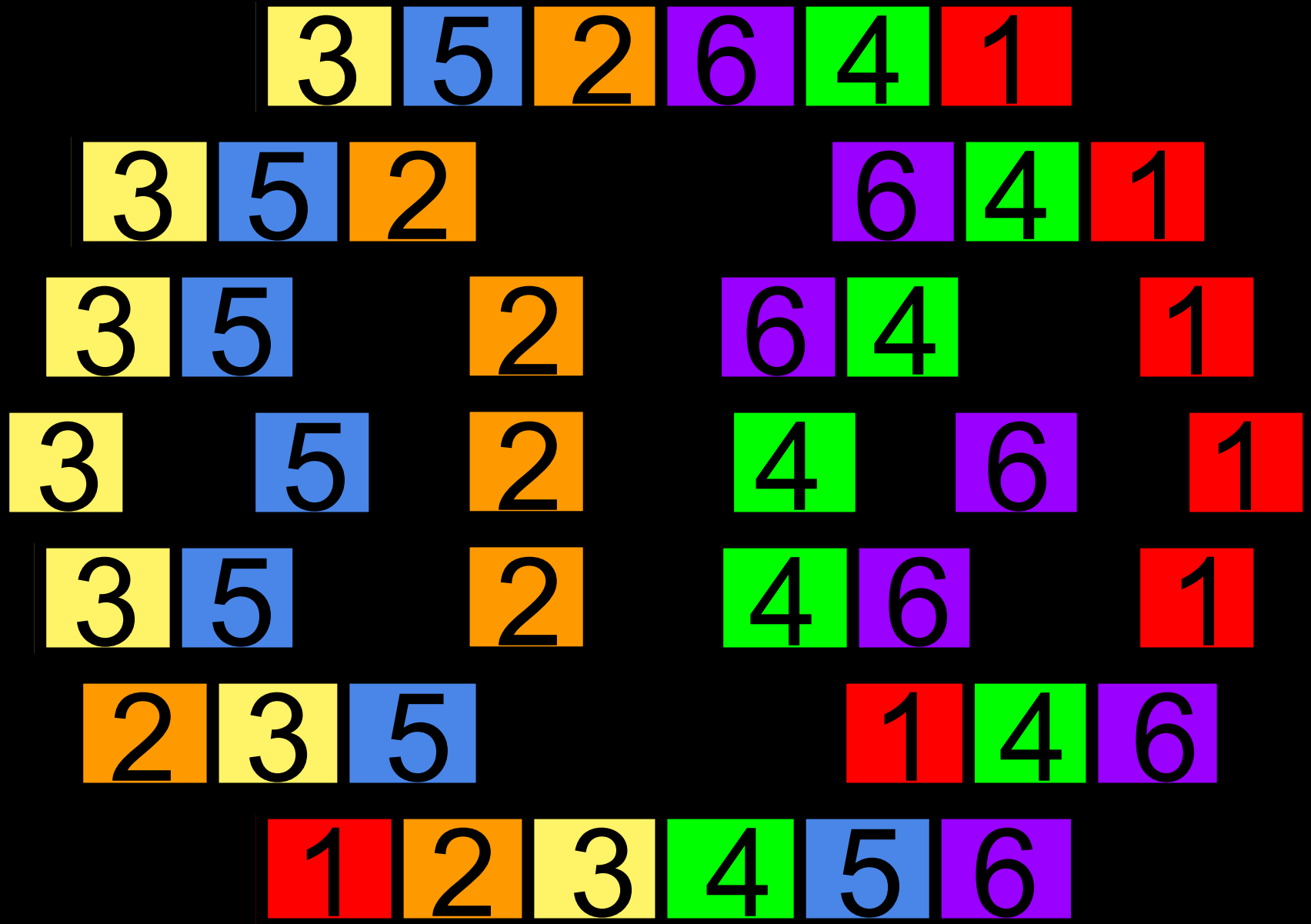
Merge Sort



On input of n elements:

If n < 2

Return.

