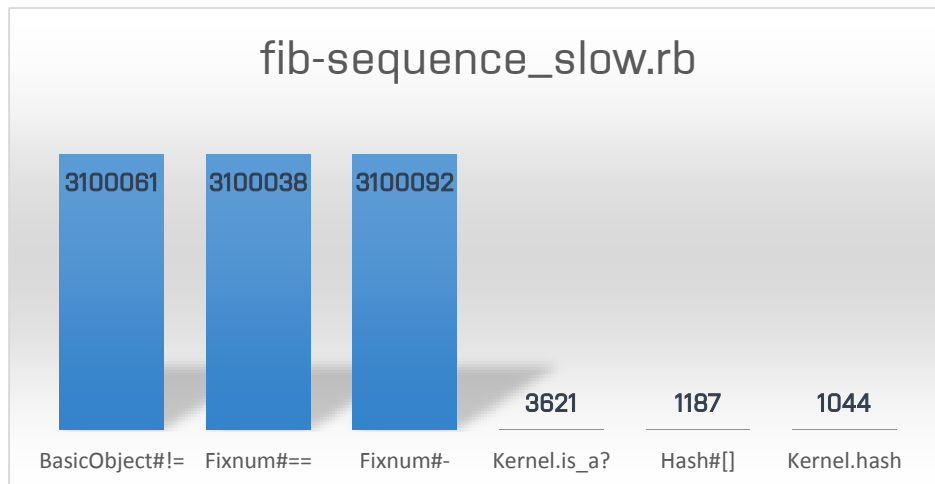


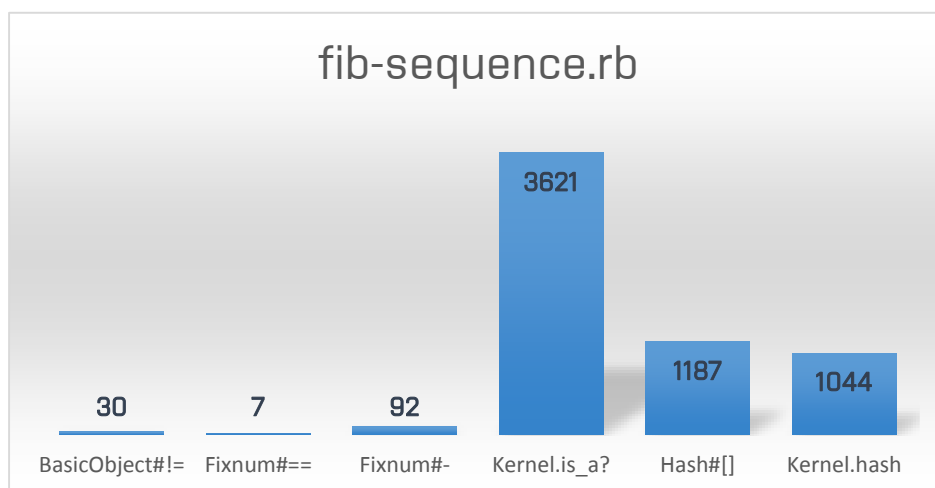
Profilování fibonacciho řady

Rozdíl mezi vzorovou a pomalou implementací fibonacciho řady byl v přidání cyklu while v metodě next.

```
21 | i = 100000
22 | while (i != 0)
23 |   i = i - 1
24 | end
```



Na následujícím grafu vidíme počty volání pomalé implementace fibonacciho řady pro nejčastější volání v pomalé implementaci fibonacciho řady.



Na následujícím grafu vidíme počty volání vzorové implementace fibonacciho řady pro nejčastější volání v pomalé implementaci fibonacciho řady.

Z porovnání grafů vyplývá, že neefektivita pomalé implementace fibonacciho řady je přesně tam, kde bychom ji čekali. Tedy ve voláních, které se uskutečňují díky cyklu while v metodě next (implementace zobrazena výše).

% time	cumulative seconds	self seconds	calls	self ms/call	total ms/call	name
49.25	35.55	35.55	31	1146.77	2318.06	FibonacciSequence#next
0.00	0.32	0.00	31	0.00	0.00	FibonacciSequence#next

pomalá implementace
vzorová implementace

Rozdíl v implementaci metody next jde vidět i na profilování samotných metod.