bce70c.rb:34: syntax error, unexpected keyword_end, expecting $end
"
```

Common error:

```
"Kalin_Marinov_14_12_bce70c.rb:24:in `block in <main>': undefined method `length' for
nil:NilClass (NoMethodError)
  from Kalin_Marinov_14_12_bce70c.rb:21:in `glob'
  from Kalin_Marinov_14_12_bce70c.rb:21:in `<main>' "
```

Solution:

```
if short_name.length == 5 => if short_name != nil and short_name.length == 5
  hash.each |key| do => hash.each do |key|
    hash[name] = short_name => hash[name.split('_').first] = short_name
```

Program rank: 3

Stanislav_Gospodinov_26_b36abb.rb

program 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
```

end

Problems (Errors: 0):

Common problem: No such file or directory - result.csv. No result.csv file was found.

Solution:

```
"results.csv" => "result.csv"
```

Program rank: 5

Borislav_Rusinov_2_6fb3ad.rb

program 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
```

Problems (Errors: 0):

Common problem: No such file or directory - result.csv. No result.csv file was found.

Solution:

```
"results.csv" => "result.csv"
```

Program rank: 5

Moretti_Georgiev_19_b7f153.rb

program 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
```

Problems (Errors: 0):

Program is working correctly!

Program rank: 5

Momchil_Angelov_18_d8aa65.rb

program 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
end
Dir.glob(b + "/*.rb") do |my_text_file2|
    short2= my_text_file2.split('/').last
    length2 = short2.length
    shorter2= short.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 { |el|
    el[1]
}

```

```

    CSV.open("result.csv", "w") do |csv|
arr3.each do |element|
    csv << element
end

```

end

Problems (Errors: 1):

Errors:

```

"Momchil_Angelov_18_d8aa65.rb:44:in `block in <main>': undefined local variable or
method `short' for main:Object (NameError)
    from Momchil_Angelov_18_d8aa65.rb:40:in `glob'
    from Momchil_Angelov_18_d8aa65.rb:40:in `<main>' "

```

Solution (only for the error):

```

    shorter2= short.split('.').first.split('_') => shorter2= short2.split('.').first.split('_')

```

Solution: There is no quick solution.

Program rank: 2

Kamena_Dacheva_13_0af18f.rb

program 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}/*.rb") 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
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

```

Problems (Errors: 0):
 Program is working correctly!

Program rank: 5

Stanimir_Bogdanov_24_ca514d.rb

program 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

```


end

Problems (Errors: 0):

Program is working correctly!

Program rank: 5

Veselin_Dechev_11A2_5f1c22.rb

program 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
```

Problems (Errors: undefined):

Unknown task's condition!

Program rank: 1

Dimitar_Terziev_6_88db52.rb

program code:

```
require 'csv'
arr = []
Dir.glob("# {ARGV[0]}*.rb*"){ |file|
  file_str = file.split('/').last
  if(file_str =~ /^[a-zA-Z]+\_[a-zA-Z]+\_\d+\.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(' ')
  }
}
```

Problems (Errors: 0):

Sort problem: Results should be sort by Lastname ASC.

Solution:

```
arr.uniq.sort.each => arr.uniq.sort_by{|key, val| [val, key]}.each
```

Program rank: 5