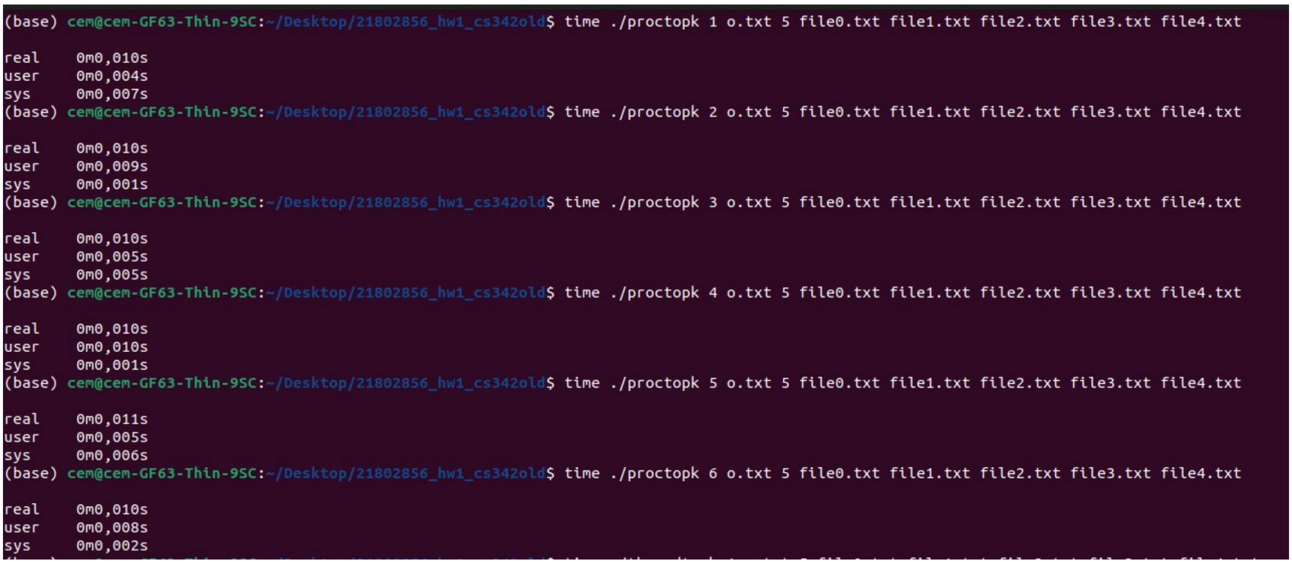
Selahattin Cem Öztürk
CS342-2

21802856
Project 1

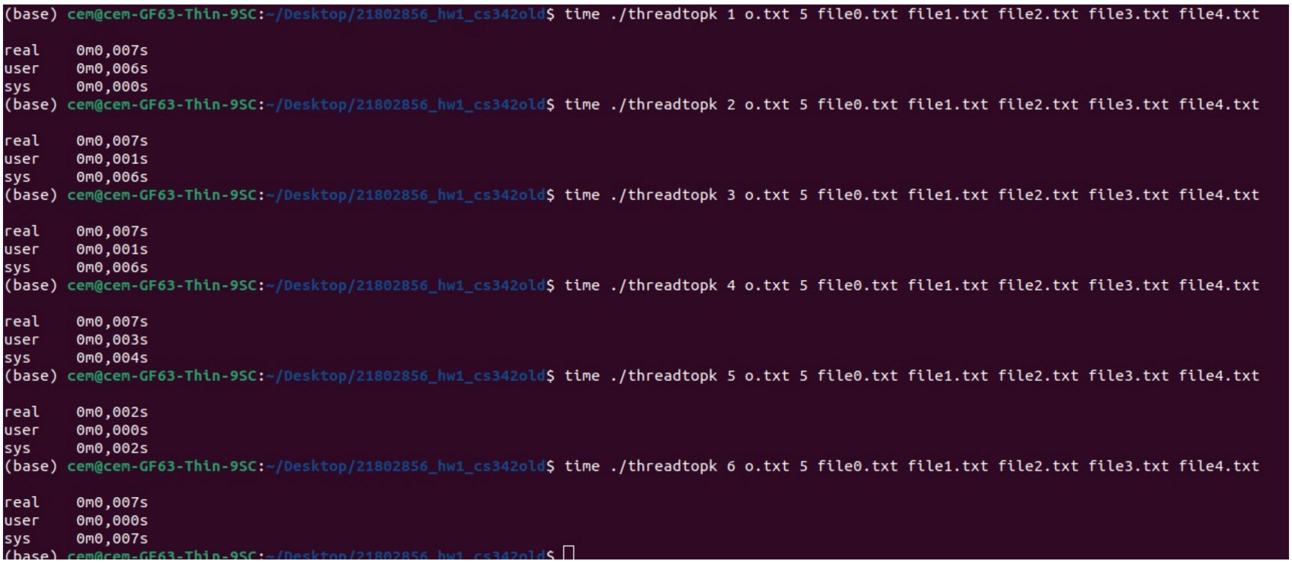
PART C

Experiment done with 6 different K values. Results are given below.



```
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 1 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,010s
user    0m0,004s
sys     0m0,007s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 2 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,010s
user    0m0,009s
sys     0m0,001s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 3 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,010s
user    0m0,005s
sys     0m0,005s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 4 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,010s
user    0m0,010s
sys     0m0,001s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 5 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,011s
user    0m0,005s
sys     0m0,006s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 6 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,010s
user    0m0,008s
sys     0m0,002s
```

