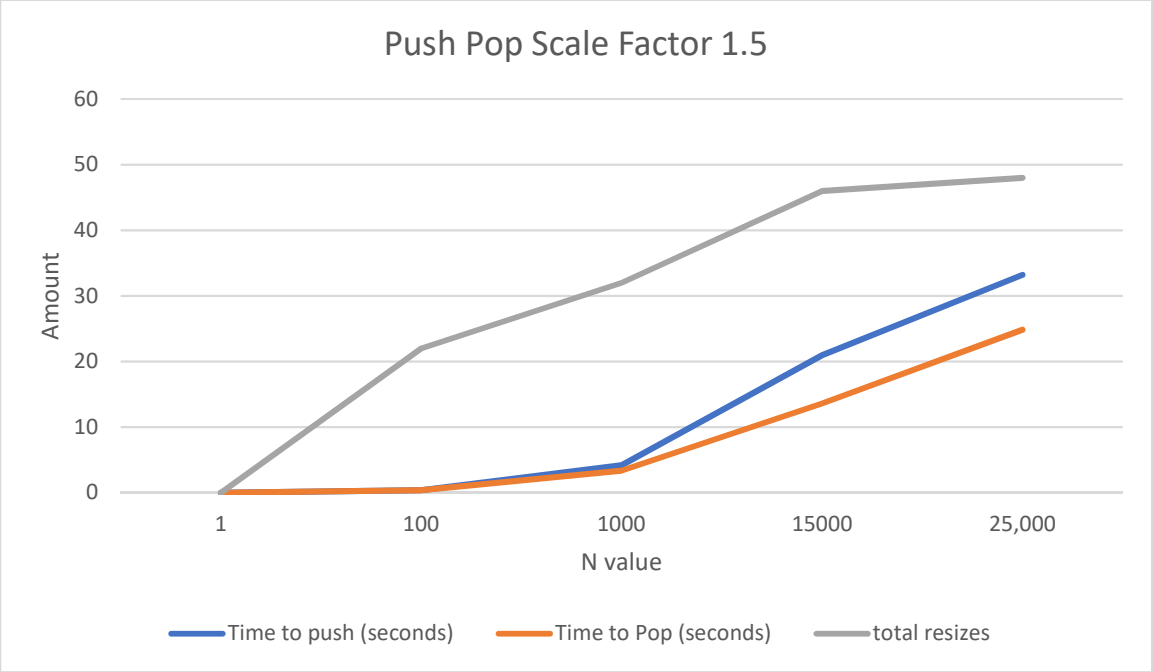
Throughout the process of collecting data points a noticeable trend was observed whereby the size of the scale factor would dramatically increase the amount of time the program took to complete its task. Of note is that when running the program it was observed that the scale factor could increase the run duration by almost 3 times depending on what the value was. For example, when pushing the function the shortest time for completion was at 25000 N with a scale factor of 3 and the time for completion was ~30 seconds but at 25000 N and a scale factor of 100 the time increased to ~80 seconds. Of all the variables changed, this one appears to result in the greatest disparity of program run time.

