**Program Structures & Algorithms**

**Fall 2021**

**Final Report - Group 18**

* **Team member:**

Danlin Lu (1567176)

Wenqi Yang (1586011)

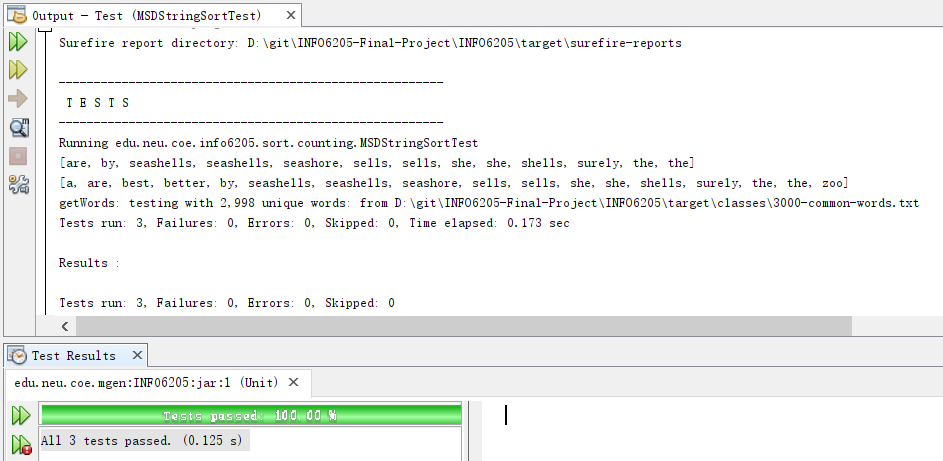
Kunhui Zhang (1563549)

* **Tasks**

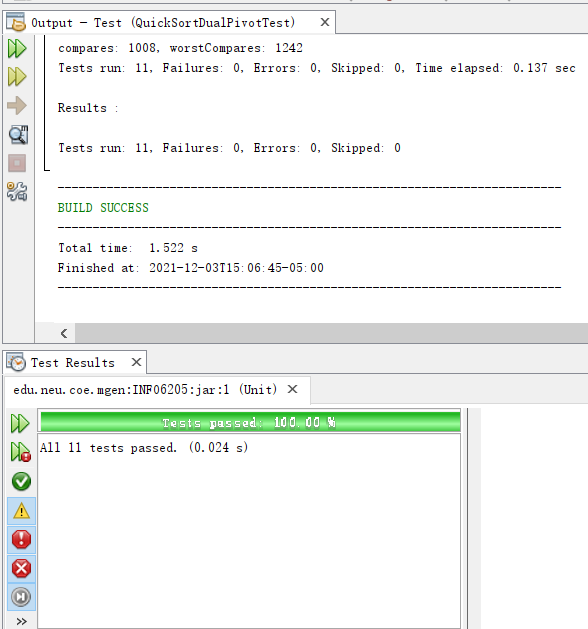
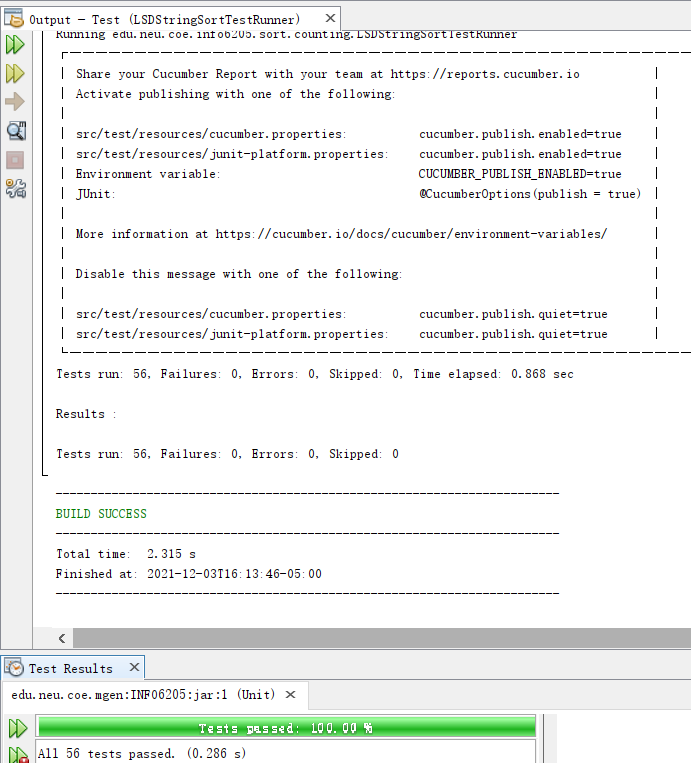
1. Implement MSD radix sort for Simplified Chinese which uses Unicode characters (UTF-8). (The conventional order for Chinese is according to the English order of the Pinyin.)
2. Compare MSD radix sort which we implemented with Timsort, Dual-pivot Quicksort, Huskysort, and LSD radix sort.
3. Show the trend of MSD radix sort’s running time under different data sizes. Show the compared result of five sorts on a graph.