Fig. 1: Runtimes of the proctopk program when it runned with 5 text files.

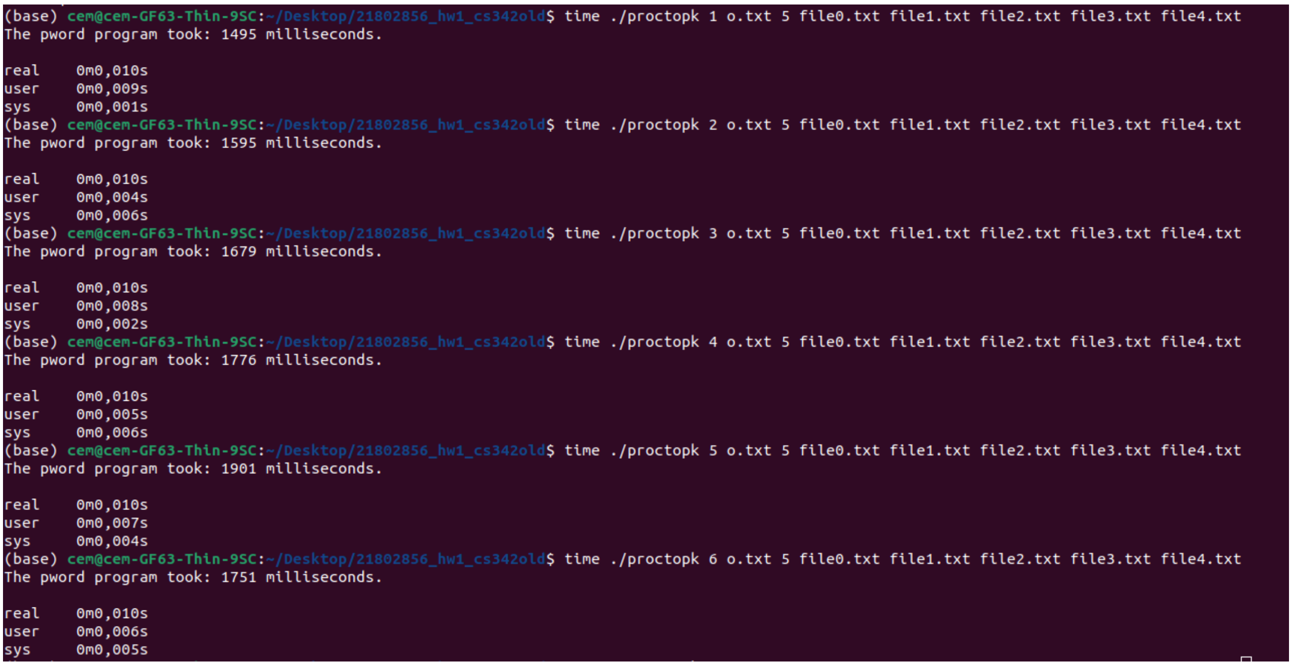


```
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 1 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,007s
user    0m0,006s
sys     0m0,000s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 2 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,007s
user    0m0,001s
sys     0m0,006s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 3 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,007s
user    0m0,001s
sys     0m0,006s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 4 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,007s
user    0m0,003s
sys     0m0,004s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 5 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,002s
user    0m0,000s
sys     0m0,002s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 6 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt

real    0m0,007s
user    0m0,000s
sys     0m0,007s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$
```

Fig. 2: Runtimes of the threadtopk program when it runned with 5 text files.

