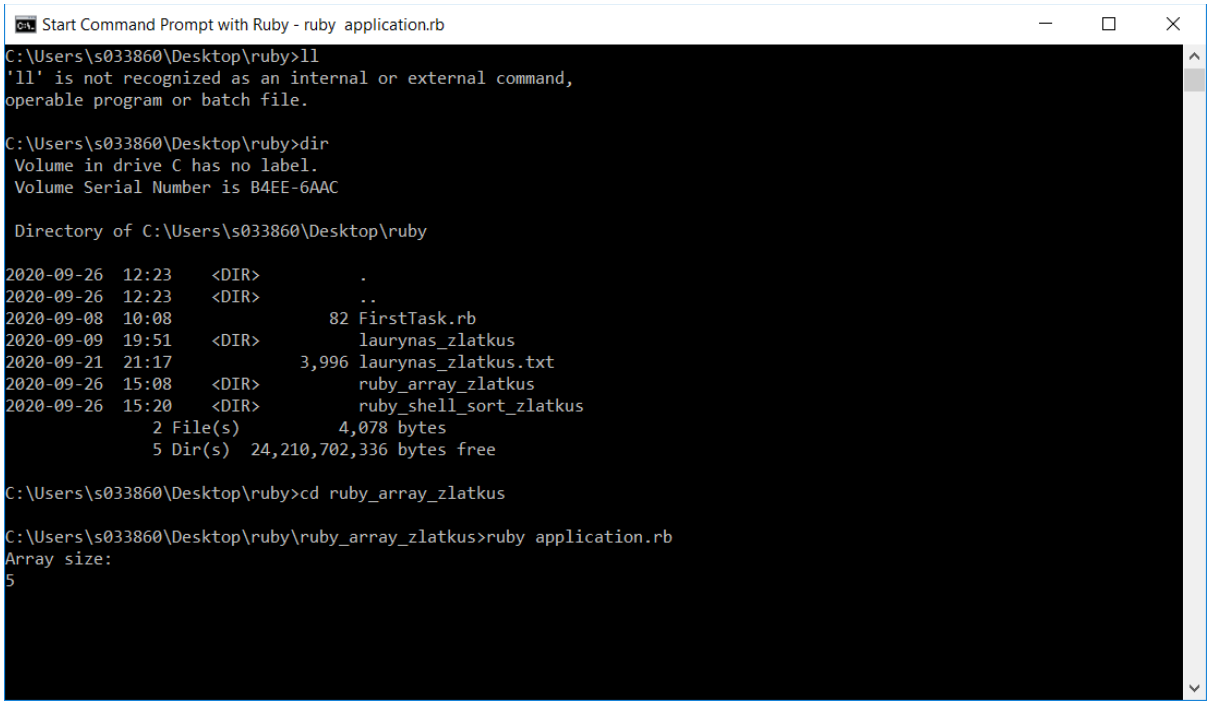
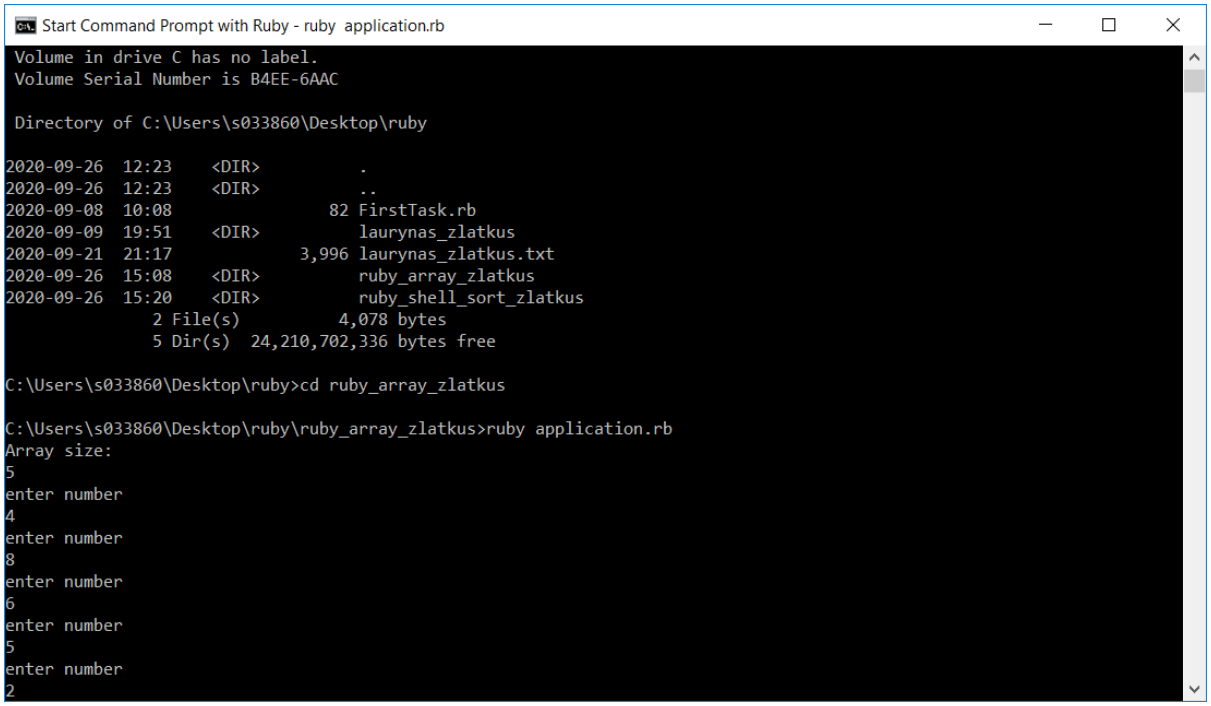
Min and Max finder in Array

