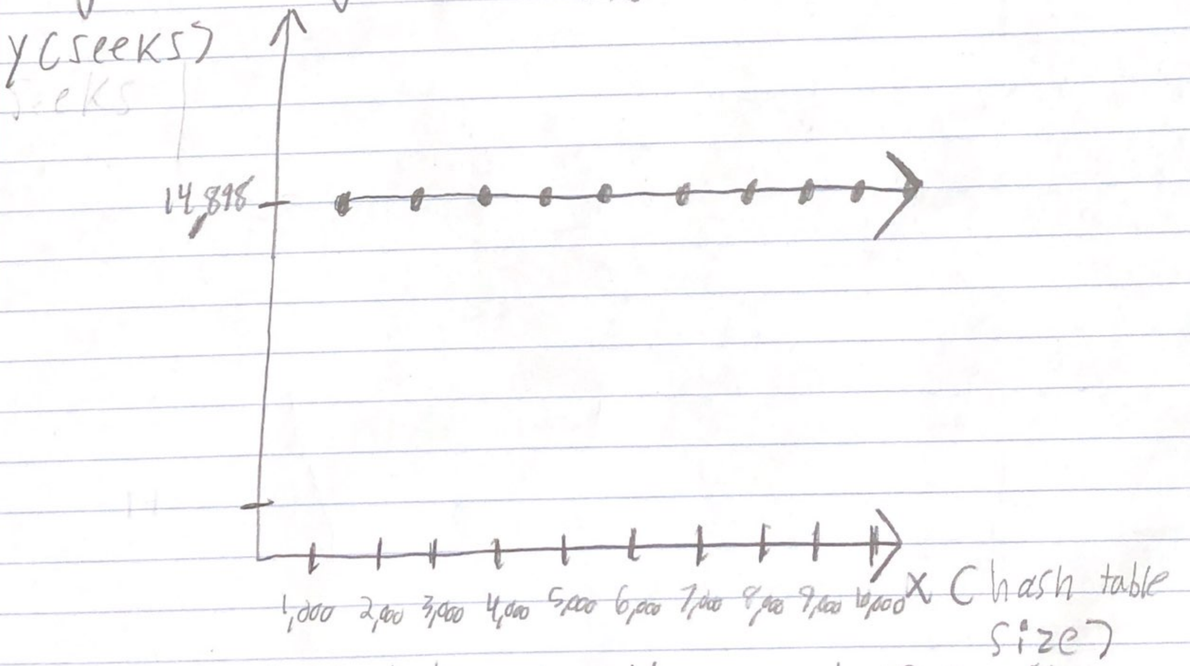
Writeup

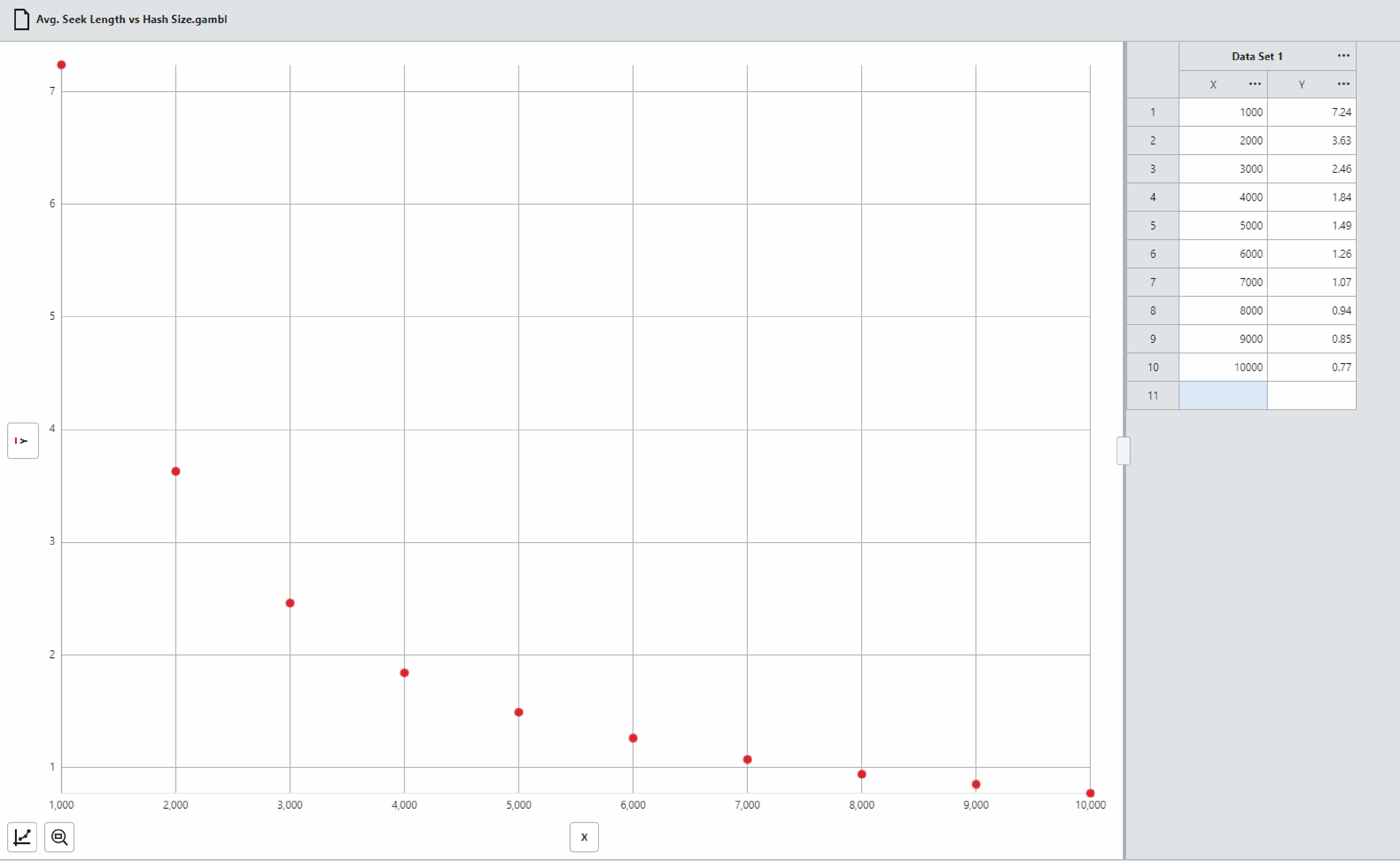
To analyze the results of assignment 7, the variables "seeks" & "links" were created to track the amount of times we called a linked list lookup & the amount of node links traversed.

