**Sorting algorithm In Linear Time (Contd…)**

**Bucket Sort:**

Bucket sort assumes that the input is generated by a random process and drawn from a uniform distribution. In other words the elements are distributed uniformly and independently over the interval [0,1]. Bucket sort divides the interval [0,1] into n equal sized subintervals or buckets. Then distributes the n inputs into these buckets.

After that the elements of each buckets are sorted using a sorting algorithm generally using insertion or quick sort.

Finally the buckets are concatenated together in order.

Consider that the input is an n-element array A and each element A[i] in the array satisfies the 0<=A[i]<1

**Bucket Sort Algorithm:**

Bucket-Sort(A)

1.Let B[0….n-1] be a new array

2.n = length[A]

3.for i = 0 to n-1

4.     make B[i] an empty list

5.for i = 1 to n

6.    do insert A[i] into list B[└ n A[i] ┘]

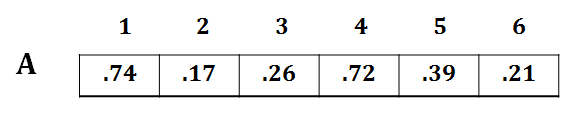
7.for i = 0 to n-1

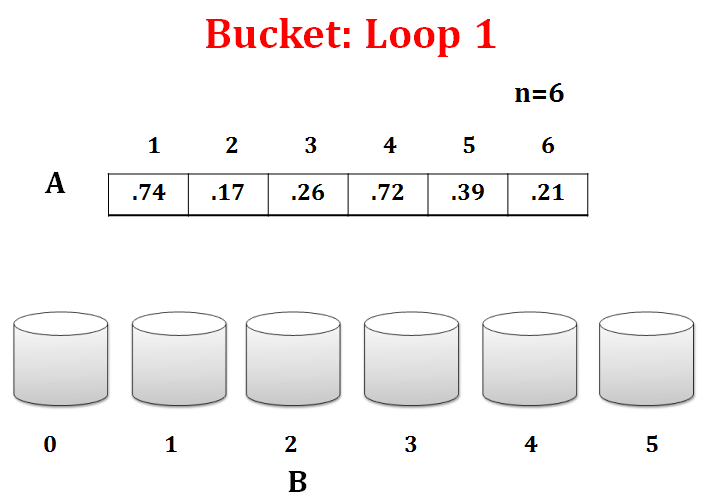
8.   do sort list B[i] with Insertion-Sort

9.Concatenate lists B[0], B[1],…,B[n-1] together in order

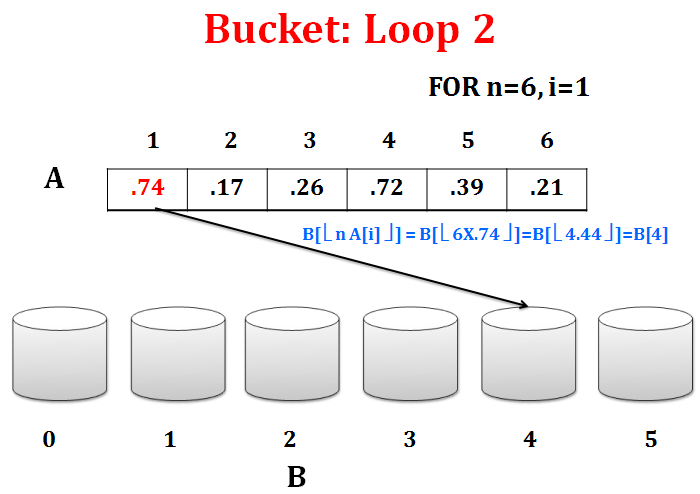
Example:

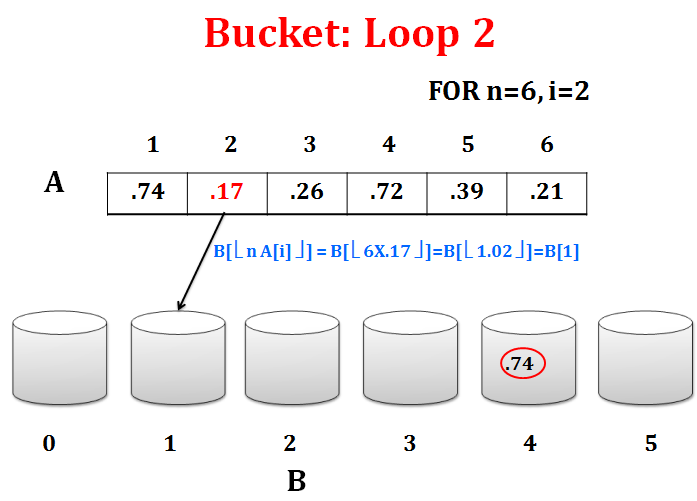
1)



