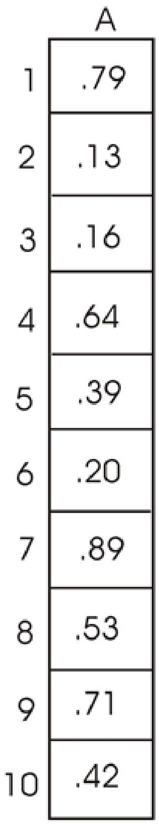
获得的答案

返回

Operation of BUCKET-SORT for n=10:

Consider the input array A[1..10] as shown below:

$A \ = \ < .79, .13, .16, .64, .39, .20, 89, .53, .71, .42 >$



	A
1	.79
2	.13
3	.16
4	.64
5	.39
6	.20
7	.89
8	.53
9	.71
10	.42

The internal [0,1) is divided into 10 equal sized buckets such that bucket *i* contains values that fall in the half-open range [*i*/10,(*i*+1)/10].

- Bucket 0 will hold the values that fall in the half-open range [0, 0.1]. Bucket 1 will hold the values that are $\geq 0 \text{and} < 0.1$.

- Bucket 1 will hold the values that fall in the half-open range [0.1, 0.2]. Bucket 1 will hold the values that are $\geq 0.1 \text{and} < 0.2$.

- Bucket 2 will hold the values that fall in the half-open range [0.2, 0.3]. Bucket 2 will hold the values that are $\geq 0.2 \text{and} < 0.3$.

- Bucket 3 will hold the values that fall in the half-open range [0.3, 0.4]. Bucket 3 will hold the values that are $\geq 0.3 \text{and} < 0.4$.

- Bucket 4 will hold the values that fall in the half-open range [0.4, 0.5]. Bucket 4 will hold the values that are $\geq 0.4 \text{and} < 0.5$.

- Bucket 5 will hold the values that fall in the half-open range [0.5, 0.6]. Bucket 5 will hold the values that are $\geq 0.5 \text{and} < 0.6$.

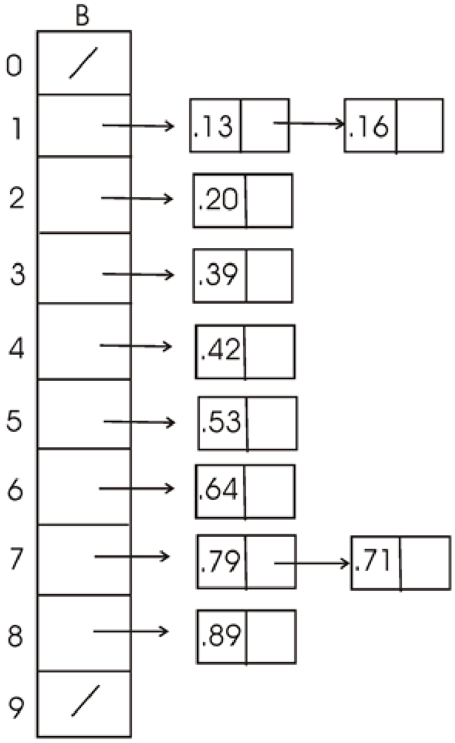
- Bucket 6 will hold the values that fall in the half-open range [0.6, 0.7]. Bucket 6 will hold the values that are $\geq 0.6 \text{and} < 0.7$.

- Bucket 7 will hold the values that fall in the half-open range [0.7, 0.8]. Bucket 7 will hold the values that are $\geq 0.7 \text{and} < 0.8$.

- Bucket 8 will hold the values that fall in the half-open range [0.8, 0.9]. Bucket 8 will hold the values that are $\geq 0.8 \text{and} < 0.9$.

- Bucket 9 will hold the values that fall in the half-open range [0.9, 1]. Bucket 9 will hold the values that are $\geq 0.9 \text{and} < 1$.

According to the lines, 5 and 6 of the algorithm BUCKET-SORT(A) given in the textbook, the buckets will contain the values as shown below:



Explanation:

- The value .79 is greater than 0.7 and less than 0.8. Hence, it falls into the bucket 7.

- The values .13 is greater than 0.1 and less than 0.2. Hence, it falls into the bucket 1.

- The values .16 is greater than 0.1 and less than 0.2. Hence, it falls into the bucket 1.

- The value .64 is greater than 0.6 and less than 0.7. Hence, it falls into the bucket 6.

- The value .39 is greater than 0.3 and less than 0.4. Hence, it falls into the bucket 3.

- The value .20 is equal to 0.2. Hence, it falls into the bucket 2.

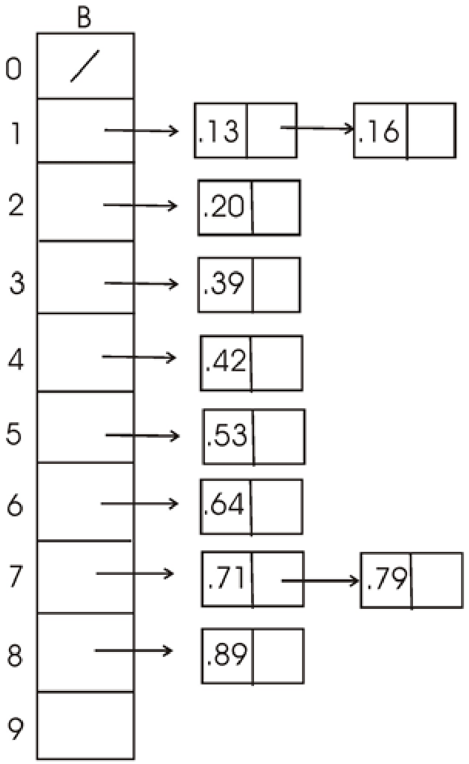
- The value .89 is greater than 0.8 and less than 0.9. Hence, it falls into the bucket 8.

- The value .53 is greater than 0.5 and less than 0.6. Hence, it falls into the bucket 5.

- The value .71 is greater than 0.7 and less than 0.8. Hence, it falls into the bucket 7.

- The value .42 is greater than 0.4 and less than 0.5. Hence, it falls into the bucket 4.

According to the lines 7 and 8 of the algorithm BUCKET-SORT(A) given in the textbook, the buckets will contain the values as shown below:



Explanation:

- All the buckets contain only one value expect the buckets 1 and 7.

- The values of bucket 1 are already sorted.

- The values of bucket 7 are not sorted. So sort the values of bucket 7 using insertion sort.

According to the line 9 of the algorithm BUCKET-SORT(A) given in the textbook, concatenate the lists B[0], B[1],B[2],..,B[9] in order.

Then the array B is as shown below:

	B
0	.13
1	.16
2	.20
3	.39
4	.42
5	.53
6	.64
7	.71
8	.79
9	.89

Hence, the sorted array is $B = < .13,\ .16,\ .20,\ .39,\ .42,\ .53,\ .64,\ .71,\ .79,\ .89 >$.