

## PA4 Report by Siyuan Yang

### Data Requests:

$$n = 15k$$

w	runtime(s)
2	19.331515
4	9.64298
8	4.80169
16	2.397181
32	1.197032
64	0.599982
128	0.30304
256	0.157553
512	0.091743

Table 1. Runtime with different numbers of worker threads

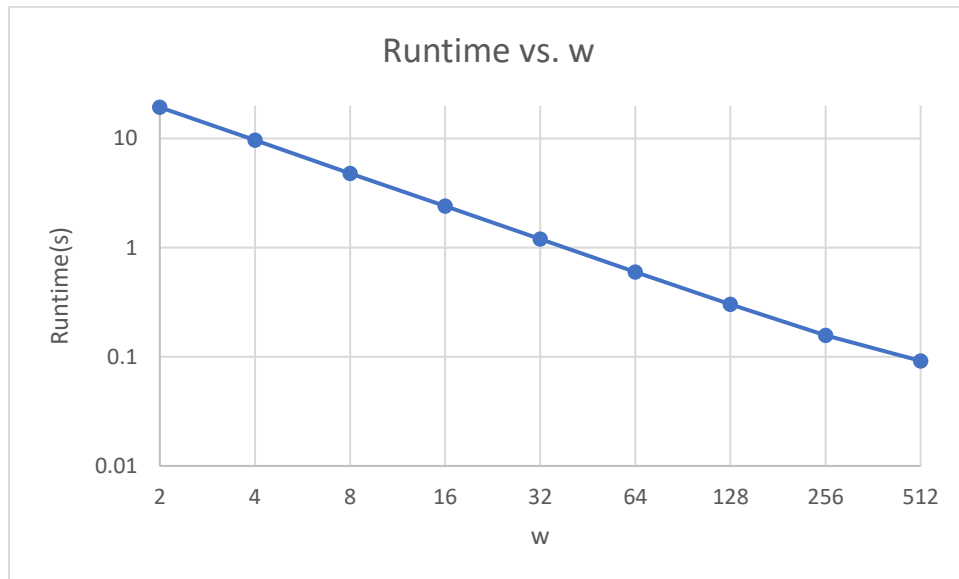


Diagram 1. Runtime vs. w

### Discussion:

For runtime with different numbers of worker threads, as the number of worker threads increases, the running time decreases proportionately. The change of runtime is linearly proportional to the change of number of worker threads, but when the number of worker threads surpasses 500, the change of runtime would be smaller and even remain unchanged.

$$n = 15k$$

b	runtime(s)
2	0.201039
4	0.198213
8	0.196908
16	0.19722
32	0.196778
64	0.197375
128	0.197239
256	0.197017
512	0.19685

