Algorithm

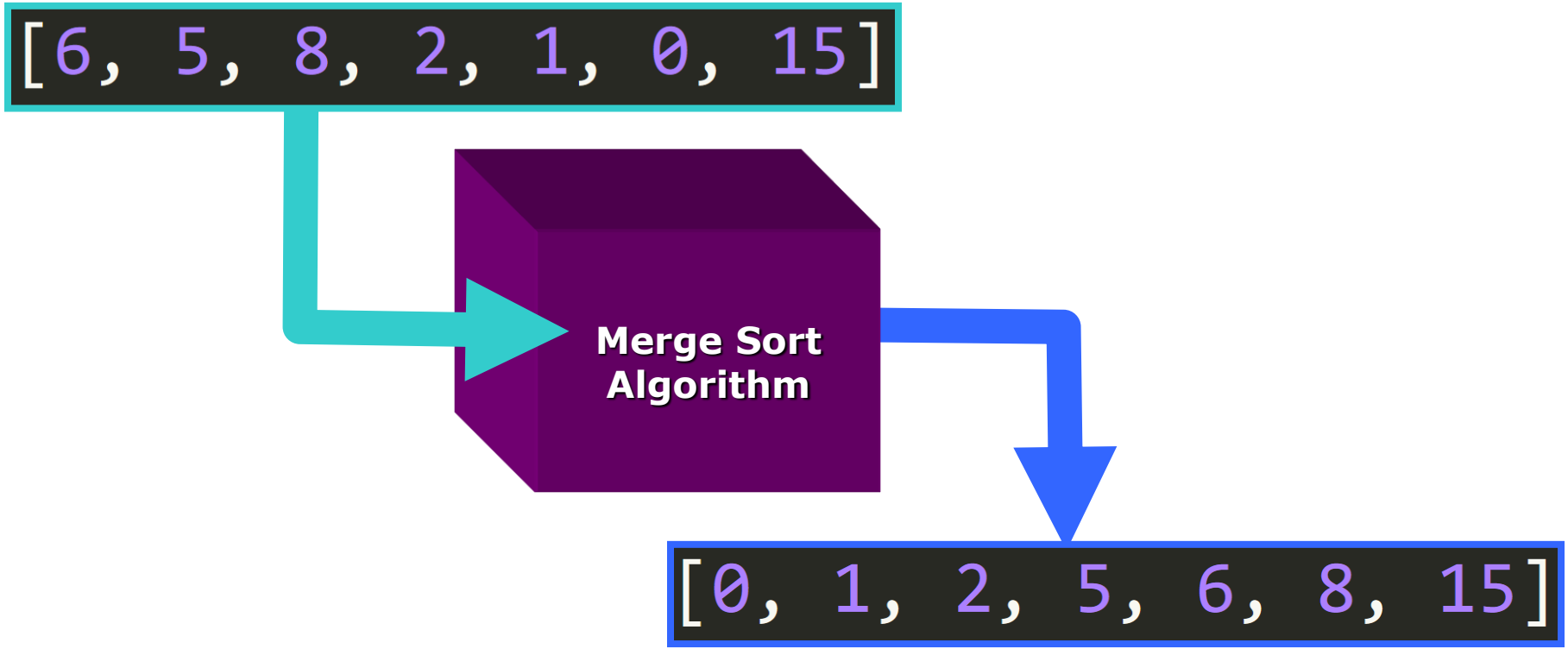
Merge Sort

Visual Walkthrough

Merge Sort

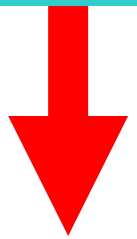
- Divide and Conquer Algorithm (recursion).
- Divides the list into **two halves.**
- Calls itself recursively for each half (left and right) until they only contain one element.
- Merges the two halves.
- Very efficient for large sequential data (such as lists).
- Not in-place sorting algorithm. Requires additional space.