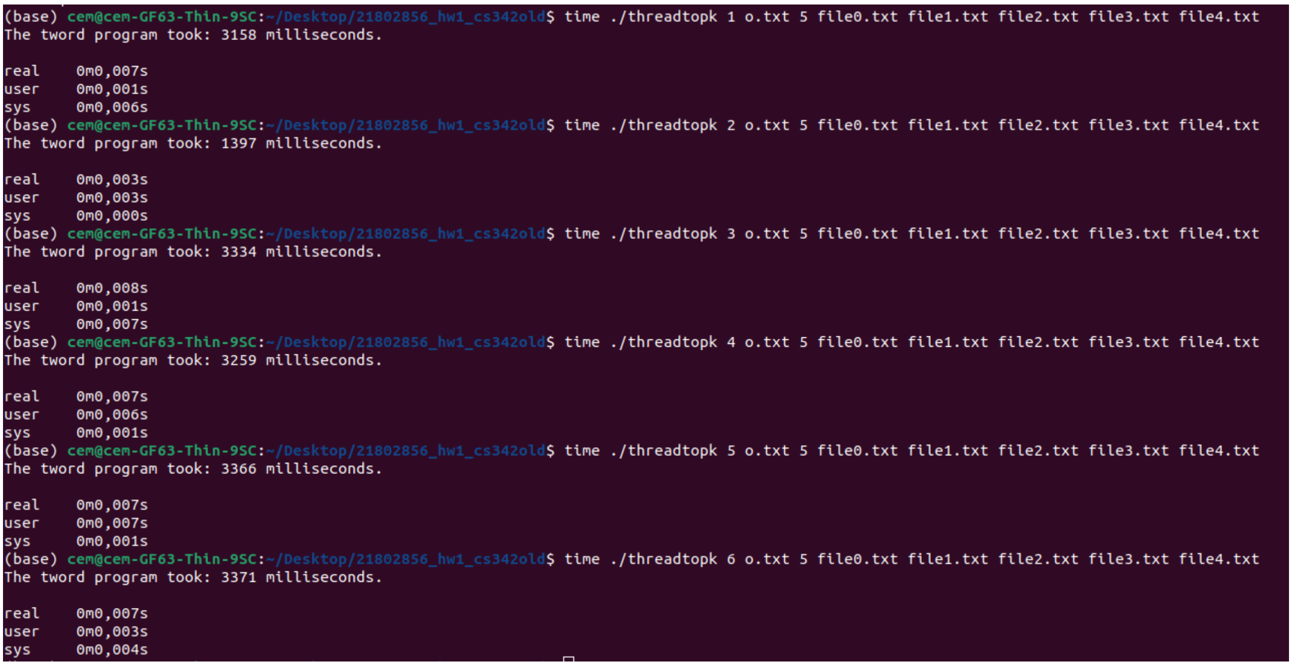
Realtime of the threadtopk program is smaller than proctopk. This is expected since the threads can utilize more than one CPU core at a time.

Runtimes also recorded with the help of the time library of C programming language. Results shown below.



```
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 1 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The pword program took: 1495 milliseconds.

real	0m0,010s
user	0m0,009s
sys	0m0,001s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 2 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The pword program took: 1595 milliseconds.

real	0m0,010s
user	0m0,004s
sys	0m0,006s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 3 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The pword program took: 1679 milliseconds.

real	0m0,010s
user	0m0,008s
sys	0m0,002s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 4 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The pword program took: 1776 milliseconds.

real	0m0,010s
user	0m0,005s
sys	0m0,006s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 5 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The pword program took: 1901 milliseconds.

real	0m0,010s
user	0m0,007s
sys	0m0,004s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./proctopk 6 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The pword program took: 1751 milliseconds.

real	0m0,010s
user	0m0,006s
sys	0m0,005s
```

Fig. 3: Runtimes of the proctopk program when it runned with 5 text files.



```
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 1 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The tword program took: 3158 milliseconds.

real	0m0,007s
user	0m0,001s
sys	0m0,006s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 2 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The tword program took: 1397 milliseconds.

real	0m0,003s
user	0m0,003s
sys	0m0,000s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 3 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The tword program took: 3334 milliseconds.

real	0m0,008s
user	0m0,001s
sys	0m0,007s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 4 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The tword program took: 3259 milliseconds.

real	0m0,007s
user	0m0,006s
sys	0m0,001s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 5 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The tword program took: 3366 milliseconds.

real	0m0,007s
user	0m0,007s
sys	0m0,001s
(base) cem@cem-GF63-Thin-9SC:~/Desktop/21802856_hw1_cs342old$ time ./threadtopk 6 o.txt 5 file0.txt file1.txt file2.txt file3.txt file4.txt
The tword program took: 3371 milliseconds.

real	0m0,007s
user	0m0,003s
sys	0m0,004s
```

Fig. 4: Runtimes of the threadtopk program when it runned with 5 text files.

