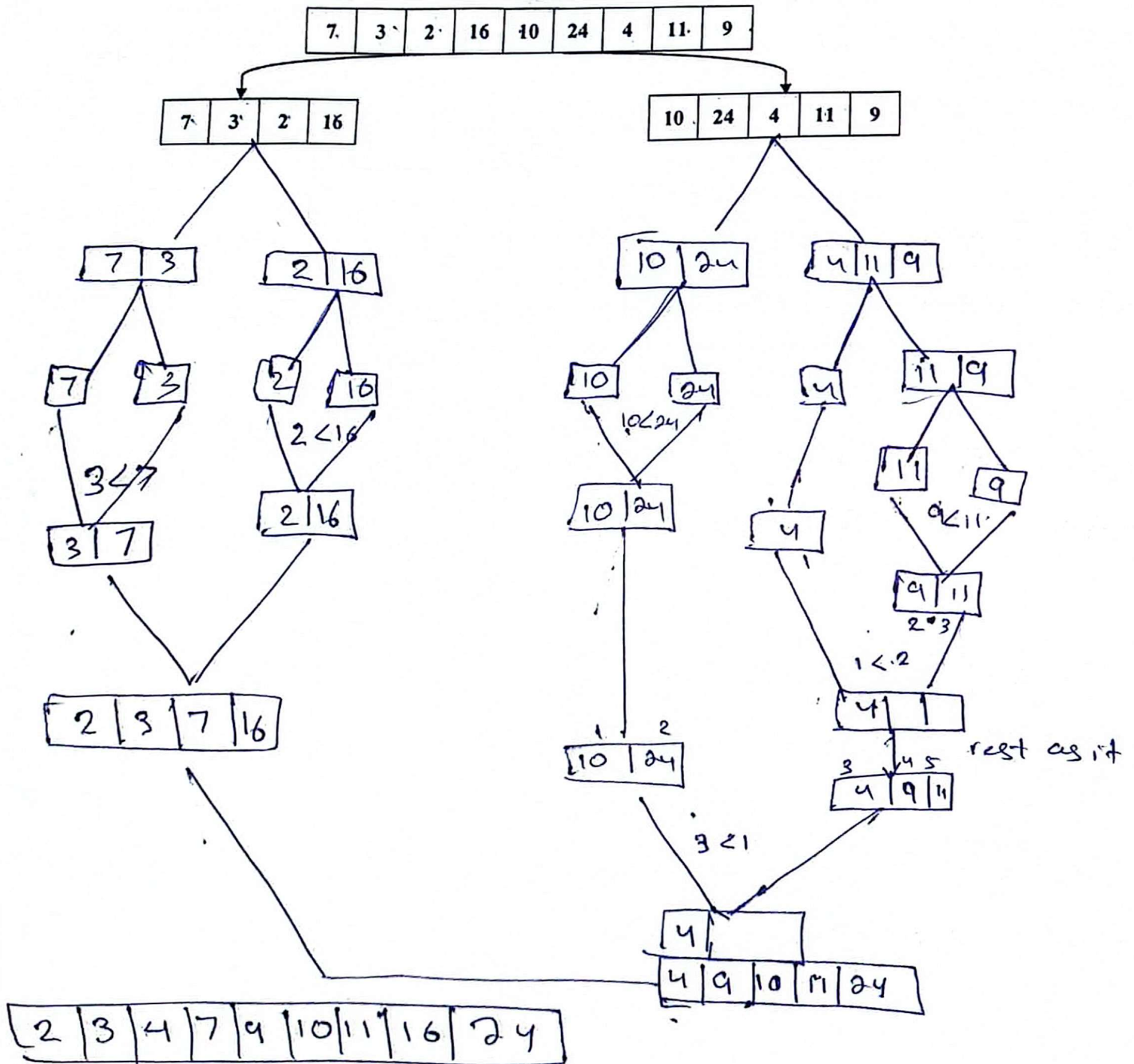**Q1: Show the state of the array at each step of sorting. Sort the following array by using** **[5]**
Merge Sort. highlight the changes as shown in example

7.	3`	2·	16	10	24	4	11.	9

7`	3`	2	16

10.	24	4	11	9

7	3

2	16

10	24

4	11	9

7		3

2	16

10		24

10<24

4

11	9

3<7

2<16

10	24

4

11

9

9<11

3	7

2	16

9	11
2·3

1<·2

2	3	7	16

1	2
10	24

4	1

rest as it

3<1

3	4 5
4	9	11

4	

4	9	10	11	24

2	3	4	7	9	10	11	16	24

Q2: What is the Time complexity of the following   *[3]*

```for (i =1; i<=n*n*n(i +=2) {```
 ```   cout<<i ;```
 ```   sum = 0 ;```
 ```   for (int j =1; j<=i/; j++) {```
 ```   sum++;```
 ```   cout<<i ;```
 ```   }```
 ```   for (int k=1; k<=n ; k++) {```
 ```   sum++;```
 ```   cout<<i ;```
 ```   }```
 ```   cout<<sum;```
 ```}```
	$n^3 \log n \,/\, n^2 \log n$
	 $n^3 , n^2 \cdot n$
	
	 n
	
	 $n^6 .$
	 $(n^2 \log n + n)$

i^2

Q3: Answer the following   *[2]*

1. What is the time complexity of the Merge Sort algorithm in the worst case?

A) O(n)
✓B) O(n log n)
C) O(n^2)
D) O(log n)

2. Which of the following statements is **not true** about the Merge Sort algorithm?

A) Merge Sort is a stable sorting algorithm.
B) Merge Sort has a space complexity of O(n)..
C) Merge Sort requires additional space proportional to the size of the input array.
✓D) Merge Sort is an in-place sorting algorithm.