Table 2. Runtime with different sizes of request buffer

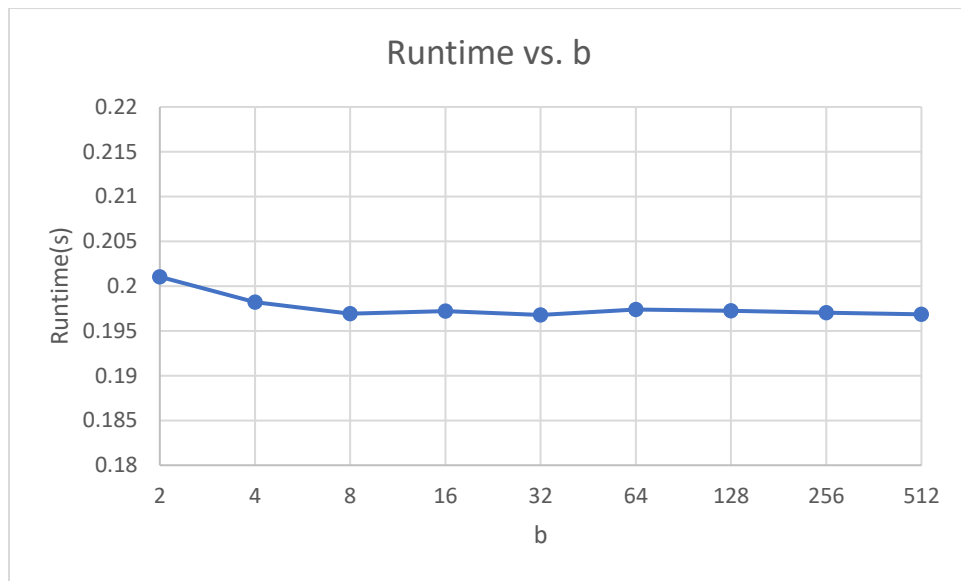


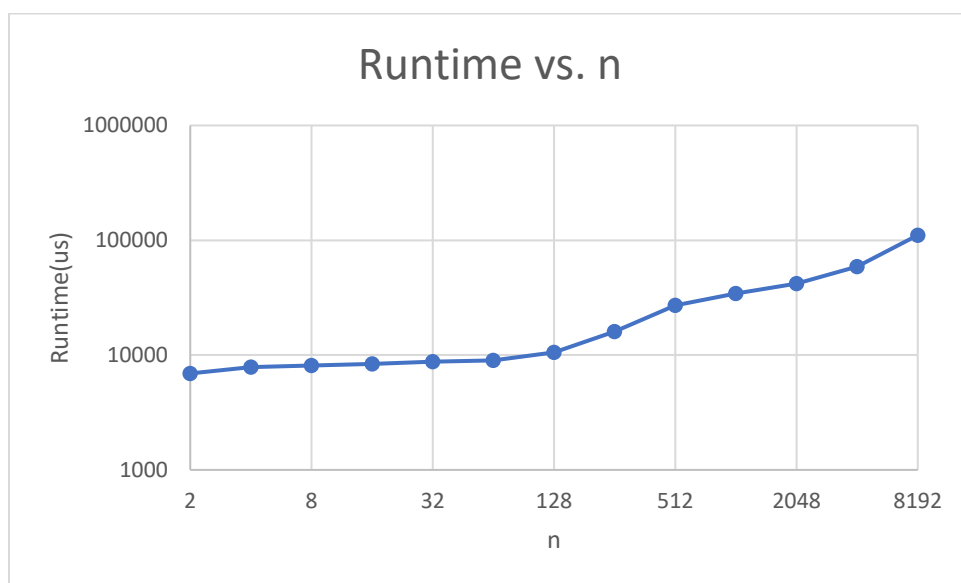
Diagram 2. Runtime vs. b

### Discussion:

For runtime with different sizes of request buffer, the change of size of does not necessarily affect the performance of the program. As  $b$  is the size of request buffer, it is no longer the bottleneck of the program since the context switch of the worker thread contribute more when the size gets larger. There is not an obvious trend in the relationship between buffer size and runtime.

n	runtime(us)
2	6924
4	7846
8	8096
16	8358
32	8770
64	9000
128	10583
256	15978
512	27078
1024	34318
2048	42030
4096	59105
8192	110765

