CS 342 – Operating Systems

Project 1

Kaan Sancak

21502708

Section-1

Report of Part C: Experiments

Experiment A: prime

Tables of prime

Note: Run-times are measured as second.

N = 1000

Trial No/ No of Children	10	20	30	40	50
Trial 1	0.21	0.192	0.173	0.184	0.183
Trial 2	0.192	0.181	0.185	0.178	0.179
Trial 3	0.201	0.19	0.186	0.18	0.18
Average	0.201	0.187667	0.181333	0.180667	0.180667

N = 10000

Trial No/ No of Children	10	20	30	40	50
Trial 1	7.741	6.608	6.318	5.749	5.307
Trial 2	7.766	6.969	6.081	5.738	6.176
Trial 3	7.472	6.886	6.176	5.637	5.611
Average	7.659667	6.821	6.191667	5.708	5.698

N = 25 000

Trial No/ No of Children	10	20	30	40	50
Trial 1	28.999	24.431	22.088	20.822	20.241
Trial 2	28.332	24.118	23.354	21.194	20.656
Trial 3	28.866	24.579	22.535	20.124	21.181
Average	28.73233	24.376	22.659	20.71333	20.69267

N = 50 000

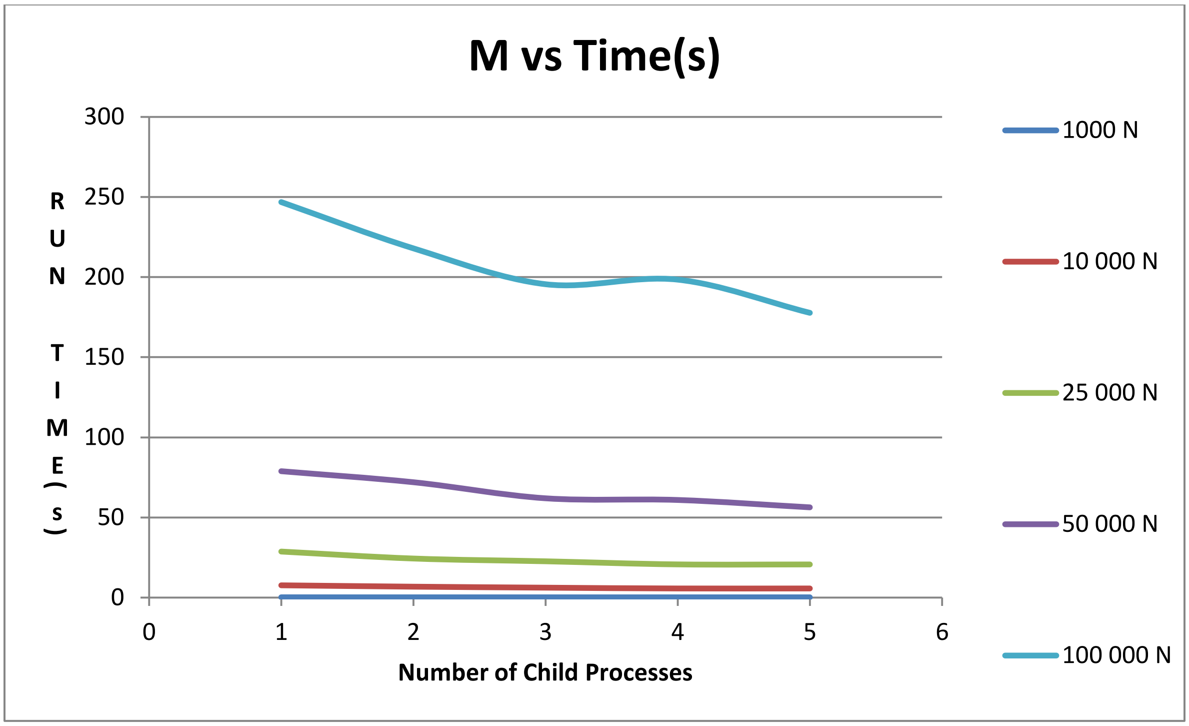
Trial No/ No of Children	10	20	30	40	50
Trial 1	79.003	80.331	63.038	57.795	54.067
Trial 2	78.576	68.137	61.975	59.306	57.665
Trial 3	79.028	67.514	60.902	65.653	57.233
Average	78.869	71.994	61.97167	60.918	56.32167

