

Porovnanie vzorovej a neefektívnej implementácii bolo vykonané na testovacej sade, ktorá pozostávala z 25-násobného zavolania metódy next. Neefektívna implementácia je výrazne pomalšia hlavne z dôvodu obrovského množstva rekurzívnych volaní a operácií s číslami (porovnanie, sčítovanie, odčítovanie). Meranie bolo vykonané pomocou ruby-prof, ktorého výstupy boli nasledovné:

#### fib-sequence.rb:

Measure Mode: wall\_time  
Thread ID: 16422900  
Fiber ID: 22736020  
Total: 0.001309  
Sort by: self\_time

%self	total	self	wait	child	calls	name
69.97	0.001	0.001	0.000	0.000	1	Kernel#require_relative
9.63	0.000	0.000	0.000	0.000	25	FibonacciSequence#next
4.86	0.000	0.000	0.000	0.000	4	IO#set_encoding
2.99	0.000	0.000	0.000	0.000	73	Array#[]
2.70	0.001	0.000	0.000	0.001	2	Global#[No method]
2.17	0.000	0.000	0.000	0.000	48	Fixnum#+
1.77	0.000	0.000	0.000	0.000	46	Fixnum#-
1.60	0.000	0.000	0.000	0.000	1	Integer#times
1.31	0.000	0.000	0.000	0.000	23	Array#[]=
0.84	0.000	0.000	0.000	0.000	23	NilClass#nil?
0.46	0.000	0.000	0.000	0.000	1	FibonacciSequence#initialize
0.40	0.000	0.000	0.000	0.000	6	Module#method_added
0.31	0.000	0.000	0.000	0.000	1	FibonacciSequence#reset
0.31	0.000	0.000	0.000	0.000	1	Class#inherited
0.29	0.000	0.000	0.000	0.000	1	Class#new
0.24	0.000	0.000	0.000	0.000	2	Kernel#nil?
0.16	0.000	0.000	0.000	0.000	1	<Class::BasicObject>#allocate

#### fib-sequence-slow.rb:

Measure Mode: wall\_time  
Thread ID: 17299440  
Fiber ID: 23620180  
Total: 1.329342  
Sort by: self\_time

%self	total	self	wait	child	calls	name
13.99	0.186	0.186	0.000	0.000	392784	Fixnum#-
12.41	0.165	0.165	0.000	0.000	392834	Fixnum#<
7.05	0.094	0.094	0.000	0.000	196417	Fixnum#+
6.93	0.092	0.092	0.000	0.000	196417	Array#[]
0.07	0.001	0.001	0.000	0.000	1	Kernel#require_relative
0.01	1.328	0.000	0.000	1.328	392809	*FibonacciSequenceSlow#fib
0.01	0.000	0.000	0.000	0.000	4	IO#set_encoding
0.00	1.328	0.000	0.000	1.328	25	FibonacciSequenceSlow#next
0.00	1.328	0.000	0.000	1.328	1	Integer#times
0.00	1.329	0.000	0.000	1.329	2	Global#[No method]
0.00	1.328	0.000	0.000	1.328	25	FibonacciSequenceSlow#current
0.00	0.000	0.000	0.000	0.000	1	FibonacciSequenceSlow#initialize
0.00	0.000	0.000	0.000	0.000	7	Module#method_added
0.00	0.000	0.000	0.000	0.000	1	Class#new
0.00	0.000	0.000	0.000	0.000	1	Class#inherited
0.00	0.000	0.000	0.000	0.000	1	Module#private
0.00	0.000	0.000	0.000	0.000	1	<Class::BasicObject>#allocate
0.00	0.000	0.000	0.000	0.000	1	FibonacciSequenceSlow#reset