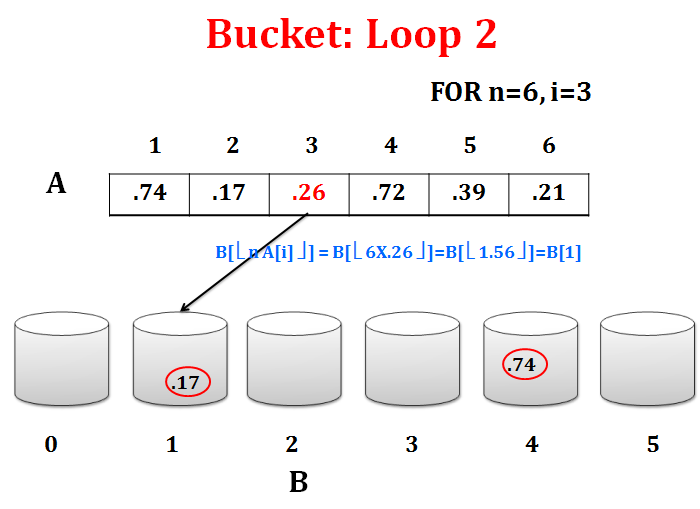
2) 

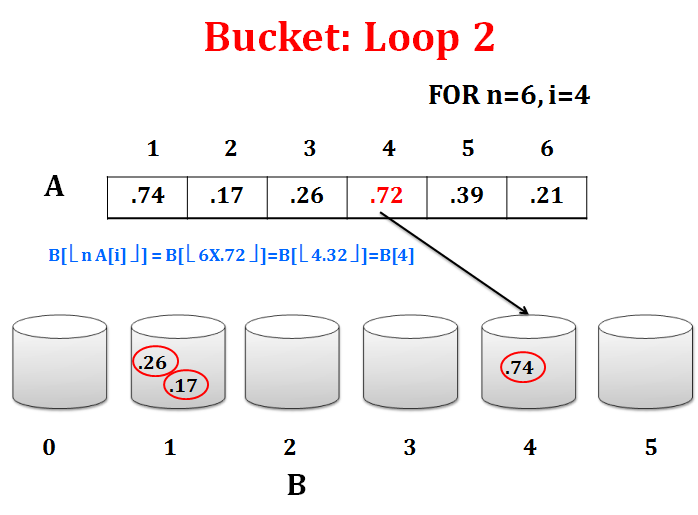
3)



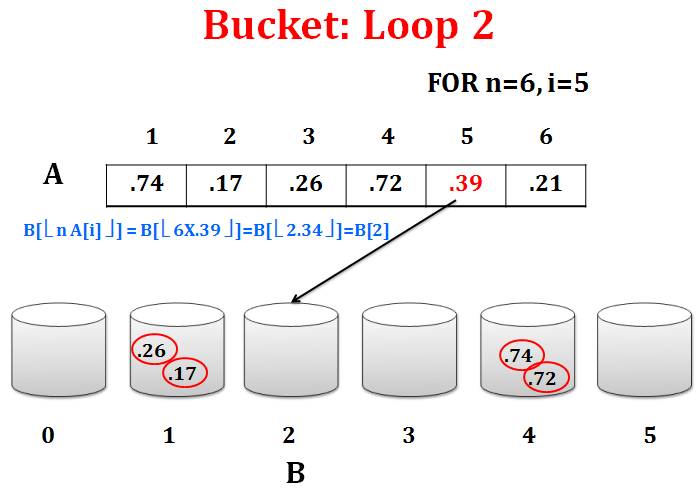
4) 

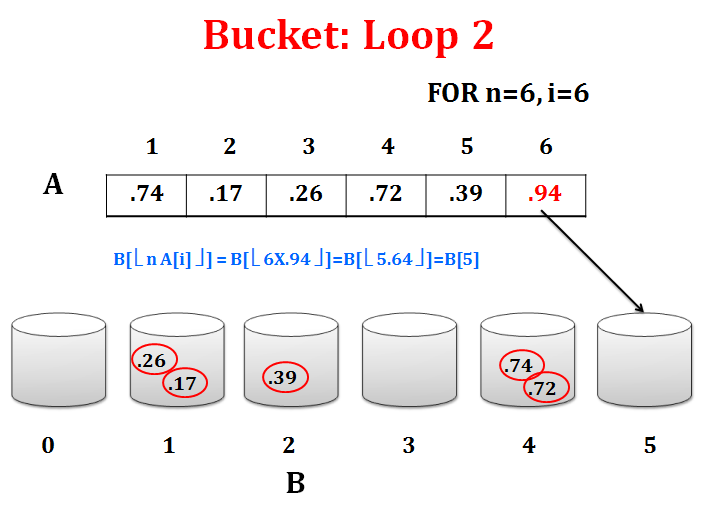
5)