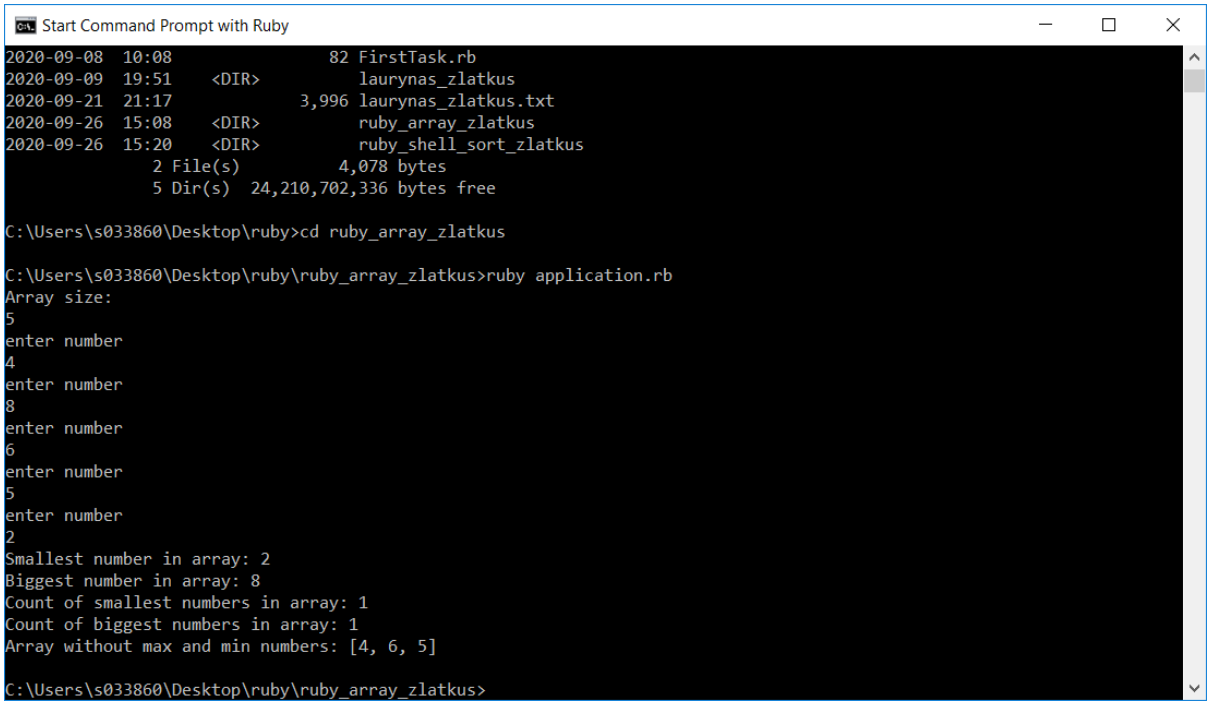
Per *Start command Prompt with Ruby* paleidus komandą *ruby application.rb* yra klausiama masyvo dydis