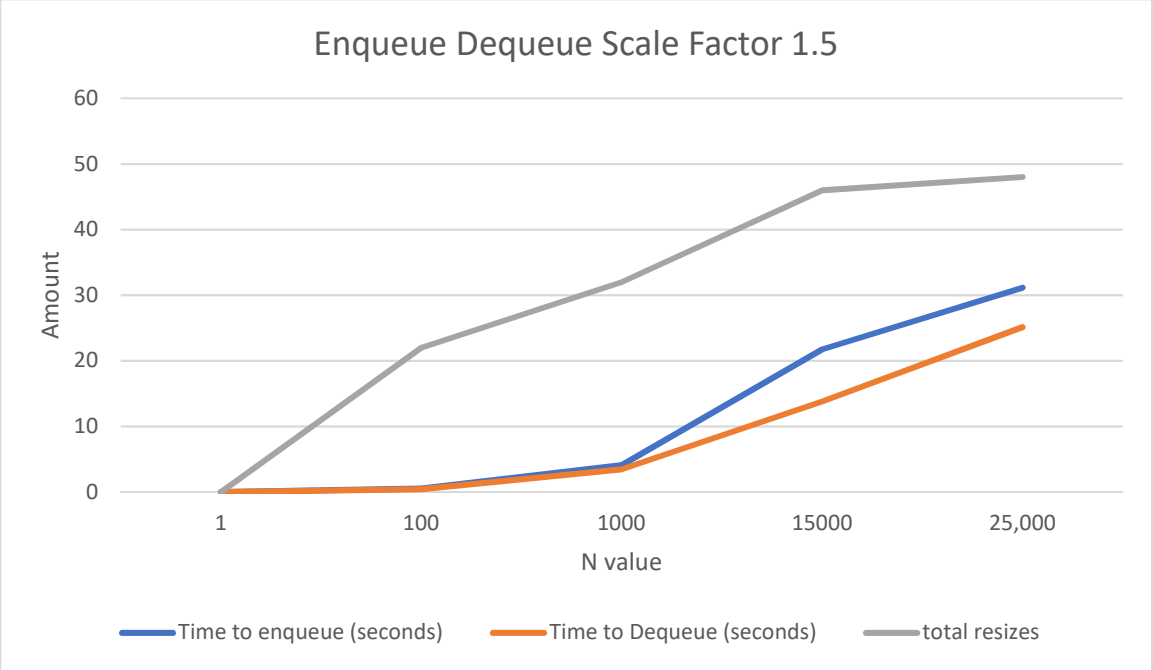
When looking at data trends with respect to the N value, observations indicate that a larger N increases the time duration before program completion. This is a natural and intuitive relationship the data has with the program as a larger N indicates a larger data load. With N values of less than 100 the program completion time is well below 1 second regardless of the other variables at play and when the N value is above 1000 initial observations appear to indicate an arithmetic increase in program run time with the sole discrepancy being when the "Scale Factor" is at 100 and the "N" value is at 25,000. At this point, when compared to the previous values, the increase begins to look like a geometric increase in program run time.

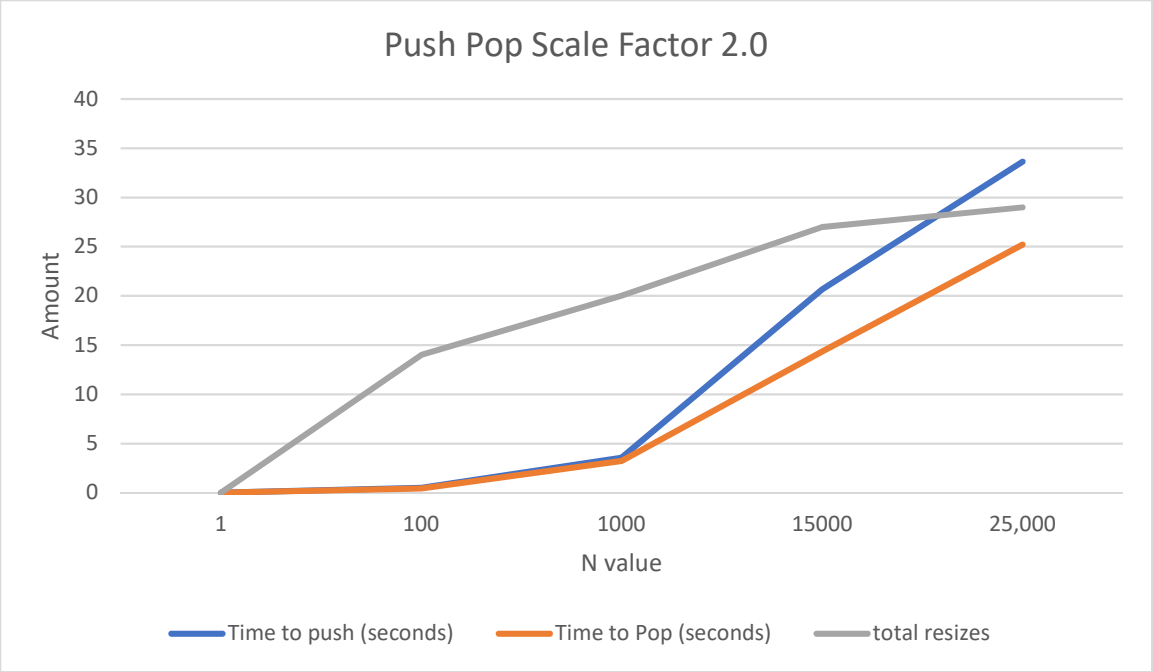
The Scale Factor and N variables impact the number of times the ABS and ABQ is resized in an inverse relationship. The reason for this is explain in two parts. First, the scale factor is what determines the new maximum capacity of the arrays when they are resized after becoming full. As such, the larger the scale factor the fewer times an array will need to be resized. The N value has the opposite effect due to the fact that larger data sets will require larger arrays to carry the data. Due to this, a sufficiently large dataset (i.e. large N) will result in an array being filled and resized and filled again more often than a smaller dataset would. In short, the larger the scale factor the fewer times a dataset will be resized and the larger the N value the more times a dataset will be resized.