N = 100 000

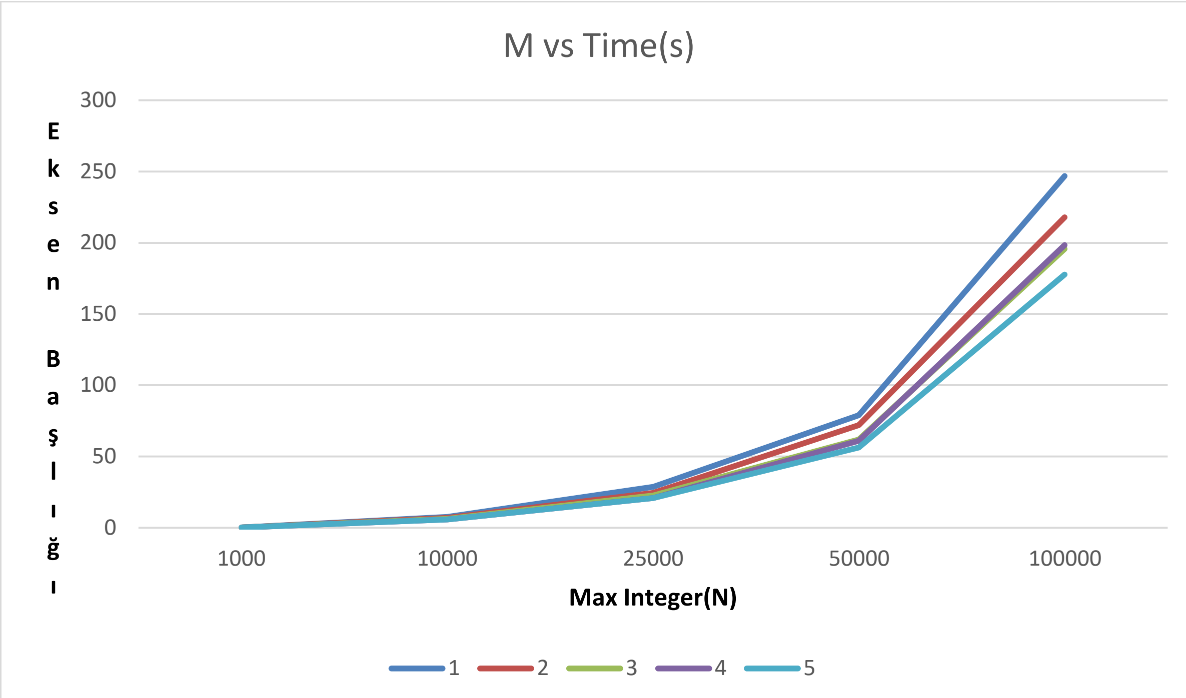
Trial No/ No of Children	10	20	30	40	50
Trial 1	247.413	196.405	197.864	236.99	175.962
Trial 2	238.718	221.187	190.453	177.573	178.709
Trial 3	254	236.161	198.334	180.634	178.429
Average	246.7913	217.9177	195.5503	198.399	177.7

Graphs of Prime

Number of Children(M) vs Run Time(s)



Max Integer(N) vs Run Time(s)



Experiment B: mqprime

Tables of mqprime

Note: Run-times are measured as second.

N = 1000

Trial No/ No of Children	1 Children	2 Children	3 Children	4 Children	5 Children
Trial 1	0.356	0.264	0.2235	0.227	0.212
Trial 2	0.355	0.269	0.234	0.216	0.262
Trial 3	0.349	0.262	0.249	0.222	0.215
Average	0.353333	0.265	0.2355	0.221667	0.229667

N = 10000

Trial No/ No of Children	1	2	3	4	5
Trial 1	17.111	12.452	10.832	10.12	9.585
Trial 2	17.144	12.429	10.683	10.185	9.424
Trial 3	16.86	12.566	10.876	10.06	9.725
Average	17.03833	12.48233	10.797	10.12167	9.578