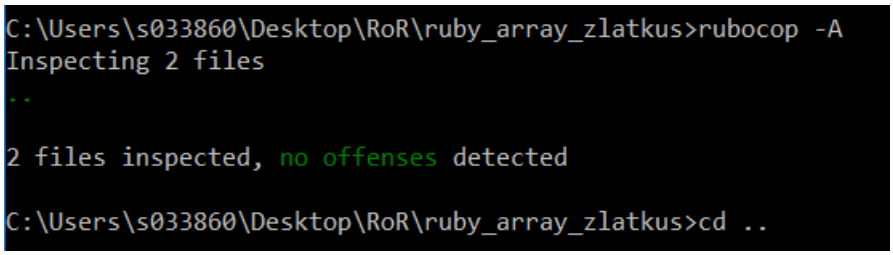
Toliau bus prašoma įvesti masyvo narius



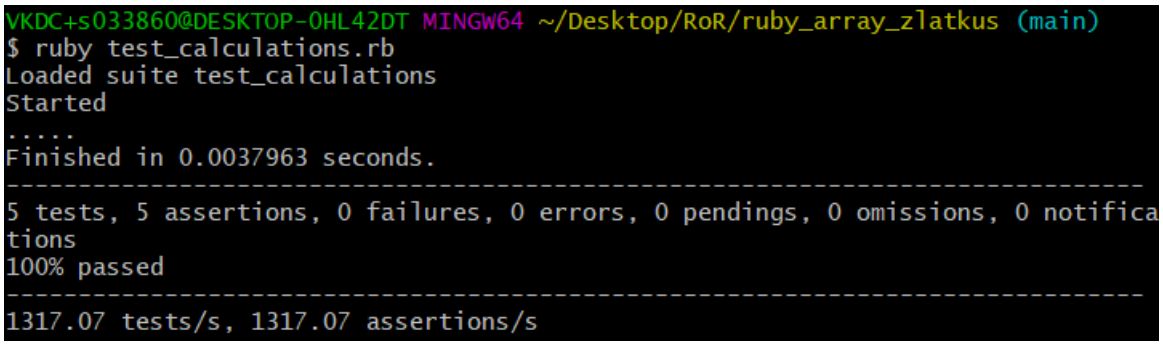
Suvedus visus narius yra apskaičiuojama masyvo min, max, min kiekis, max kiekis ir išvedamas masyvas be min ir be max skaitmenų.