Else

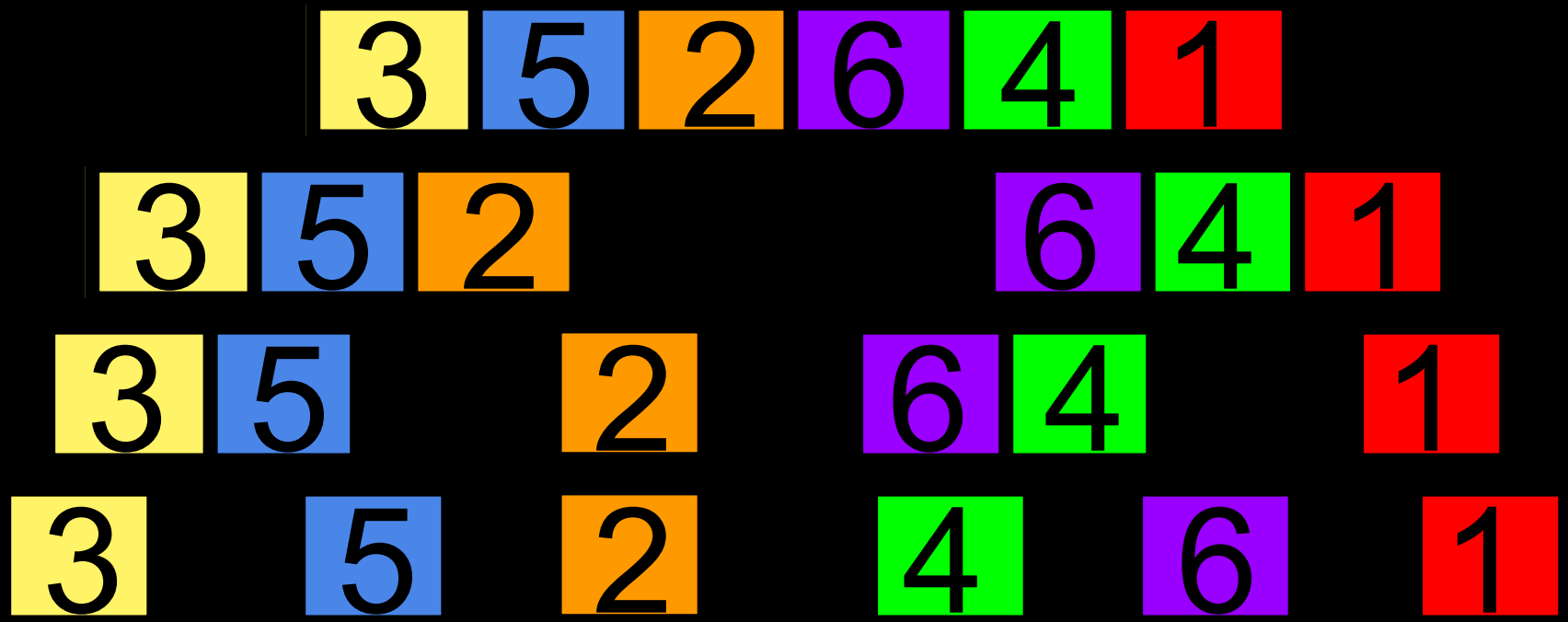
Sort left half of elements.

Sort right half of elements.

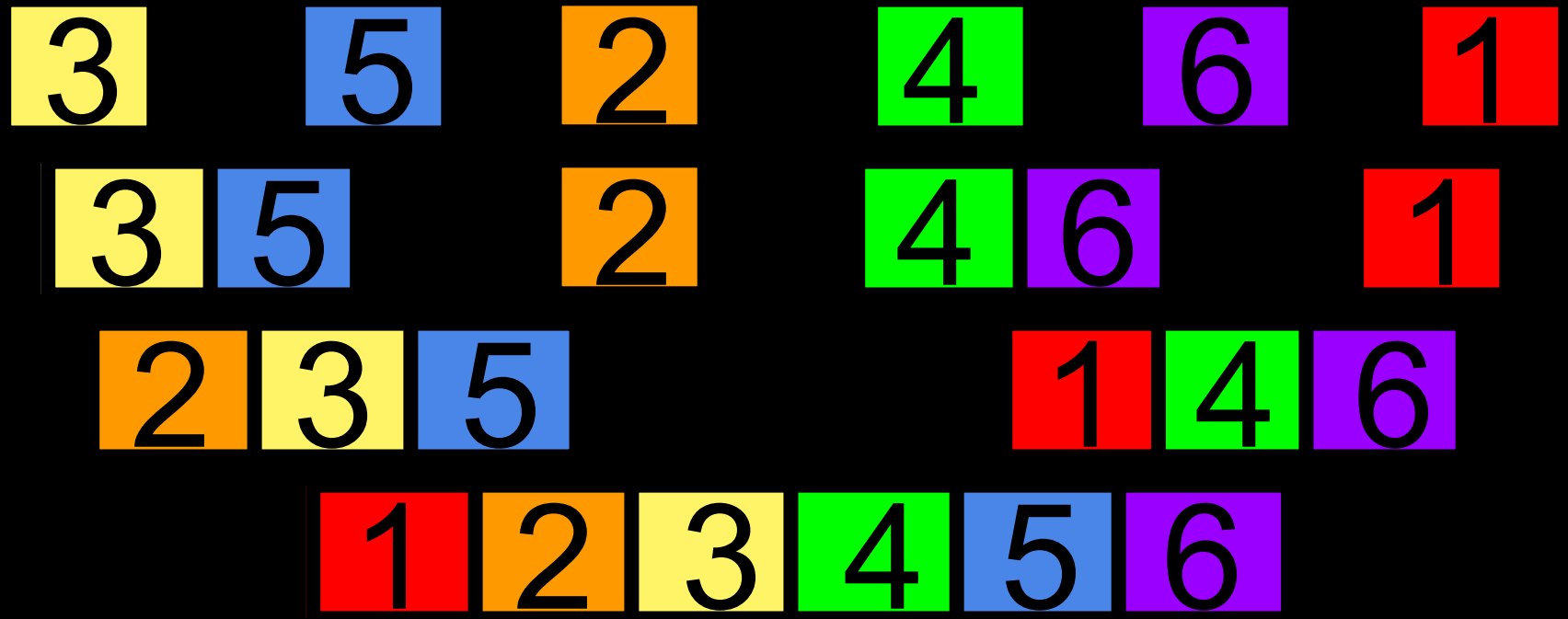
Merge sorted halves.


