

# Final Paper

## Different Sorts and Radix Sort

001582498 Xingyu Huang

001528318 Zhen Luo

001048178 Haoyuan Qin

### 1. Abstract:

Unlike English letters which are encoded by ASCII, Chinese characters normally use Unicode to encode. Hence, how to sort Chinese characters in order, became a real problem we may face in the real world. So, in this paper, we will mainly discuss different Radix sorts and give a conclusion that why we choose radix sort as a sorting algorithm in sorting Chinese Characters.

### 2. Difference Between Algorithms

There are a lot of different sorting algorithms in the world, like insertion sort, selection sort, quicksort and so on. However, some sorting algorithms are not suitable for their time consuming and space consuming. Here we'd like to talk about some fast sort algorithms: timsort, husky sort, quicksort and radix sort.

## 2.1 Time Consumption

Normally, these fast-sorting algorithms' time consumption is  $N \lg N$ .  $N$  stands for the number of items. For timsort, the average time consumption is  $N \lg N$ , and for quicksort it's  $2N \lg N$  and for Husky sort it is  $N \lg N$  as well. (Hillyard) However, the quicksort's time consumption will be decreased if quicksort has more pivots, and the best is quicksort with a median of three pivots. (Lakshmi) For example Dual-pivot quicksort is faster than normal quicksort, and quicksort with three pivots can be faster than Dual-pivot as well. However, when it comes to MSD radix sort and LSD radix sort the average case is  $NL$ .  $L$  stands for the number of digits. So, radix sort's speed is depending on the length of inputted items.

## 3. Radix Sort

When we use radix sort, we will compare digits instead of items and before we use LSD radix sort, we must make sure that the length of different items should be the same. What's more, like we can improve the performance of quicksort by increasing pivots, we can also make some improvement on radix sorts as we discuss below.

### 3.1 Different Kinds of Radix Sort

Beside MSD and LSD radix sort there are also other radix sort algorithms and different usage of radix sorts. For example, CC radix sort and Fast radix. CC stands for Cache Conscious which can improve the performance as well as fast radix sort. CC Radix and MSD Radix performed well in broad uniform and normal distributions, in the case of MSD Radix, performing exceptionally well, but with catastrophic performance on narrower distributions. Fast Radix consistently outperformed LSD Radix in all cases, and its improvement was reliably apparent given input sizes of 1000. Fast Radix's performance gains over LSD Radix Sort were in the 4–8% range. (S Thiel)

### 3.2 The Reason to Choose Radix Sort

Chinese characters do not have a standard sorting rule. Hence, we'd like to choose 'pinyin' sequence as the rule of sorting.

Because radix sort can sort item based on digits, as 'pinyin' is made up by digits. We do believe radix sort can sort Chinese characters in 'pinyin' order.

#### 4. Conclusion

Unlike other sorting algorithms, radix sorts focus on digits. So now we can understand that the radix sort can be powerful in some cases like average length of digits of items are not big. And if the import length of digits is not the same. We have to convert them into same length before using LSD radix sort, which will cost some time. And with the help of Fast Radix, we can improve our performance. When it comes to sort Chinese characters, we can implement radix sort to finish our task.

## 5. References

Lakshmi, D. I. (2018). Performance analysis of four different types of sorting algorithms using different languages. *International Journal of Trend in Scientific Research and Development*, Volume-2(Issue-2), 535–541.

<https://doi.org/10.31142/ijtsrd9464>

Wild, S., Schneider, K., Nebel, M., Sedgewick, R., & Dietzfelbinger, M. (2016). *Dual-pivot quicksort and beyond: Analysis of multiway partitioning and its practical potential*. Technischen Universität Kaiserslautern.

Shukla, A., Saxena, A.K., Baruah, P.K., Bandyopadhyay, S., Sahni, S., Jiménez-González, D., Navarro, J.J., Larrba-Pey, J., Cederman, D., & Tsigas, P. (2012). ' Review of Radix Sort & Proposed Modified Radix Sort for Heterogeneous Data Set in Distributed Computing Environment '.

Relaxing the counting requirement for least significant digit radix sorts. PDF Free Download. (n.d.). Retrieved December 5, 2021, from <https://docplayer.net/197070654-Relaxing-the-counting-requirement-for-least-significant-digit-radix-sorts.html>.

Hillyard, Robin & Liao Zheng, Yunlu & R, Sai. (2020). Huskysort.