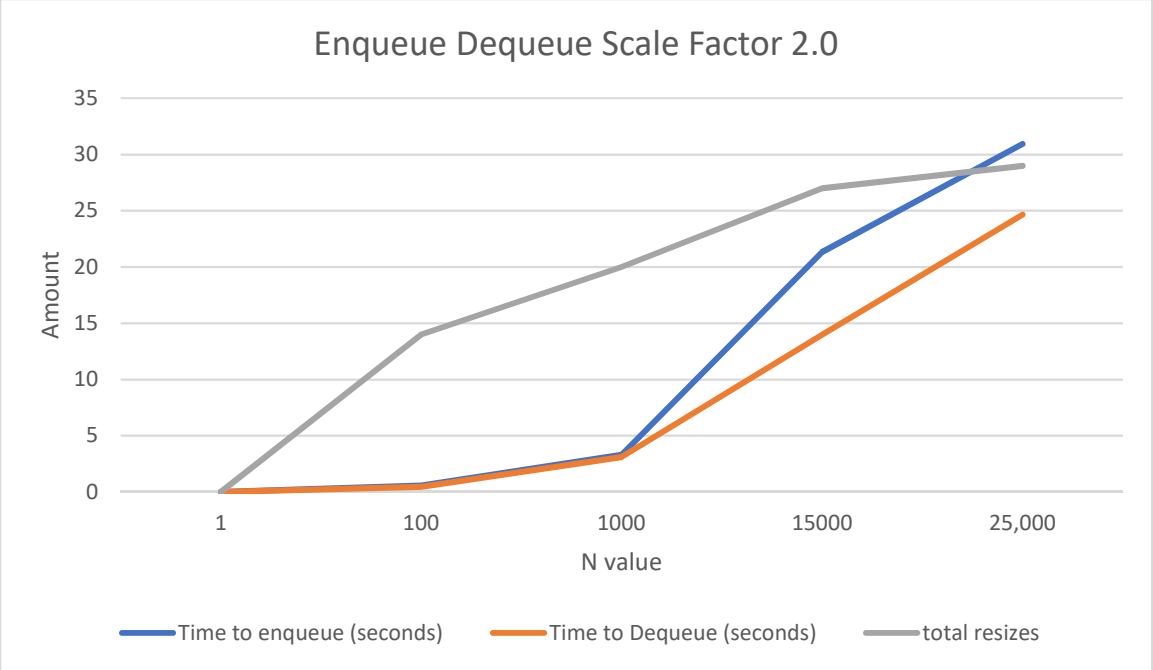
The best scale factor, based on the parameters input during this test, appears to be 3. This assertion stems from averaging the data collected with respect to the amount of time the program took to complete. On average, when looking at the time it takes to push a stack the data indicates that a scale factor of 1.5 = 11.75167 seconds; 2 = 11.66988 seconds; 3 = 11.33556 seconds; 10 = 12.06401 seconds; and 100 = 21.01401 seconds. As such, while a scale factor of 2 and 3 are similar in the time the program takes to complete. A scale factor of 3 still proves to be the fastest of the various options.