* **Process (What we did)**

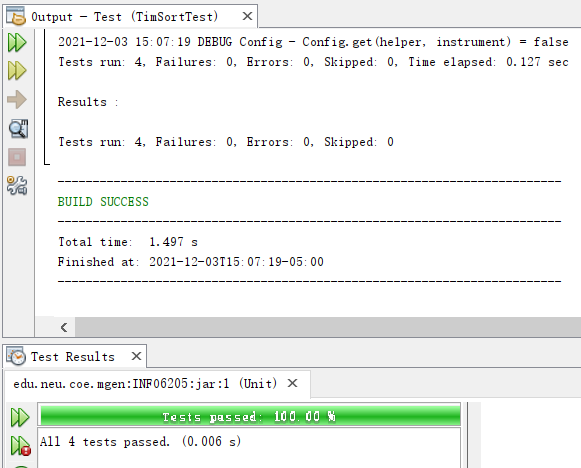
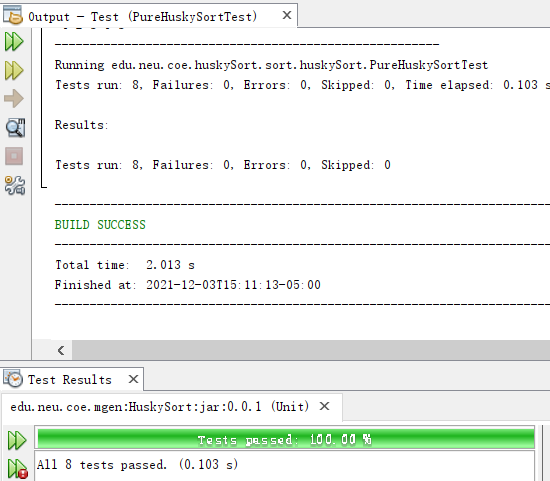
1. First, we used unit tests to test feasibility of these five sort methods, and made sure every method correct.



Picture 1. Unit tests of MSD radix sort



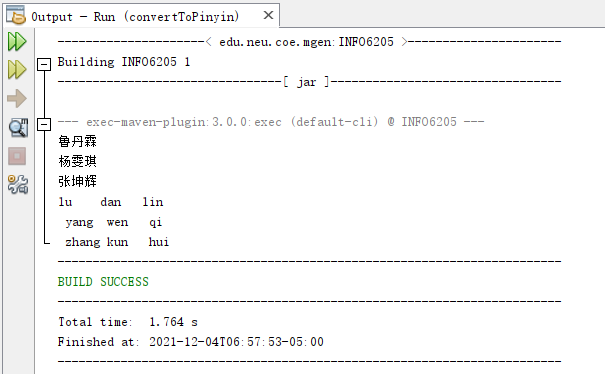
Picture 2. Unit tests of LSD radix sort Picture 3. Unit tests of QuickSort\_DualPivot



Picture 4. Unit tests of Pure HuskySort Picture 5. Unit tests of Tim Sort

The above unit tests mean that these sort methods were all available and usable.

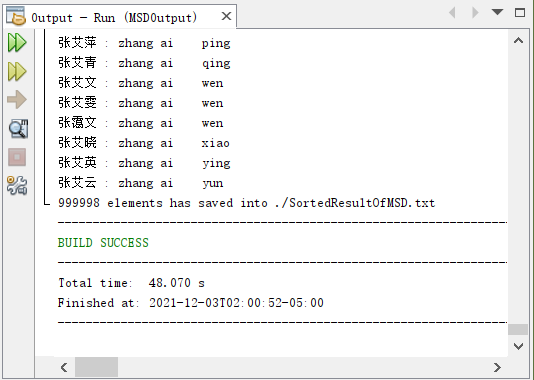
1. The second step is converting the Simplified Chinese into the Pinyin for be sorted by five sort methods. We imported the package called “pinyin4j-2.5.0” to help us convert the Simplified Chinese into the Pinyin.