Rubocop patirikinimas

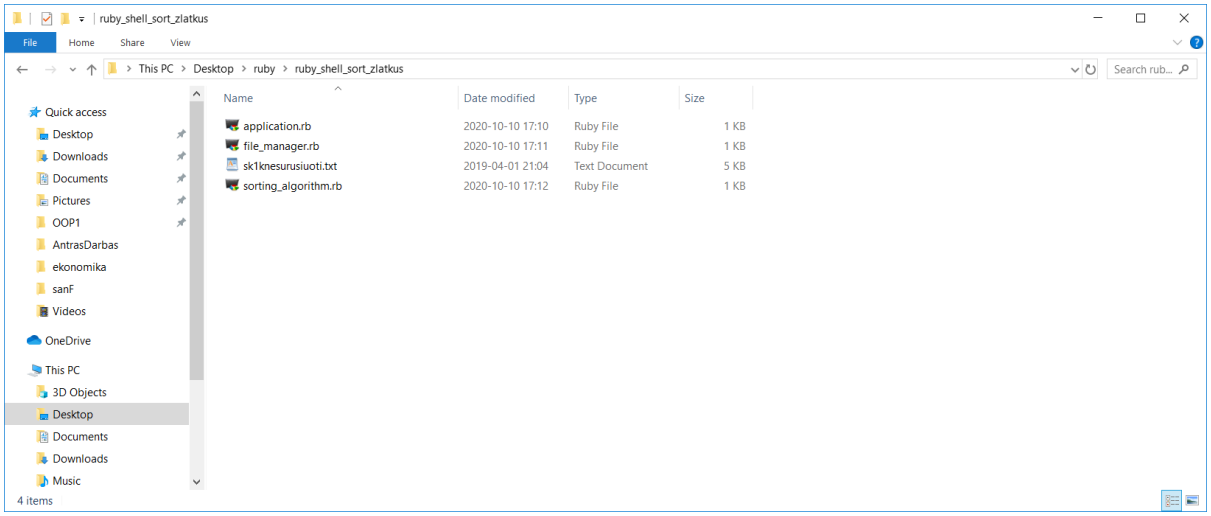


Unit testai

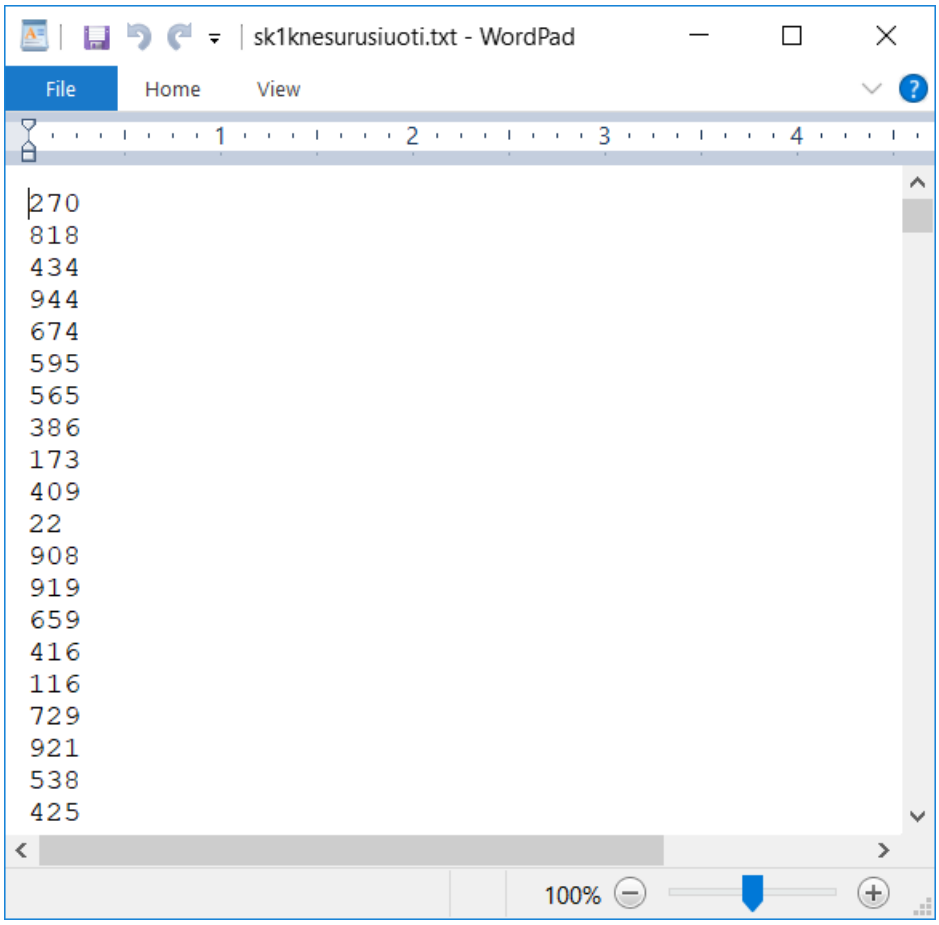


Shell Sort Algorithm

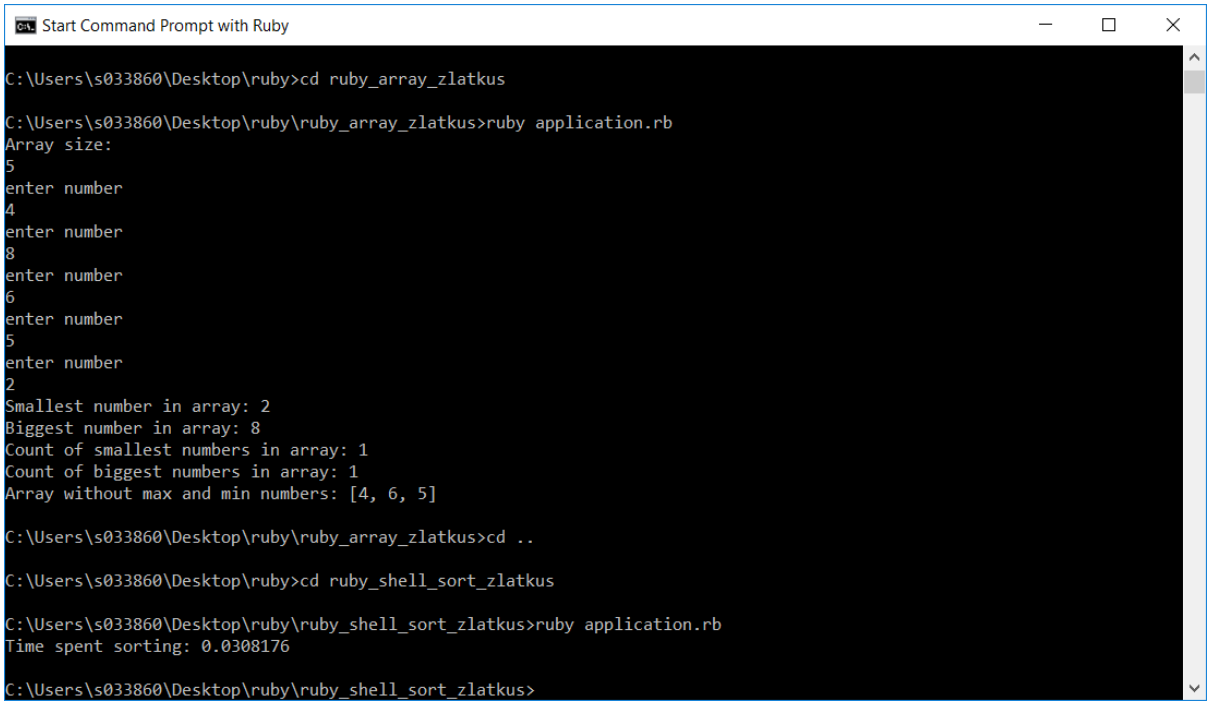
Svarbu jog darbiniame aplankale būtu .txt pavidalu paruoštas pavydžiui 1 tūkstančio skaitmenų failas.



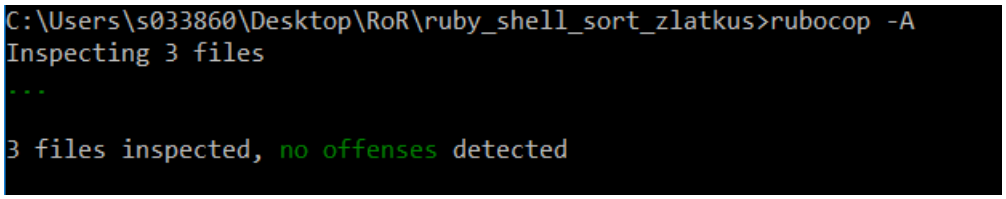
Kuriame visi skaitmenys yra nesurūšiuoti.



