

Practical 2 - Analysis of Russian Peasant Algorithm

Results of Empirical Analysis

Number A	No.digits	Number B	No.digits	Result	Time Elapsed
1	1	10	2	10	0.001
100	3	1000	4	100000	0.001
10000	5	100000	6	100000000 0	0.001
1000000	7	10000000	8	100000000 00000	0.002
100000000	9	1000000000	10	100000000 000000000	0.002
10000000000	11	100000000000	12	100000000 000000000 0000	0.001
1000000000000	13	10000000000000	14	100000000 000000000 00000000	0.001
1000000000000000	15	100000000000000 000	16	100000000 000000000 000000000 000	0.002
10000000000000000 10000000000000000	30	100000000000000 00010000000000 000000	32	100000000 000000110 000000000 000010000 000000000 000000000 0000000	0.003
10000000000000000 10000000000000000 10000000000000000 10000000000000000	60	100000000000000 00010000000000 0000001000000 0000000001000 000000000000	64	100000000 000000110 000000000 000111000 000000000 111100000 000000011 100000000 000001100 000000000 000100000 000000000 000000000	0.03

				110111111 111111111 111111111 111111111 111111111 111101111 111111111 100111111 111111100 011111111 111100001 111111111 100000111 111111100 000011111 111100000 001111111 100000000 111111100 000000011 111100000 000001111 100000000 000111100 000000000 011100000 000000001 100000000 000000100 000000000 000000000 000000000	
--	--	--	--	--	--

Input size versus Time Graph

From looking at how the algorithm performs relative to size as illustrated in the graph below, I believe the time complexity of the Russian Peasant algorithm to be $O(\log n)$.

Russian Peasant