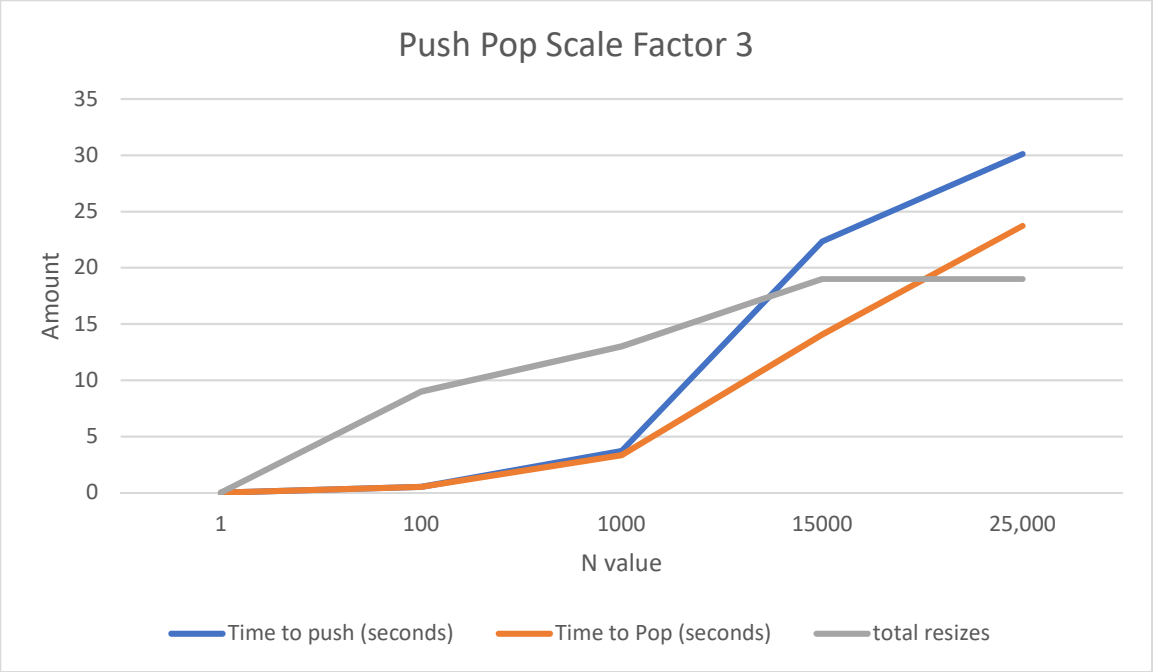
As a general, trend the time difference between the ABS and ABQ programs are similar however, ABQ does appear to have a consistent edge over the ABS program. A possible reason for this could stem from a difference in how the leaker files are treating each one. If one program is structured in such a way that these files are being called more often the program would be expected to finish run time slightly later than an iteration of the program that does not call the leaker files as often. An additional, though adjacent, explanation could stem from the number of operations being called for by each program. If one program has more operations than the other this could also impact the programs performance.