The following graphs display the total number of seeks and the average seek length as we vary the hash table and bloom filter size. Numbers are based of the same file "banhammer.c"

NOTE: avg. seek length = $\frac{\text{links}}{\text{seeks}}$



- As we vary hash table size the amount of seeks does not change. This is likely due to the fact that we will always need to look up the same amount of words that we suspect are bad.



Avg. Seek Length vs Hash Size.gambl

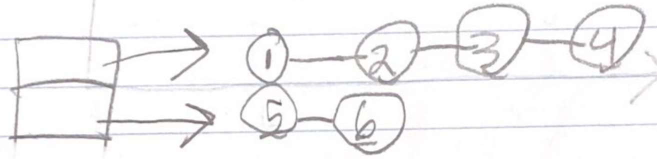
Data Set 1

	X	Y
1	1000	7.24
2	2000	3.63
3	3000	2.46
4	4000	1.84
5	5000	1.49
6	6000	1.26
7	7000	1.07
8	8000	0.94
9	9000	0.85
10	10000	0.77
11		

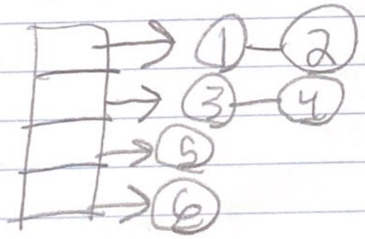
avg. seek length = $\frac{\text{links}}{\text{seeks}}$