Paleidus programą per *Start command Prompt with Ruby* su komandą *ruby application.rb* yra išvedamas laikas per kiek buvo surikiuoti visi faile esantys skaitmenys.

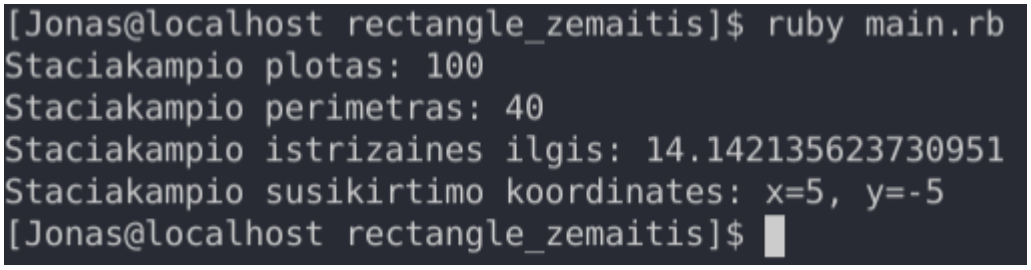


Rubocop patikrinimas



Rectangle Claculator

Programa apskaičiuoja duomenis apie stačiakampį. Paleidę programą komanda “ruby main.rb” matome žemiau esančius rezultatus.



