Data Structures Lab 9 Benchmarking Report:

For the tests from the lab description, these were my first 6 results:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Number of words found** | **Time in seconds** | **Time in minutes** |
| 4way ArrayList | 7 words (3x3) | 1.005412498 | 0.016756875 |
| 8way ArrayList | 18 words (3x3) | 1.472623299 | 0.024543722 |
| 4way LinkedList | 12 words (4x4) | 29.9274349 | 0.498790582 |
| 8way LinkedList | 40 words (4x4) | 141.0841306 | 2.351402177 |
| 4way HashSet | 33 words (6x6) | 107.8288084 | 1.797146807 |
| 4way TreeSet | 33 words (6x6) | 131.3407856 | 2.189013093 |

From this data, it appears that doing an 8-way search takes longer than doing a 4-way search, which is to be expected. Additionally, more words were found in 8-way searches, which is to be expected as well. For the 6 by 6 word search, the hash set appeared to have found the same amount of words in a shorter amount of time, but there isn’t much of a conclusion to be drawn from this data yet.

More data other than the data in the lab description:

* 4way LinkedList (3x3): 7 words, 1.778297299 seconds, and 0.029638288316666667 minutes.
* 8way LinkedList (3x3): 18 words, 3.9979678 seconds, and 0.06663279666666667 minutes.
* 4way HashSet (3x3): 7 words, 0.021480101 seconds, and 3.580016833333334E-4 minutes.
* 4way TreeSet (3x3): 7 words, 0.022806700000000003 seconds, and 3.8011166666666673E-4 minutes.
* 8way HashSet (3x3): 18 words, 0.035323599000000004 seconds, and 5.8872665E-4 minutes.
* 8way TreeSet (3x3): 18 words, 0.038731399 seconds, and 6.455233166666667E-4 minutes.
* 4way ArrayList (4x4): 12 words, 9.186981600000001 seconds, and 0.15311636 minutes.
* 8way ArrayList (4x4): 40 words, 49.248735800000006 seconds, and 0.8208122633333335 minutes.
* 4way HashSet (4x4): 12 words, 0.09287639900000001 seconds, and 0.0015479399833333335 minutes.
* 8way HashSet (4x4): 40 words, 0.5993728 seconds, and 0.009989546666666667 minutes.
* 4way TreeSet (4x4): 12 words, 0.11238159900000001 seconds, and 0.0018730266500000003 minutes.
* 8way TreeSet (4x4): 40 words, 0.7013623 seconds, and 0.011689371666666667 minutes.
* 8way ArrayList (2x2): 3 words, 0.056166600000000004 seconds, and 9.361100000000001E-4 minutes.
* 8way LinkedList (2x2): 3 words, 0.13891320100000001 seconds, and 0.002315220016666667 minutes.
* 4way HashSet (2x2): 1 word, 0.0027534 seconds, and 4.589E-5 minutes.
* 8way HashSet (2x2): 3 words, 0.002992401 seconds, and 4.987335E-5 minutes.
* 4way TreeSet (2x2): 1 word, 0.003201001 seconds, and 5.335001666666667E-5 minutes.
* 8way TreeSet (2x2): 3 words, 0.005669601000000001 seconds, and 9.449335000000002E-5 minutes.
* 8way ArrayList (2x3): 4 words, 0.1806694 seconds, and 0.003011156666666667 minutes.
* 8way LinkedList (2x3): 4 words, 0.5063062 seconds, and 0.008438436666666667 minutes.
* 4way HashSet (2x3): 2 words, 0.0057450010000000004 seconds and 9.575001666666667E-5 minutes.
* 8way HashSet (3x3): 4 words, 0.007907002 seconds, and 1.3178336666666666E-4 minutes.
* 4way TreeSet (2x3): 2 words, 0.0076157 seconds, and 1.2692833333333335E-4 minutes.
* 8way TreeSet (2x3): 4 words, 0.011237399 seconds, and 1.8728998333333334E-4 minutes.

There is quite a lot of data to analyze here. One thing that is consistent is that the amount of words found in each format is the same no matter which data structure is used. More words are found when going in 8 directions than when going in 4 directions, which is to be expected, as more combinations of letters are possible. The 8-way searches also took longer than the 4-way searches, which also isn’t terribly unexpected.

There’s quite a bit to analyze regarding the performance between each data structure. For ArrayLists, the time in seconds ranged from around 0.05 seconds to less than 50 seconds. For LinkedLists, the times ranged from 0.13 seconds to around 2.4 minutes. For HashSets, the times ranged from 0.0027 seconds to 1.8 minutes. For TreeSets, the times ranged from 0.0032 seconds to around 2.2 minutes. For ArrayLists and LinkedLists, there was no data taken for the 6-by-6 grid because doing so would take too long. There was data taken for the HashSets and TreeSets because they ended up not taking as much time for that grid. It seems like LinkedLists were the slowest data structures overall, with ArrayLists being faster than them in nearly every test. It seems like HashSets were the fastest data structures overall, with them having the shortest run times out of all of the data structures for each type of tests. If I were to rank them from fastest to slowest, HashSets would be first, followed by TreeSets, then ArrayLists, and finally LinkedLists.