3 5 2 6 4 1

Halve until each subarray is size 1



Merge Sorted Halves

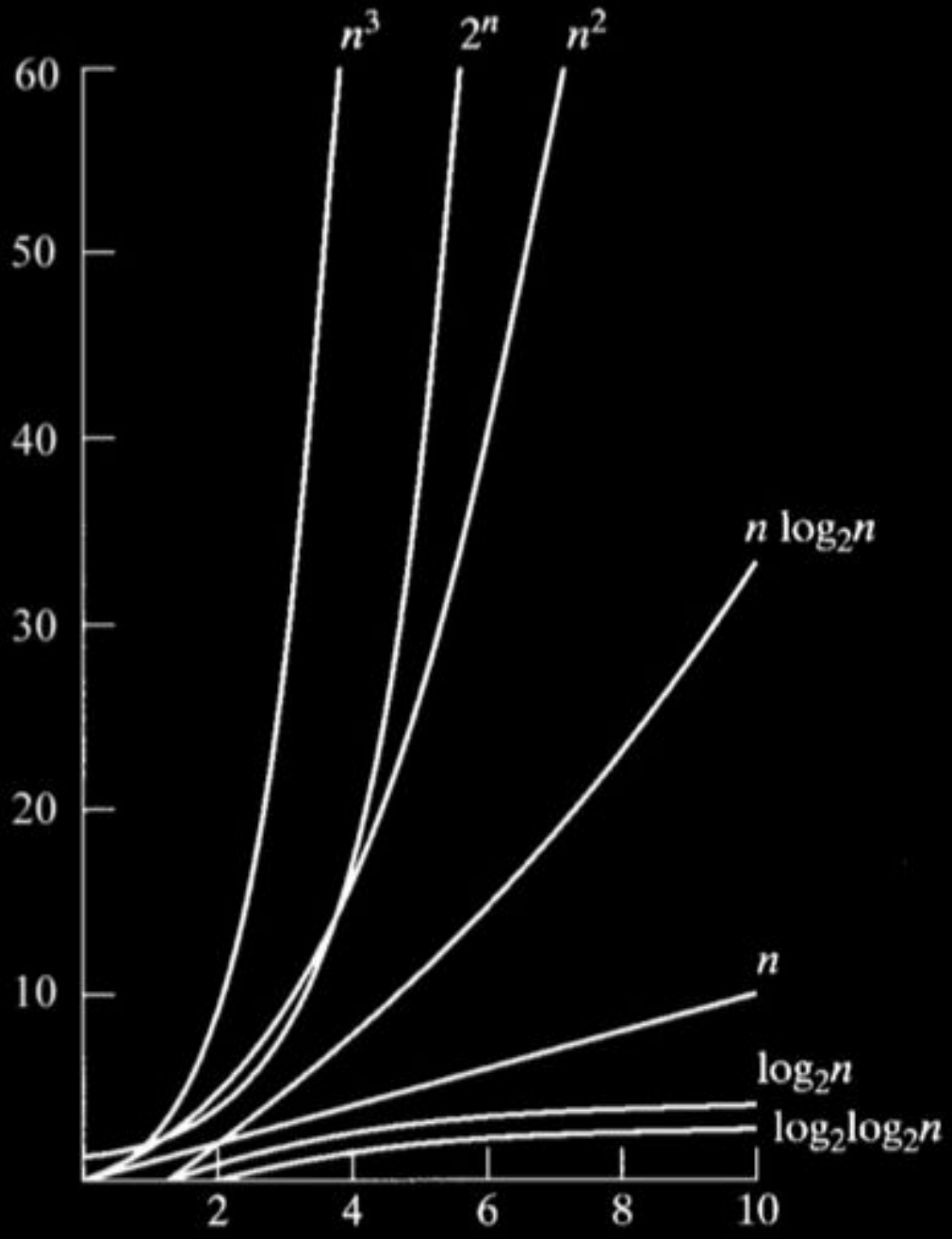


```
sort (int array[], int start, int end)
{
   if (end > start)
     {
         int middle = (start + end) / 2;

         sort(array, start, middle);
         sort(array, middle + 1, end);

         merge(array, start, middle, middle + 1, end);
     }
}
```

What's the best case runtime of merge sort?

What's the worst case runtime of merge sort?

What's the expected runtime of merge sort?


n^3
2^n
n^2
60
50
40
30
20
10
$n \log_2 n$
n
$\log_2 n$
$\log_2 \log_2 n$
2
4
6
8
10

	Bubble Sort	Selection Sort	Insertion Sort	Merge Sort
O	n^2	n^2	n^2	$nlogn$
Ω	n	n^2	n	$nlogn$
Θ		n^2		$nlogn$