Mūsų pradiniai stačiakampio duomenys:

```
rect = Rectangle.new(10, 10, 0, 0)
```

,tai yra (ilgis, plotis, x, y [koordinates]).

Rubocop:

```
C:\Users\s033860\Desktop\RoR\rectangle_zemaitis>rubocop -a
Inspecting 3 files
...

3 files inspected, no offenses detected
```

Unit testai

```
VKDC+s033860@DESKTOP-0HL42DT MINGW64 ~/Desktop/RoR/rectangle_zemaitis (main)
$ ruby test_rectangle.rb
Loaded suite test_rectangle
Started
....
Finished in 0.0011899 seconds.
------------------------------------------------------------------------------
4 tests, 4 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifica
tions
100% passed
------------------------------------------------------------------------------
3361.63 tests/s, 3361.63 assertions/s
```

Ceasar Chiper

Programa naudodama cezario šifravimo būdą užšifruoja duotą tekstą, šiuo atveju: `'Labas 99 67'` ir gražina užšifruotą tekstą, bei dešifruotą užšifruotą tekstą. Ši programa taip pat yra paleidžiama ”ruby main.rb” komanda ir apačioje matome gautą rezultatą.

```
[Jonas@localhost ceasar_chiper_zemaitis]$ ruby main.rb
Encrypted: Qfgfx%>>%;<
Decrypted: Labas 99 67
[Jonas@localhost ceasar_chiper_zemaitis]$
```

P.S naudojama raktas yra skaičius 5:

```ruby
chiper = CeasarChiper.new(5)
```

Rubocop:

```
C:\Users\s033860\Desktop\RoR\ceasar_chiper_zemaitis>rubocop -A
Inspecting 2 files
..

2 files inspected, no offenses detected
```

Polindrome finder

Paleidus programą, vartotojo klausiama kiek polindromų jis nori rasti.

```
rytisrazmus@rytiss-MacBook-Pro trecias % ruby polindromes/main.rb
Enter the count of polindromes you want to find
10
```

Gaunamas atsakymas.

```
rytisrazmus@rytiss-MacBook-Pro trecias % ruby polindromes/main.rb
Enter the count of polindromes you want to find
10
11,22,33,44,55,66,77,88,99,101,
rytisrazmus@rytiss-MacBook-Pro trecias %
```

Rubocop patikra.

```
rytisrazmus@rytiss-MacBook-Pro polindromes_razmus % rubocop
Inspecting 2 files
..

2 files inspected, no offenses detected
```

Unit testai

```
VKDC+s033860@DESKTOP-0HL42DT MINGW64 ~/Desktop/RoR/polindromes_razmus/tests (mai
n)
$ ruby test_plindromes.rb
Loaded suite test_plindromes
Started
..
Finished in 0.0006396 seconds.
----------------------------------------------------------------------------
2 tests, 8 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifica
tions
100% passed
----------------------------------------------------------------------------
3126.95 tests/s, 12507.82 assertions/s
```

Matrix multiplier

Programa skaičiuoja dviejų matricų sandaugą. Abi matricos yra apibrėžtos programos kode.

```
rytisrazmus@rytiss-MacBook-Pro trecias % ruby matrixes/main.rb
Mtrica A:
   1   2   3
   4   5   6
   1   2   3

Matrica B:
   7   8   4
   9  10   3
  11  12   6

Rezultatas
  58  64  28
 139 154  67
  58  64  28
rytisrazmus@rytiss-MacBook-Pro trecias %
```

Rubocop patikra.

```
rytisrazmus@rytiss-MacBook-Pro matrix_multiplication_razmus % rubocop
Inspecting 3 files
...

3 files inspected, no offenses detected
```