N = 25 000

Trial No/ No of Children	1	2	3	4	5
Trial 1	111.282	69.414	57.335	51.364	45.853
Trial 2	111.889	75.651	58.384	51.378	46.262
Trial 3	109.209	67.851	58.422	51.541	46.503
Average	110.7933	70.972	58.047	51.42767	46.206

N = 50 000

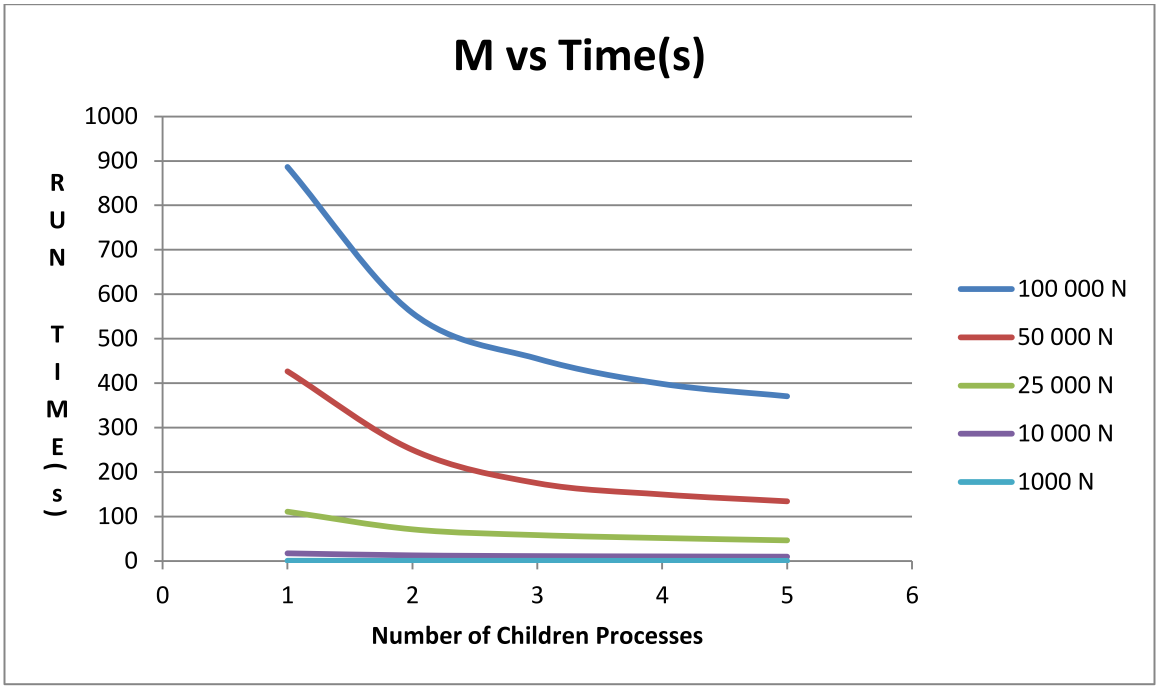
Trial No/ No of Children	1	2	3	4	5
Trial 1	422.102	218.103	183.629	149.726	139.076
Trial 2	430.446	256.929	158.136	149.08	130
Trial 3	426.842	274.6	182.892	149.68	133.05
Average	426.4633	249.8773	174.8857	149.4953	134.042