As seen in the figure 3, up to K=5, there is a noticeable increase in runtimes. This is probably because my corpus have 6 different words. So not filtering at all decreases the runtime.

In the threadtopk program unexpected results occurred. This is probably due to program spends more time in kernel mode.

PLOTS

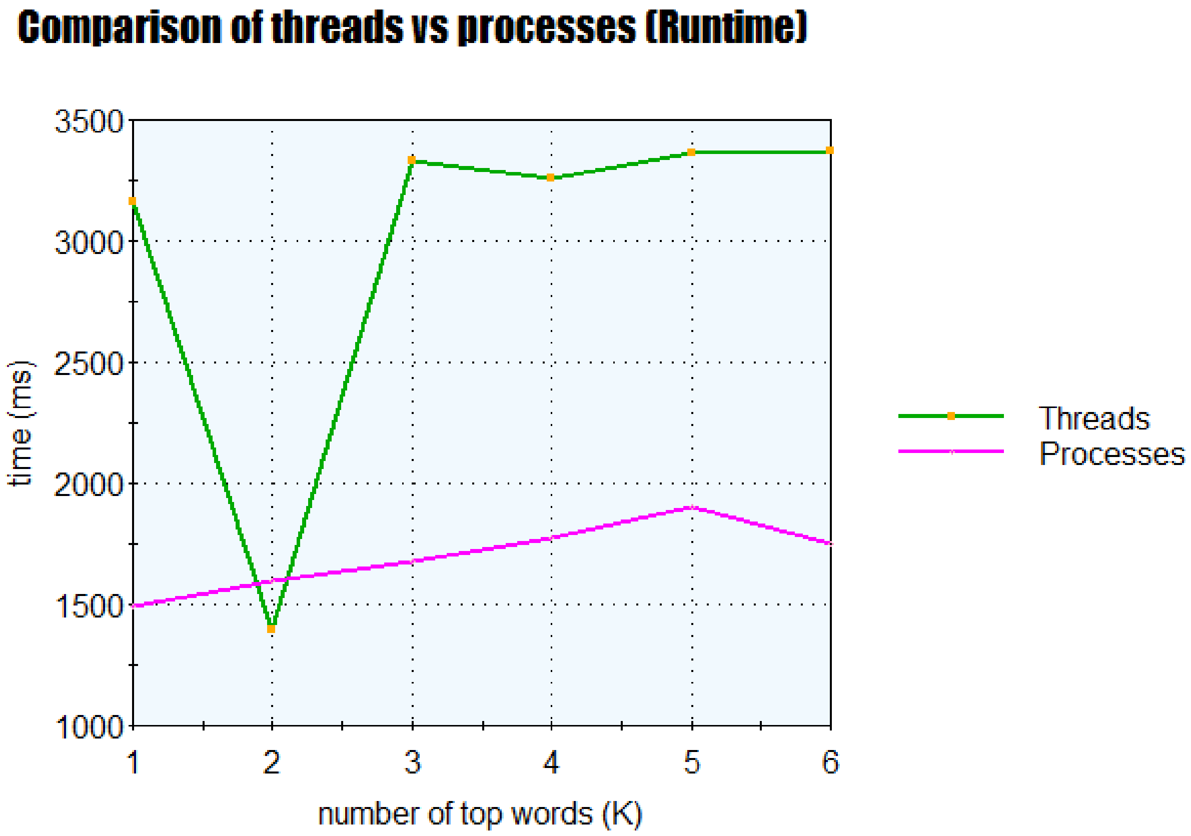


Fig. 5: K vs time(ms) of processes and threads.

Threads seems like they have a disadvantage over processes, however, this is probably due to the corpus size. Since the number of different words and size of the text files were not big, it is really impossible to derive a conclusion from this graph.

But in processes until the K equals to 5, there is a steady increase but in threads K=1, 3 ,4 ,5 , and 6 seems like have a close value. With the different data it seems like threads dominate processes.