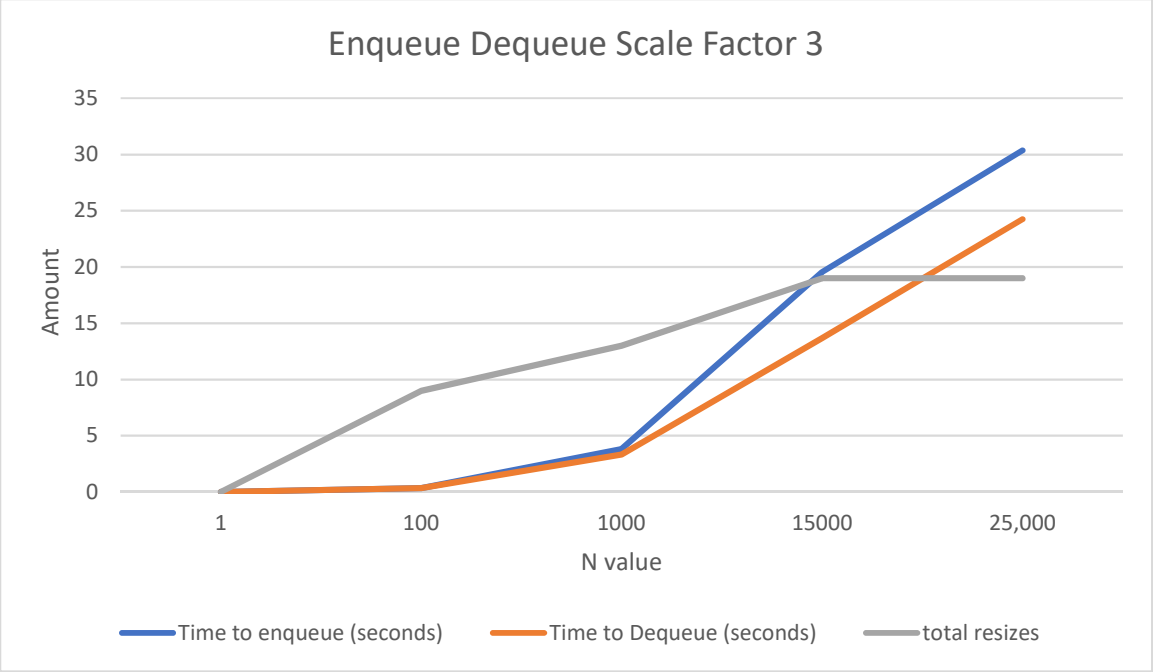

Push Pop Scale Factor 1.5
60
50
40
30
20
10
0
Amount
1
100
1000
15000
25,000
N value
Time to push (seconds)
Time to Pop (seconds)
total resizes