N = 100 000

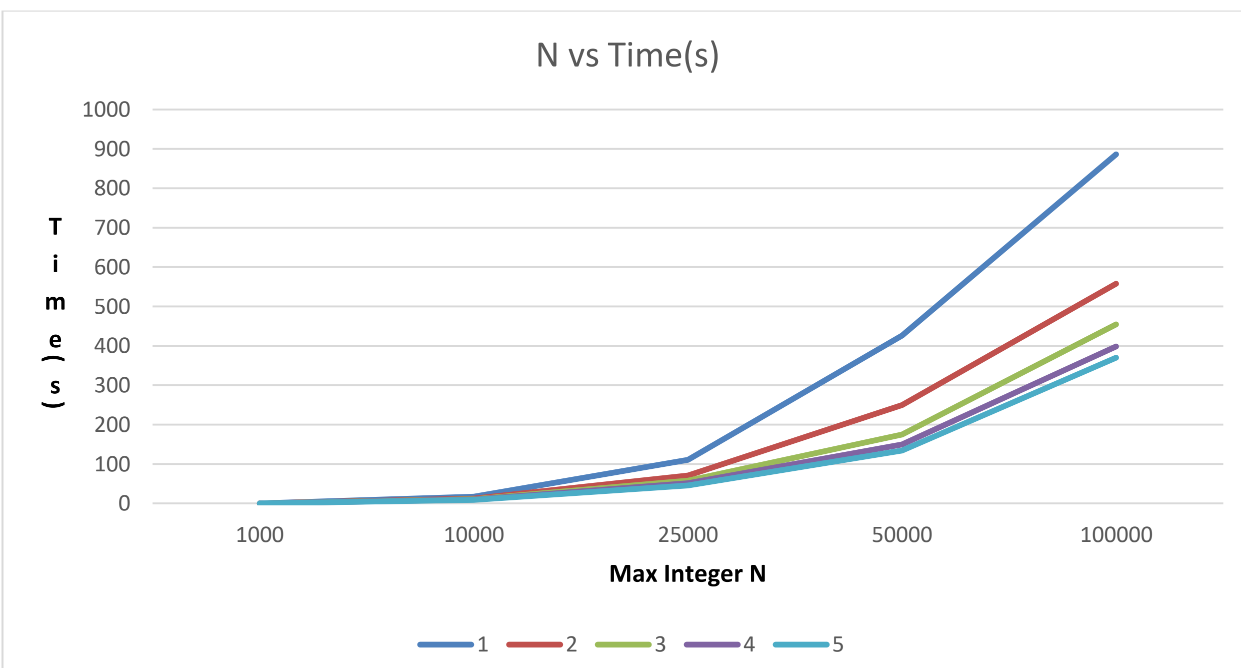
Trial No/ No of Children	1	2	3	4	5
Trial 1	889.883	540.0753	444.903	394.086	386.081
Trial 2	905.76	582.611	493.345	408.605	364.318
Trial 3	863	550.632	426.711	392.143	360.875
Average	886.2577	557.7728	454.9863	398.278	370.4247

Graphs of mqprime

Number of Children(M) vs Run Time(s)



Max Integer(N) vs Run Time(s)



*Lines represents number of child processes

Conclusion

In this part, I have done several experiments for the implementations done in part A and part B. Basically, I have attached 10 tables. 5 of them are for prime and 5 of them are for mqprime.

Firstly, for prime experiment, I have conducted run time experiment for 5 different M(no of child) and 5 different N(max integer). Each experiment repeated 3 times and average runtime values are calculated. Totally 75 experiments conducted for prime. I presented 2 graphs for prime. One of them represents no of children vs time when N is constant and the other one represents max integer N vs run time when no of children is constant. From the tables and graphs I have presented, the relationship between number of processes (M) and max integer(N) can be easily detected. Since it is easier to read graphs, i will explain my results through them. In first graph (M vs time), the series(lines) represents max integer for the experiment, you can see that as the number of children increases the time also tend to decrease as well. However it is also clear that there is no absolute linear relation between M and runtime. Meaning, run time does not down to half as the child number increases 1 to 2. This inconsistency is caused by the overhead created by multiple processes and their interprocess communications. For large N values, the impact of multiple processes is more obvious. From the second graphs it can be seen that as the max integer(N) increases the runtime tend to be increase as well however it can also be seen that as the number of child processes increases the time limit decreases

Secondly, for mqprime I have conducted same experiment for 5 different M and 5 different N values however in mqprime number of children is different than the first experiment. I presented 2 graphs for the mqprime as well. First graph represents no of children vs time when N is constant and the other one represents max integer N vs run time when no of children is constant. In first graph (M vs time), the series(lines) represents max integer for the experiment, you can see that as the number of children increases the time also tend to decrease as in case of prime experiment. From the same reason, it can be said that there is no linear relation between number of child processes and time. The overhead created by child processes and their interprocess communication decreases the efficiency. From the second graphs, we can interpret the same results as in the prime experiment case(second graph).

In conclusion, in part C of the project, I have done several experiments to understand interprocess commination and the relation between number of children processes and the amount of work they can do through their execution time. To get strong results, I have repeated each experiment 3 times for every variable changes (either for N or M). I have presented my results through the tables and the graphs and gave their explanations.