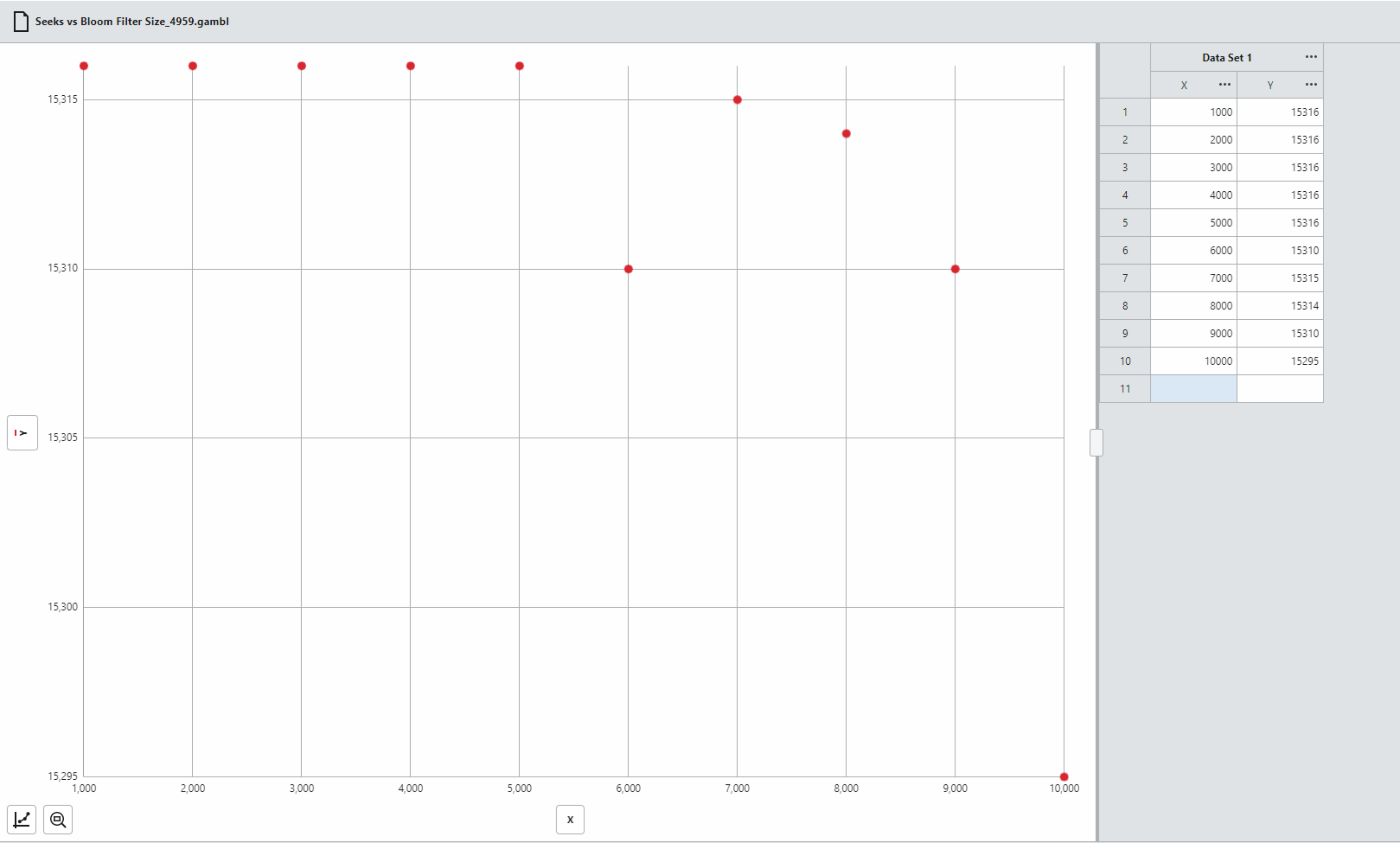
- As the Hash table size is increased by 1,000, we can clearly see a decrease in the average seek length. This is because the more space we have in our hash table the less "cramped" it becomes for linked lists. In other words, more hash table space means less nodes to traverse per linked list.

Small Hash Table



Large Hash Table





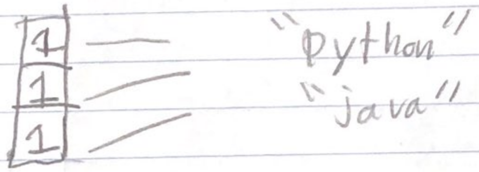
Seeks vs Bloom Filter Size_4959.gambl

Data Set 1

	X	Y
1	1000	15316
2	2000	15316
3	3000	15316
4	4000	15316
5	5000	15316
6	6000	15310
7	7000	15315
8	8000	15314
9	9000	15310
10	10000	15295
11		

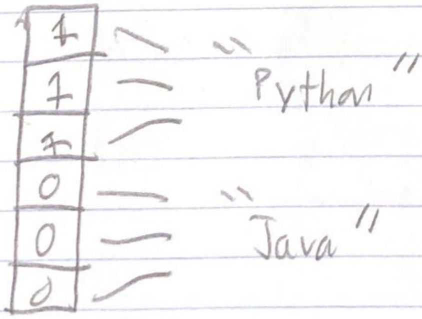
- As we increase our bloom filter size by 1,000 we can clearly see that the amount of seeks remain at a constant (15316) until we reach a bloom filter size of 6,000. Then, our seeks noticeably decrease which is likely due to our system having less false-positives and thus decreasing the amount of times we must verify a word ~~of~~ being bad

