Benchmark Results

all times in µs		**C++98 gcc -O2**		**FP 3.2.2 -O4**		**FP 3.2.2 -O3**		**FP 3.2.2 -O2**		**Node.js 12.16**		**LuaJIT 2.0 Lua 5.1**		**OBXMC 21-09 Mono3**		**OBXMC 21-08 Mono5**		**ObxIDE 0.9.38 C gen, with GC**	
Benchmark:	*n*	*average*	*factor*	*average*	*factor*	*average*	*factor*	*average*	*factor*	*average*	*factor*	*average*	*factor*	*average*	*factor*	*average*	*factor*	*average*	*factor*
...																			
Bounce	1500/1	51	0.20	74	0.3	74	0.3	82	0.3	119	0.5	249	1.0	116	0.5	126	0.5	68	0.3
List	1500/1	78	0.12	165	0.2	165	0.2	165	0.2	208	0.3	676	1.0	199	0.3	222	0.3	90	0.1
Mandelbrot	500/1	1	0.50	1	0.5	2	1.0	1	0.5	12	6.0	2	1.0	2	1.0	11	5.5	1	0.5
NBody	250000/1	1	0.13	2	0.3	3	0.4	3	0.4	3	0.4	8	1.0	4	0.5	9	1.1	3	0.4
Permute	1000/1	98	0.30	133	0.4	130	0.4	132	0.4	168	0.5	328	1.0	220	0.7	272	0.8	132	0.4
Queens	1000/1	96	0.32	118	0.4	126	0.4	125	0.4	231	0.8	297	1.0	210	0.7	228	0.8	148	0.5
Sieve	3000/1	31	0.26	34	0.3	34	0.3	33	0.3	103	0.9	119	1.0	84	0.7	99	0.8	31	0.3
Storage	1000/1	756	0.34	4'373	2.0	4'408	2.0	4'538	2.1	310	0.1	2'202	1.0	337	0.2	384	0.2	533	0.2
Towers	600/1	159	0.53	271	0.9	268	0.9	274	0.9	307	1.0	299	1.0	500	1.7	482	1.6	260	0.9
sum of averages:		1'271		5'171		5'210		5'353		1'461		4'180		1'672	0.40	1'833	0.44	1'266	0.30
geomean of factors:			0.27		0.45		0.51		0.48		0.63		1.0		0.57		0.81		0.35
1/geomean:			3.75		2.22		1.96		2.09		1.58		1.00		1.77		1.23		2.87

Benchmarks used from https://github.com/smarr/are-we-fast-yet commit 770c664 3.4.2020 Mono 3.12.1 Mono 5.20.1.34 gcc 4.8.2 -O2
and https://github.com/rochus-keller/Are-we-fast-yet Boem GC 7.2d
Testmachine: HP EliteBook 2530p, Intel Core Duo L9400 1.86GHz, 4GB RAM, Linux i386
All binaries compiled with GCC 4.8.2 and FPC 3.2.2
LuaJIT params, deviations from default values:

maxtrace	100000
maxrecord	40000
maxside	100
maxsnap	1000
sizemcode	64
maxmcode	5120

NOTE that the FP compiler per default
does no range or overflow checking;
these have explicitly to be enabled
by the -Cr and -Co options.
The results here are without these options.

NOTE that this report only includes the Are-we-fast-yet microbenchmarks so far; the macrobenchmarks are work in progress.