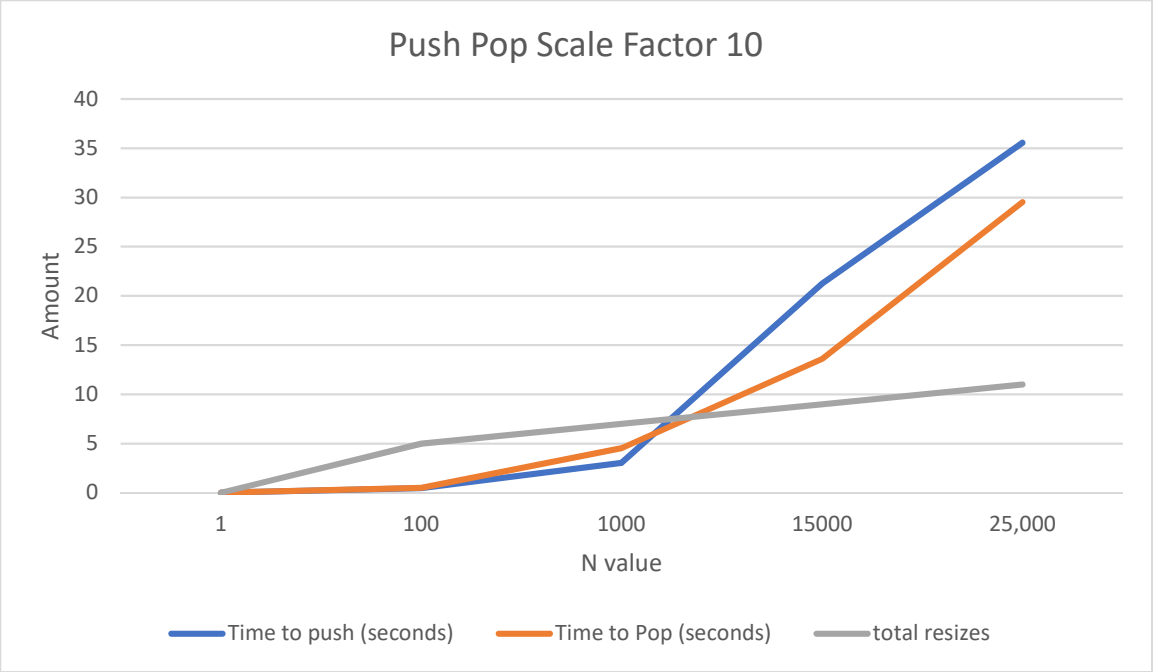

Enqueue Dequeue Scale Factor 1.5
60
50
40
30
20
10
0
Amount
1
100
1000
15000
25,000
N value
Time to enqueue (seconds)
Time to Dequeue (seconds)
total resizes

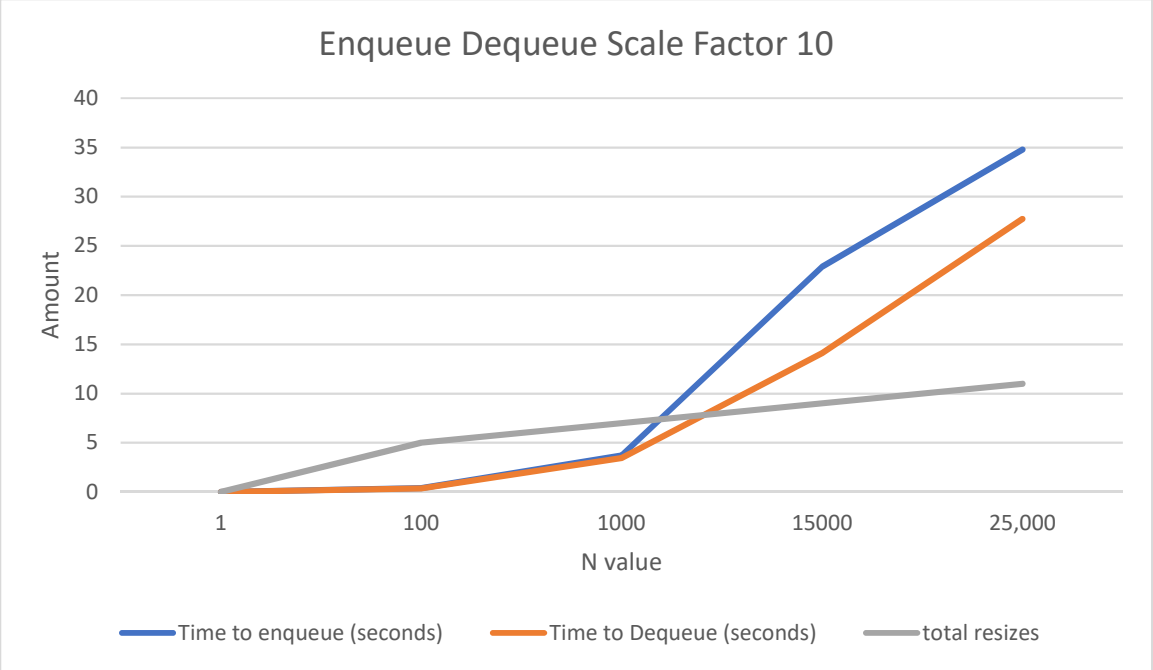

Push Pop Scale Factor 2.0
40
35
30
25
20
15
10
5
0
Amount
1
100
1000
15000
25,000
N value
Time to push (seconds)
Time to Pop (seconds)
total resizes

