

## Sifting benchmark on jjs-bdd0.1alpha1

One of our major goals for this this release was to close the lack of performance using our sifting algorithm. This benchmark compares runtimes of the elder jjs-bdd0.01 to the new release jjs-bdd0.1alpha1. Like in our last benchmark, the SUM function was created for several input sizes  $n$  with an exponential variable order  $\pi = (x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n)$ . The following table presents the runtimes in seconds and the number of nodes before and after sifting.

### Results:

n	old time(s)	new time(s)	nodes before sifting	nodes after sifting
8	0.58	0.14	1253	753
9	0.97	0.26	2530	1519
10	3.41	0.50	5087	3053
11	14.65	1.05	10204	6123
12	61.43	2.33	20441	12265
13	252.49	5.24	40918	24551
14	1049.43	11.74	81875	49125
15	4313.00	26.33	163792	98275
16	-	59.13	327629	196577
17	-	135.76	655306	393183
18	-	304.27	1310663	786997
19	-	768.48	2621380	1572827

So we improved the runtime of the new release not only by a constant factor!