Handout Bucket Sort

Group members: Cuiting Huang, Zhe Cheng, Hongyang Ye

Authors affiliation: Northeastern University

Country location: China

E-mail address: [huang.cui@northeastern.edu](mailto:huang.cui@northeastern.edu)

...

...

III、Bucket sort

To guarantee the time complexity O(n) of bucket sort

*A. Assumption of bucket sort*

The data set is generated by a random process, which distributes elements independently over the interval.

Only the elements are as evenly distributed as possible, the time complexity of bucket sort would be O(n)

*B. Basic Idea*

What’s bucket? A bucket is a container with a certain volume, bucket represents an interval.

*1) First step:* Creating buckets and determining the range of each bucket. There are many different ways to determine the number of buckets. One rule we can take into account is that it is best to have elements evenly distributed in each bucket. Here we will introduce two buckets splitting principle.

*a) Simple Bucket Splitting:*

*b) Reduced Bucket Splitting:* If the range of the data set is large, with using simple buckets splitting, bucketNum will be extremely overflowing, there will be a bunch of empty buckets in the middle.

*2) Second step:* Traverse the data set and place the elements into each corresponding bucket. Here we will use a mapping function, each element will be mapped to the nth bucket. The mapping function the key of making our algorithm more efficient.

*3) Third Step:* Sort the elements inside each bucket separately. You can use whatever comparison sort you like

*4) Last Step:*Traverse over all buckets which are not empty and put all elements back to the original data set in turn

Notice that, bucket sort is the algorithm out-of-place, to avoid wasting space, the number of buckets should be just enough, neither overflowing nor too little

*C**. Sorting character strings*

In addition to the numerical array, bucket sort can be used dealing with String array. For example, there is a string array with lowercase letters uppercase letters and digits.

All lowercase letters are required to precede uppercase letters, but no order is required between themself. Then put the digits behind the letters. By using bucket sort, we can separate upper and lower case letters and digits.

First, traverse the array, count the number of elements of three types

Next, calculate the starting index of each type

Then, according to the index interval, put elements into the corresponding bucket, the interval of each bucket is

Last, write back to the original array. Done!

*D. Bonus*

Animation Bucket\_sort <https://youtu.be/vt1YX_ndHMk>

Animation Bucket\_sort for string array <https://youtu.be/Ggf4NUe7bCg>