Small Bloom Filter

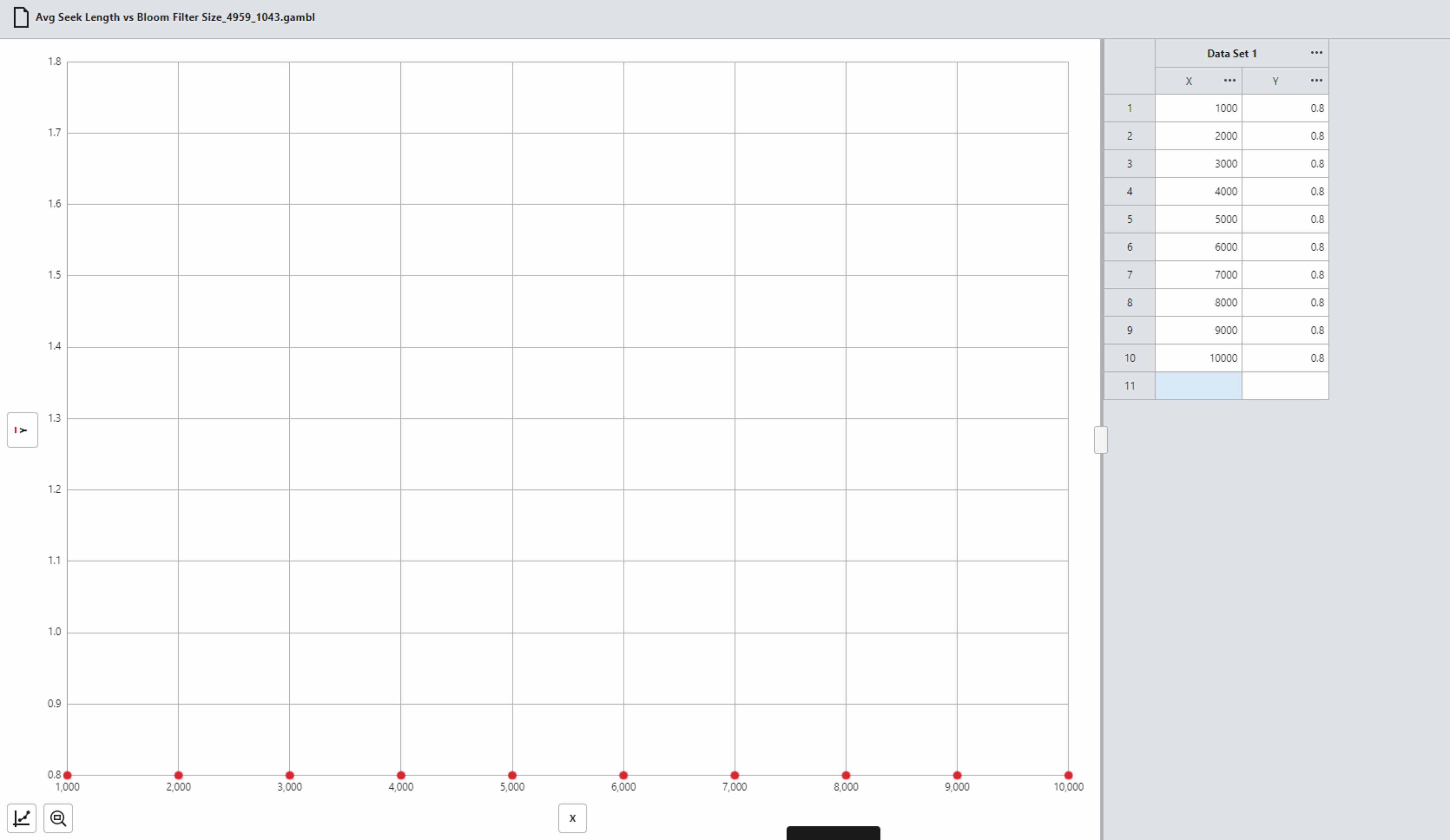


- "Python" and "Java" are forced to share the bits bc filter is too small... we aren't sure which one is the real positive transgression.

Large Bloom Filter



- Now we have more space & have more certainty as to which word might be a transgression



Avg Seek Length vs Bloom Filter Size_4959_1043.gambl

Data Set 1

	X	Y
1	1000	0.8
2	2000	0.8
3	3000	0.8
4	4000	0.8
5	5000	0.8
6	6000	0.8
7	7000	0.8
8	8000	0.8
9	9000	0.8
10	10000	0.8
11		

- As we increase the bloom filter size, there is a minimal decrease in our avg. seek length. This makes sense because our seeks or the amount of times we need to call lookup() is decreased.

$$\text{avg. seek length} = \frac{\text{links}}{\text{seeks}}$$

Now, when we compare our results w/ the move-to-front option enabled compared to when it is not we will notice a difference.

Move To Front Enabled	Move-To Front Disabled
Seeks; 14898	Seeks: 14898
Avg. Seek Length: 0.740099	Avg. Seek Length: 0.771781

- It is clear that the avg. seek length is the only difference w/ the disabled version being greater than when move to front was enabled.
- This means that the amount of links we are traversing is greater which makes sense because humans tend to be repetetive in their speech & some words are used more often i.e. "the" or "something".