*Table 3. Runtime with different numbers of requests per patient*



*Diagram 3. Runtime vs. n*

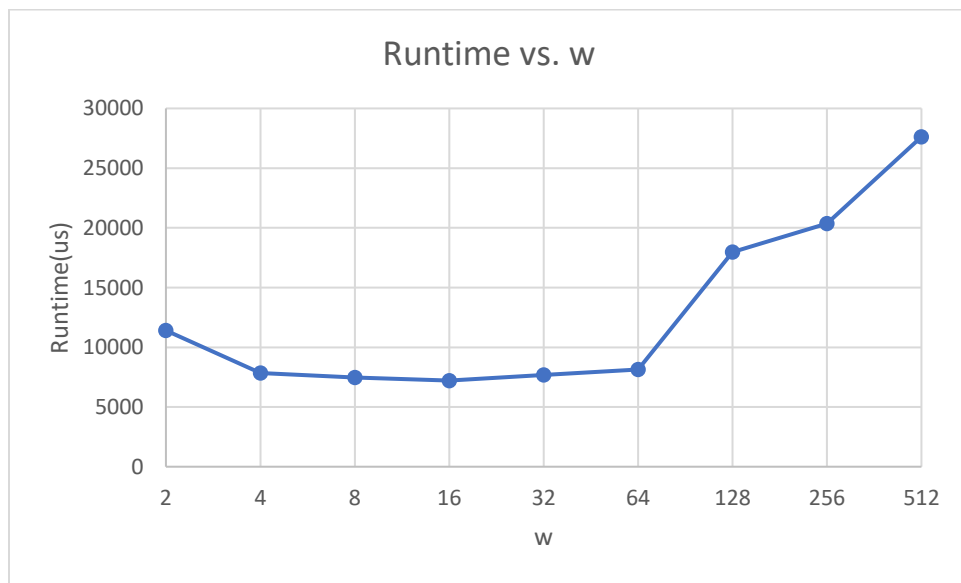
### Discussion:

For runtime with different numbers of requests per patient, as the number of requests per patient increases, the running time increases. The change is linear for certain range, as the number of requests would allow the program to process the data longer.

### File Request:

w	runtime(us)
2	11400
4	7836
8	7461
16	7214
32	7671
64	8126
128	17966
256	20351
512	27621

*Table 4. Runtime with different numbers of worker threads*



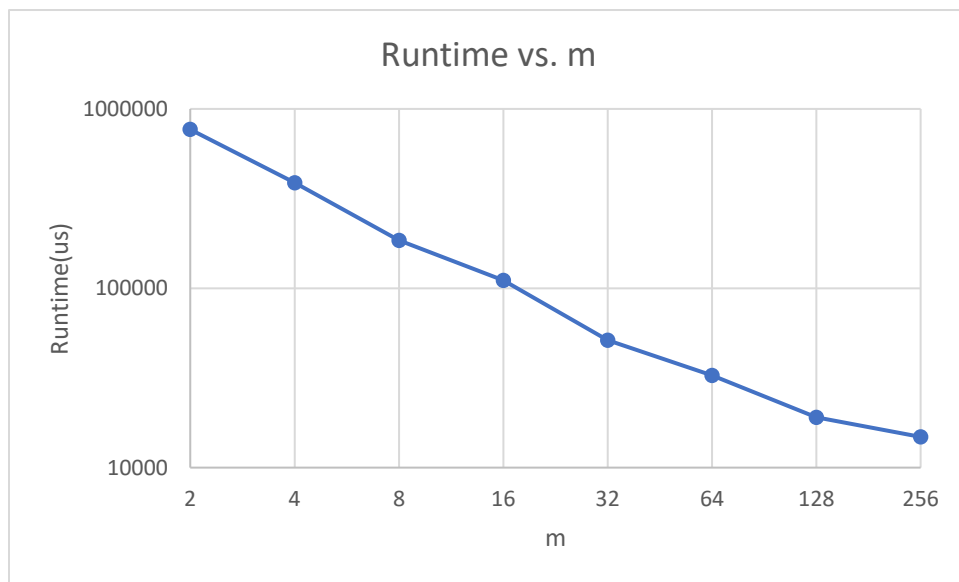
*Diagram 4. Runtime vs. w*

### Discussion:

For runtime with different numbers of worker threads, as the number of requests per patient increases, the running time decreases then increases. There is no linear scaling overall, but it has some linear feature within certain range.

m	runtime(us)
2	768052
4	388155
8	185187
16	110490
32	51301
64	32696
128	19066
256	14833

*Table 5. Runtime with different sizes of buffer capacity*



*Diagram 5. Runtime vs. m*

### Discussion:

For runtime with different buffer capacity, as the number of buffer capacity increases, the running time decreases. The change of runtime is linearly proportional to the change of buffer capacity, as the buffer capacity increases, the program can process the data more efficiently.