

Test con `--inspect`: Podemos ver que el `console.log` es el proceso que mas tarda

Perfiles		Gruesa (de abajo hacia arriba) ▼			
		Tiempo individual		Tiempo total	
				Función	
ERFILES DE CPU		48627.8 ms		48627.8 ms	
		369.3 ms 8.16 %		504.6 ms 11.15 %	
▼consoleCall					
Perfil 1		369.3 ms 8.16 %		504.6 ms 11.15 %	
		207.1 ms 4.58 %		207.1 ms 4.58 %	
		184.3 ms 4.07 %		184.3 ms 4.07 %	
				writeUtf8String	

Autocannon y 0x : con `console.log`

```
PS C:\Users\aye19\Desktop\Backend-32070\Desafio Clase32> npm test
> clase-25@1.0.0 test
> node benchmark.js

Running all benchmarks in parallel...
Running 20s test @ http://localhost:8080/info
100 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	858 ms	1002 ms	1831 ms	1984 ms	1060.38 ms	237.23 ms	2996 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	23	23	95	111	91.65	16.24	23
Bytes/Sec	46.9 kB	46.9 kB	194 kB	226 kB	187 kB	33.1 kB	46.9 kB

Req/Bytes counts sampled once per second.
of samples: 20
2k requests in 20.17s, 3.74 MB read

Autocannon y 0x : sin `console.log`

```
PS C:\Users\aye19\Desktop\Backend-32070\Desafio Clase32> npm test
> clase-25@1.0.0 test
> node benchmark.js

Running all benchmarks in parallel...
Running 20s test @ http://localhost:8080/info
100 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	842 ms	1004 ms	1864 ms	1988 ms	1053.05 ms	228.93 ms	2951 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	21	21	95	116	92.15	17.03	21
Bytes/Sec	42.8 kB	42.8 kB	194 kB	237 kB	188 kB	34.7 kB	42.8 kB

Req/Bytes counts sampled once per second.
of samples: 20
2k requests in 20.15s, 3.76 MB read
PS C:\Users\aye19\Desktop\Backend-32070\Desafio Clase32>