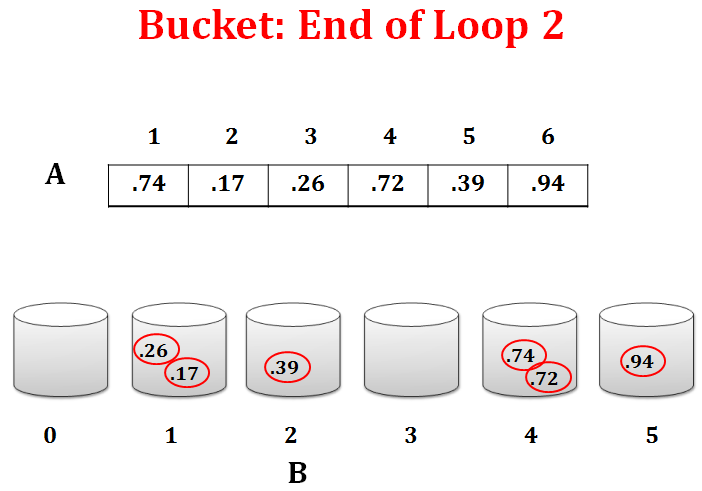
6) 

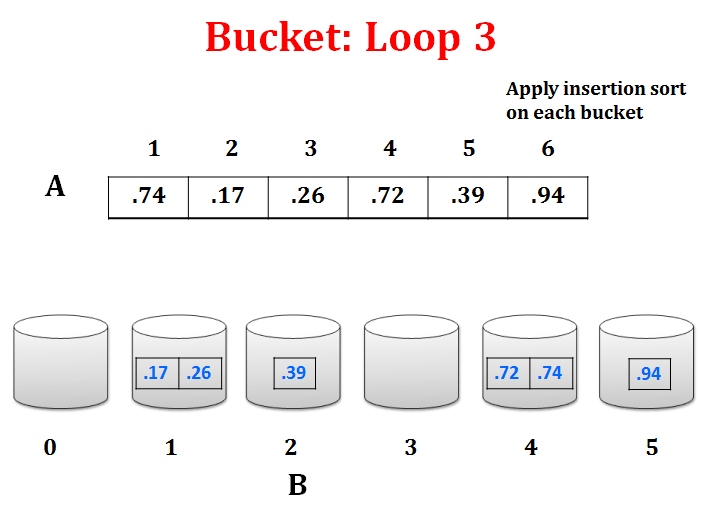
7)



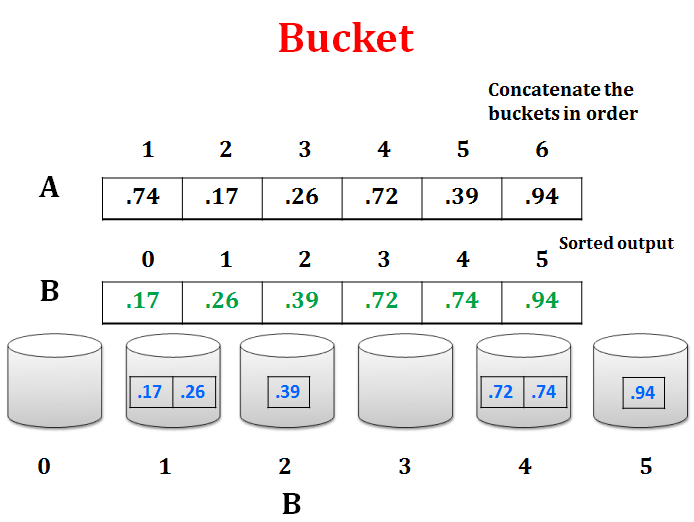
8) 

9)



10) 

11)



**Bucket Sort Review:**

**Assumption:** input is uniformly distributed across a range

**•Basic idea:**

–Partition the range into a fixed number of buckets.

–Toss each element into its appropriate bucket.

–Sort each bucket.

**•Pro’s:**

–Fast

–Asymptotically fast (i.e., O(*n*) when distribution is uniform)

–Simple to code

–Good for a rough sort.

**•Con’s:**

–Doesn’t sort in place

**RELEVANT READING MATERIAL AND REFERENCES:**

**Source Notes:**

1. [https://www.geeksforgeeks.org › bucket-sort-2](https://d2aqwpvls1cndj.cloudfront.net/ultra/uiv3800.11.0-rel.10_e9cd3c0" \t "_blank)
2. <https://www.slideshare.net/shimulsakhawat/bucket-sort>

**Lecture Video:**

1. <https://youtu.be/VuXbEb5ywrU>

**Online Notes:**

1. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos/MIT6_006F11_lec07.pdf>

**Text Book Reading:**

1. Cormen, Leiserson, Rivest, Stein, “*Introduction to Algorithms*”, Prentice Hall of India, 3rd edition 2012. problem, Graph coloring.

**In addition: PPT can be also be given.**Bottom of Form