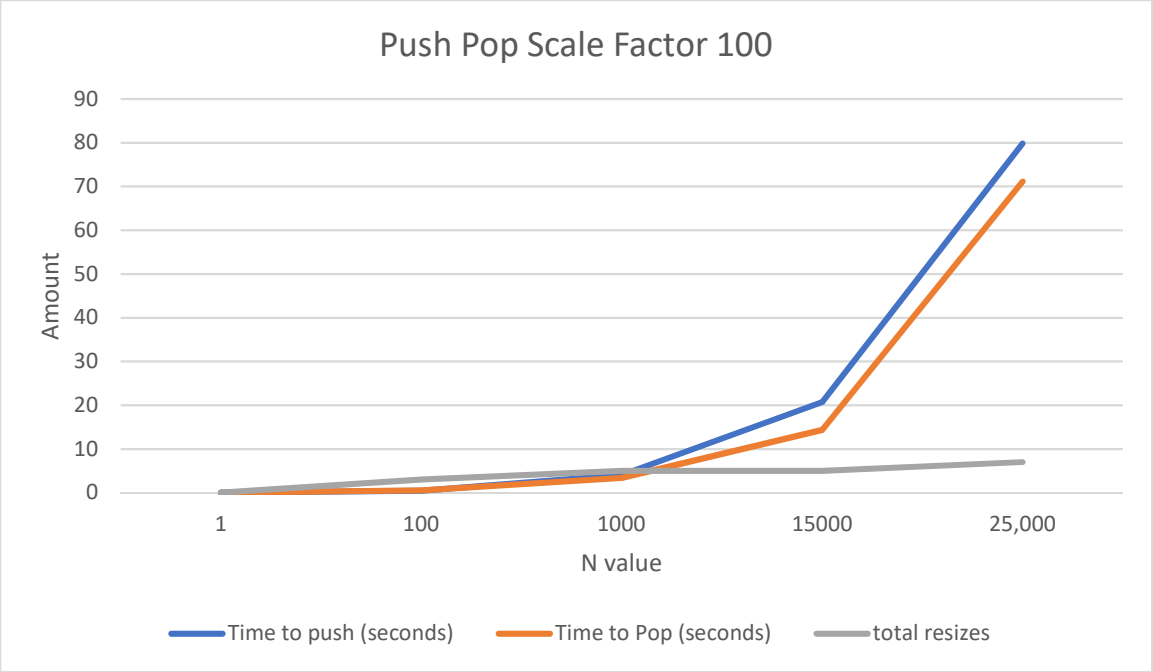

Enqueue Dequeue Scale Factor 2.0
35
30
25
20
15
10
5
0
Amount
1
100
1000
15000
25,000
N value
Time to enqueue (seconds)
Time to Dequeue (seconds)
total resizes

