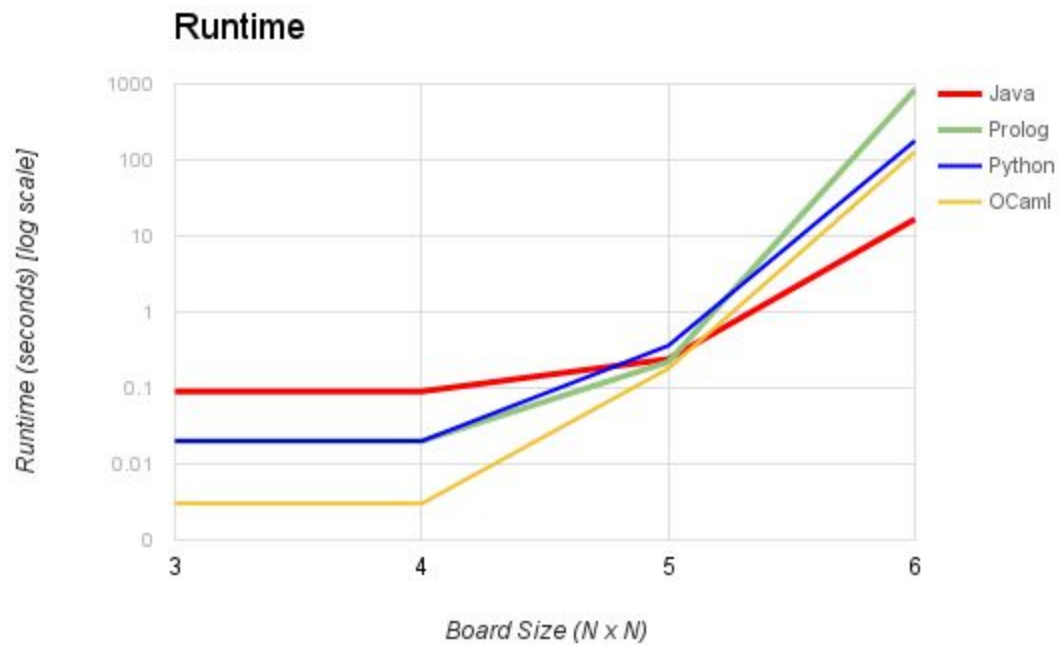# Performance

Runtime

Runtime is measured via the "user"+"sys" time output by the unix time tool (seconds)

Language	Runtime of N=3	Runtime N=4	Runtime N=5	Runtime N=6
Java	0.09	0.09	0.24	16.66
Prolog	0.03	0.02	0.22	844.79
Python	0.02	0.02	0.66	179.05
OCaml	0.0	0.0	0.18	128+ (stack overflow)

Memory Usage

Memory usage is measured via "maximum resident set size" output by the unix time tool (bytes)

Language	Mem usage of N=3	Mem usage of N=4	Mem usage of N=5	Mem usage of N=6
Java	27299840	27406336	75755520	2103099392
Prolog	4050944	4059136	4083712	4255744
Python	6905856	7135232	27222016	3529547776
OCaml	1101824	1191936	7950336	105431040+ (stack overflow)
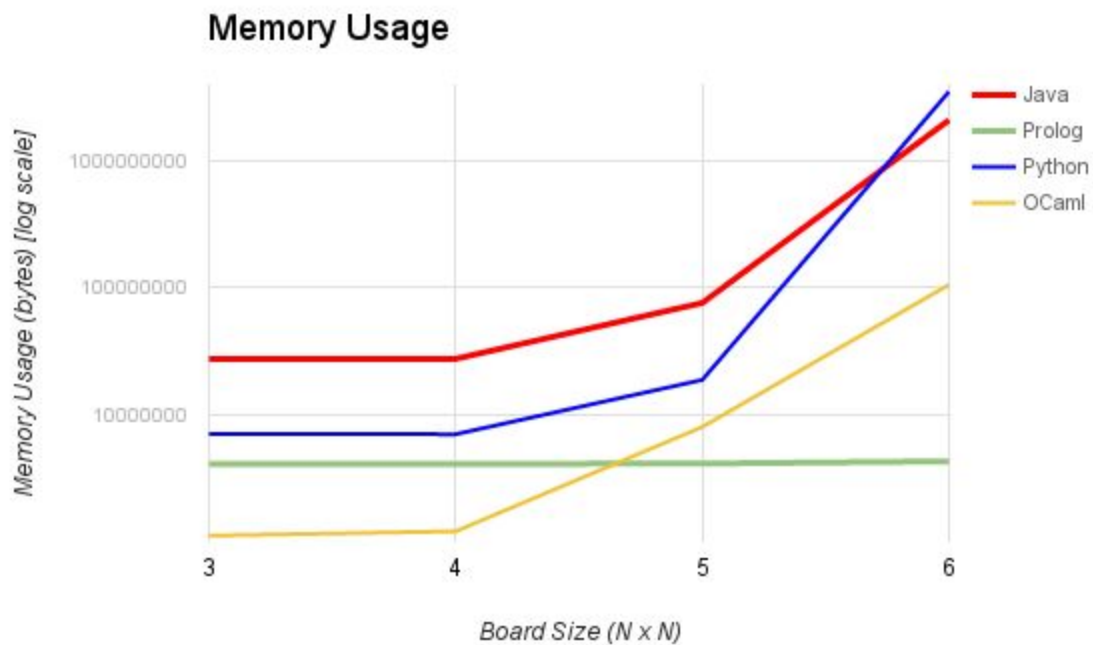
Development

Language	Time(No Code → Basic Impl)	Time(Basic Impl → Robust Solution)
Java	3 hours	3 hours
Prolog	10 hours	1 hour
Python	4 hours	1 hour
OCaml	8 hours	2 hours

Language	Time (fighting static checker)	Time(fighting runtime errors)	Grappling with libraries
Java	3 hours	2 hour	1 hour
Prolog	2 hours	6 hours	3 hours
Python	0 hours	2 hour	3 hours
OCaml	4 hours	2 hours	4 hours

Language	Lines of code (approx)	Notes
Java	230	Lots of boilerplate
Prolog	130	Lots of basic library functions (making lists, searching…) The actual logic was easy once all that was out of the way.
Python	130	
OCaml	120	

Language	Difficulty of increasing time efficiency (0[easy]-10[hard])
Java	2 (just have to change some smaller functions)
Prolog	9 (would have to restructure most of my logic)
Python	2 (just have to change some smaller functions)
OCaml	5 (only slight restructuring)