Picture 6. Convert Chinese to Pinyin

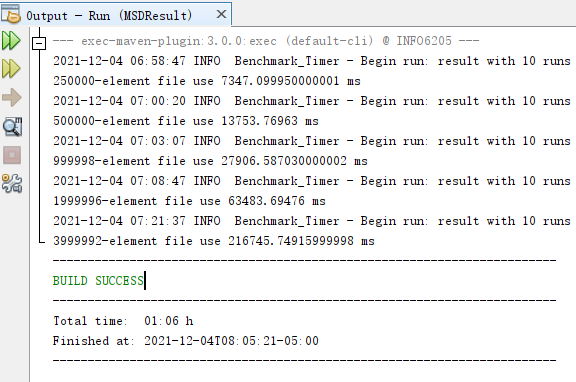
This means we converted the Simplified Chinese into Pinyin successfully.

1. Then, it is necessary that we should check the sorted results before continuing next task. Therefore, we printed the all sorted results and output them into txt files. Here is outputs under 1M size by MSD radix sort method: (you could see outputs of 1000 sizes by all sort methods in repository)



Picture 7. Output of MSD radix sort

1. Last but not least, we benchmarked the results of all methods(MSD radix sort, Timsort, Dual-pivot Quicksort, Huskysort, and LSD radix sort) for 250k, 500k and 1M, 2M, 4M Chinese names.(Here we only show the result of MSD radix sort, other sort methods’ result we will upload in repository)

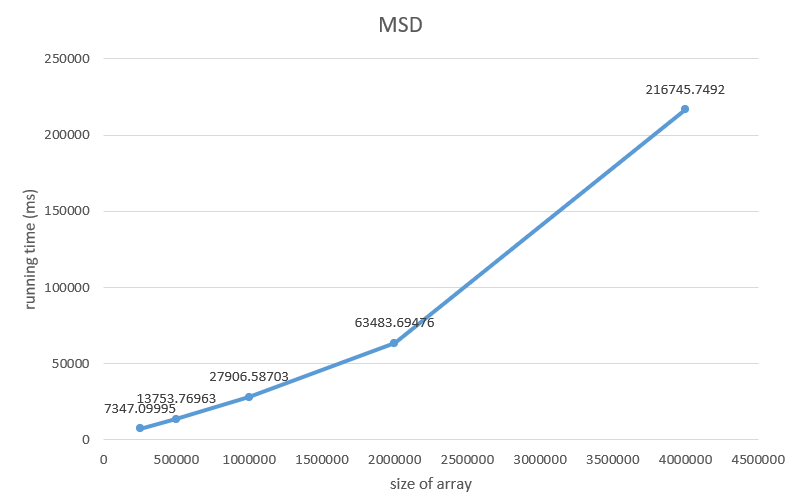


Picture 8. Result of MSD radix sort

With the results, we could draw the graphs to observe the growth of sort methods.

* **Conclusion:**

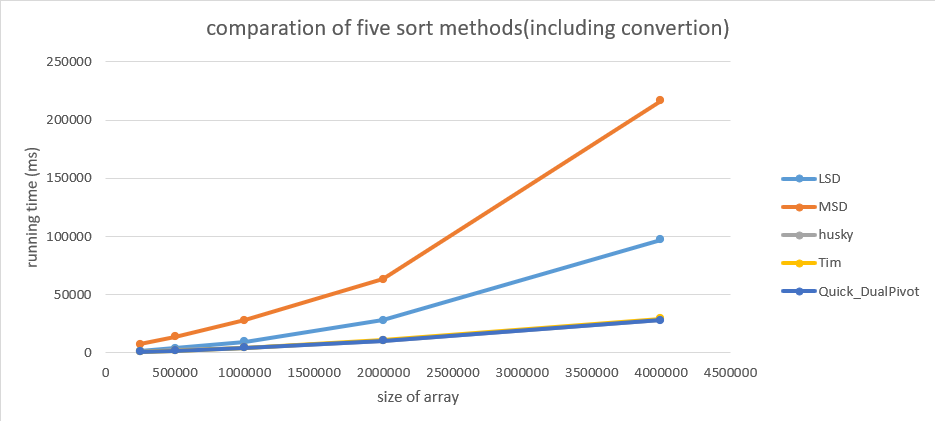
1. Before comparing sort methods, let’s observe the growth of MSD radix sort method:



Picture 9. Growth of MSD radix sort

From the growth, the increasing speed of running time is faster than that of size, therefore, the growth rate of MSD is gradually getting faster.

1. This is the graph of comparation among MSD radix sort, Timsort, Dual-pivot Quicksort, Huskysort, and LSD radix sort:



Picture 10. Comparation of five sort methods

According to above graph, we could find that the MSD spent most time to sort, and much more than other methods. Same as radix sort, LSD took less time than MSD. Compared with MSD, the HuskySort, TimSort and QuickSort\_DualPiovt nearly spent same running time.