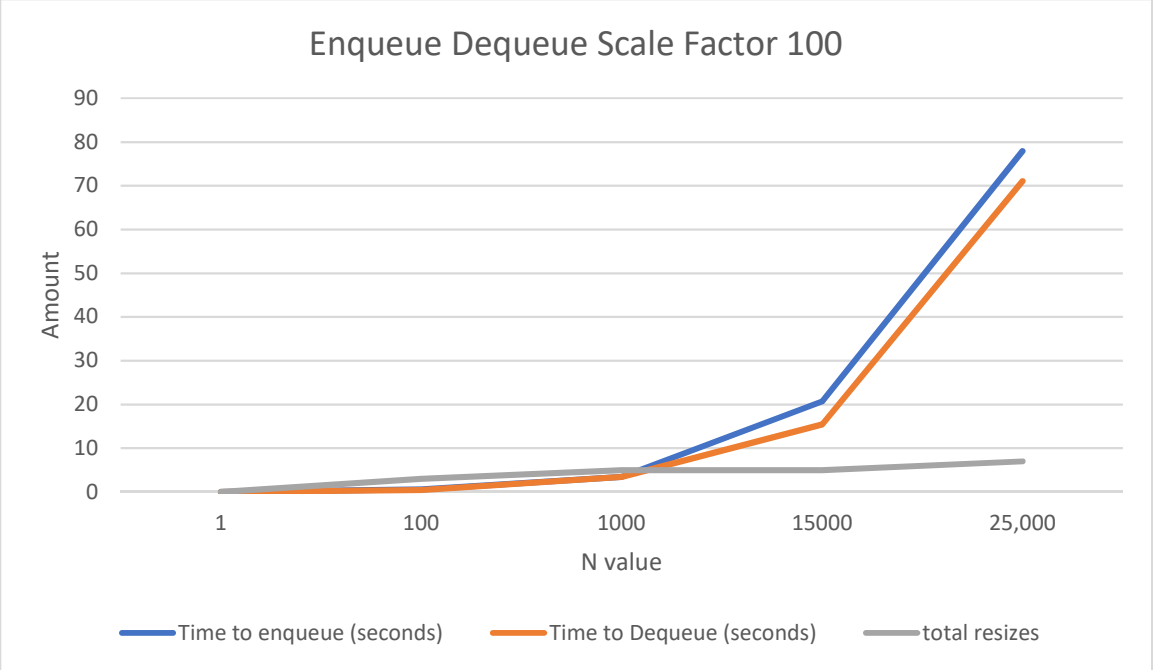

Push Pop Scale Factor 3
35
30
25
20
15
10
5
0
Amount
1
100
1000
15000
25,000
N value
Time to push (seconds)
Time to Pop (seconds)
total resizes


Enqueue Dequeue Scale Factor 3
35
30
25
20
15
10
5
0
Amount
1
100
1000
15000
25,000
N value
Time to enqueue (seconds)
Time to Dequeue (seconds)
total resizes


Push Pop Scale Factor 10
40
35
30
25
20
15
10
5
0
Amount
1
100
1000
15000
25,000
N value
Time to push (seconds)
Time to Pop (seconds)
total resizes


Enqueue Dequeue Scale Factor 10
40
35
30
25
20
15
10
5
0
Amount
1
100
1000
15000
25,000
N value
Time to enqueue (seconds)
Time to Dequeue (seconds)
total resizes


Push Pop Scale Factor 100
90
80
70
60
50
40
30
20
10
0
Amount
1
100
1000
15000
25,000
N value
Time to push (seconds)
Time to Pop (seconds)
total resizes


Enqueue Dequeue Scale Factor 100
90
80
70
60
50
40
30
20
10
0
Amount
1
100
1000
15000
25,000
N value
Time to enqueue (seconds)
Time to Dequeue (seconds)
total resizes

Scale Factor	N	Time to push (seconds)	Time to Pop (seconds)	total resizes	Time to enqueue (seconds)	Time to Dequeue (seconds)	total resizes
1.5	1	0.0005553	0.0000484	0	0.0006104	0.0000682	0
1.5	100	0.397174	0.416626	22	0.507273	0.443653	22
1.5	1000	4.18261	3.38232	32	4.10229	3.45904	32
1.5	15000	20.9707	13.6023	46	21.7616	13.8005	46
1.5	25,000	33.2073	24.8632	48	31.1365	25.1414	48
2	1	0.0004982	0.0000404	0	0.0004869	0.0000604	0
2	100	0.493134	0.427895	14	0.577178	0.470551	14
2	1000	3.56115	3.24358	20	3.31442	3.11084	20
2	15000	20.6605	14.3367	27	21.3379	13.9765	27
2	25,000	33.6341	25.2153	29	30.9397	24.6542	29
3	1	0.0007473	0.0000878	0	0.0010734	0.0000465	0
3	100	0.52847	0.521242	9	0.350301	0.343235	9
3	1000	3.69556	3.35172	13	3.82178	3.32265	13
3	15000	22.3376	14.0628	19	19.5615	13.6924	19
3	25,000	30.1154	23.7155	19	30.3607	24.2435	19
10	1	0.0007824	0.0000505	0	0.0004565	0.0000408	0
10	100	0.47464	0.497267	5	0.378655	0.376262	5
10	1000	3.04523	4.5221	7	3.72343	3.44626	7
10	15000	21.2351	13.5939	9	22.8972	14.1013	9
10	25,000	35.5643	29.537	11	34.7936	27.7474	11
100	1	0.0007551	0.000069	0	0.0004974	0.0000682	0
100	100	0.449369	0.569657	3	0.558536	0.413056	3
100	1000	4.03642	3.3696	5	3.45677	3.44056	5
100	15000	20.7459	14.3785	5	20.6902	15.463	5
100	25,000	79.8376	71.1285	7	77.9348	71.0662	7

Table 1: Collected Data