Merge Sort



Merge Sort

```
[6, 5, 8, 2, 1, 0, 15]
```



Sort

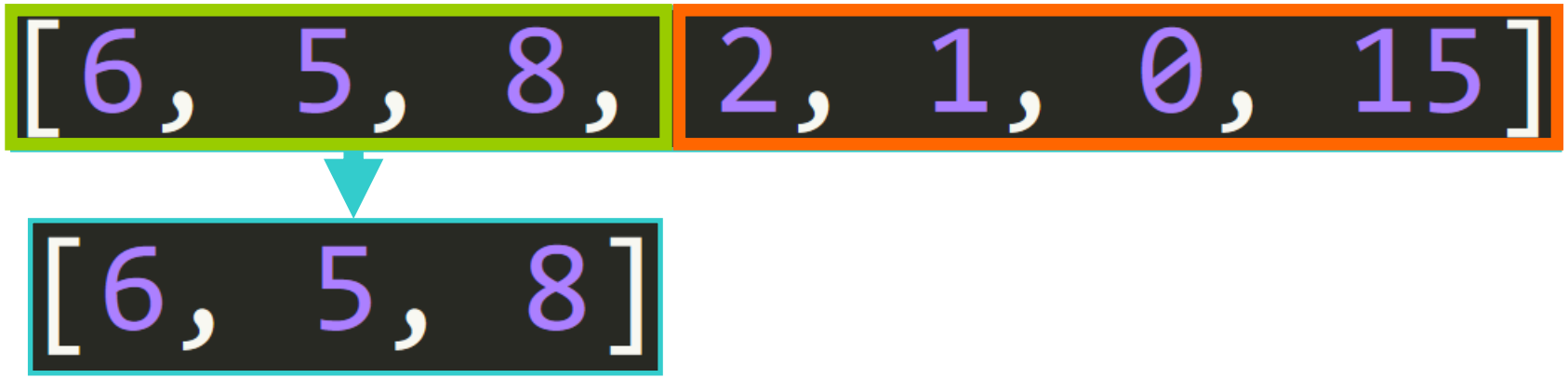
```
[0, 1, 2, 5, 6, 8, 15]
```

Merge Sort

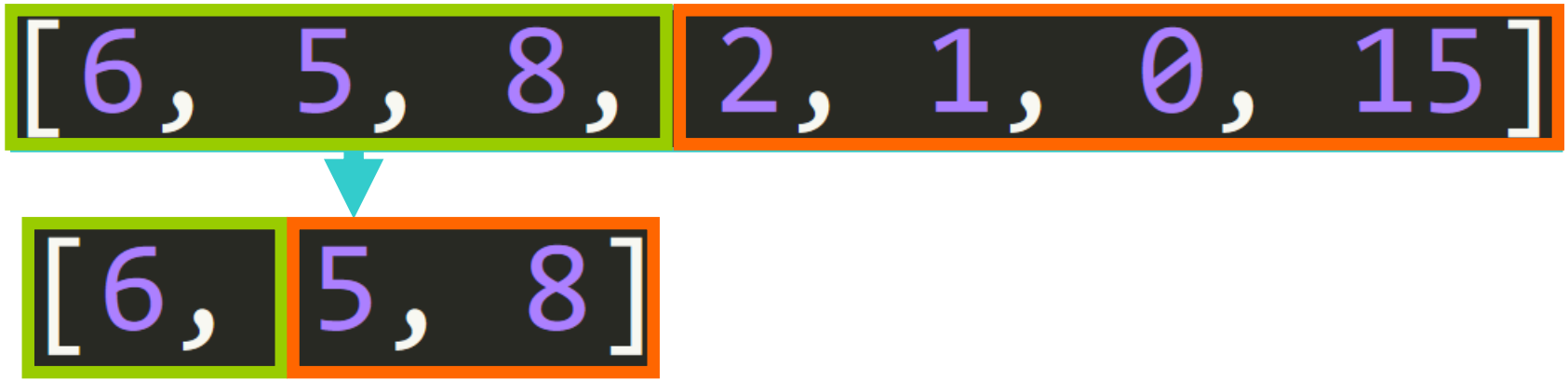
```
[6, 5, 8, 2, 1, 0, 15]
```

Merge Sort

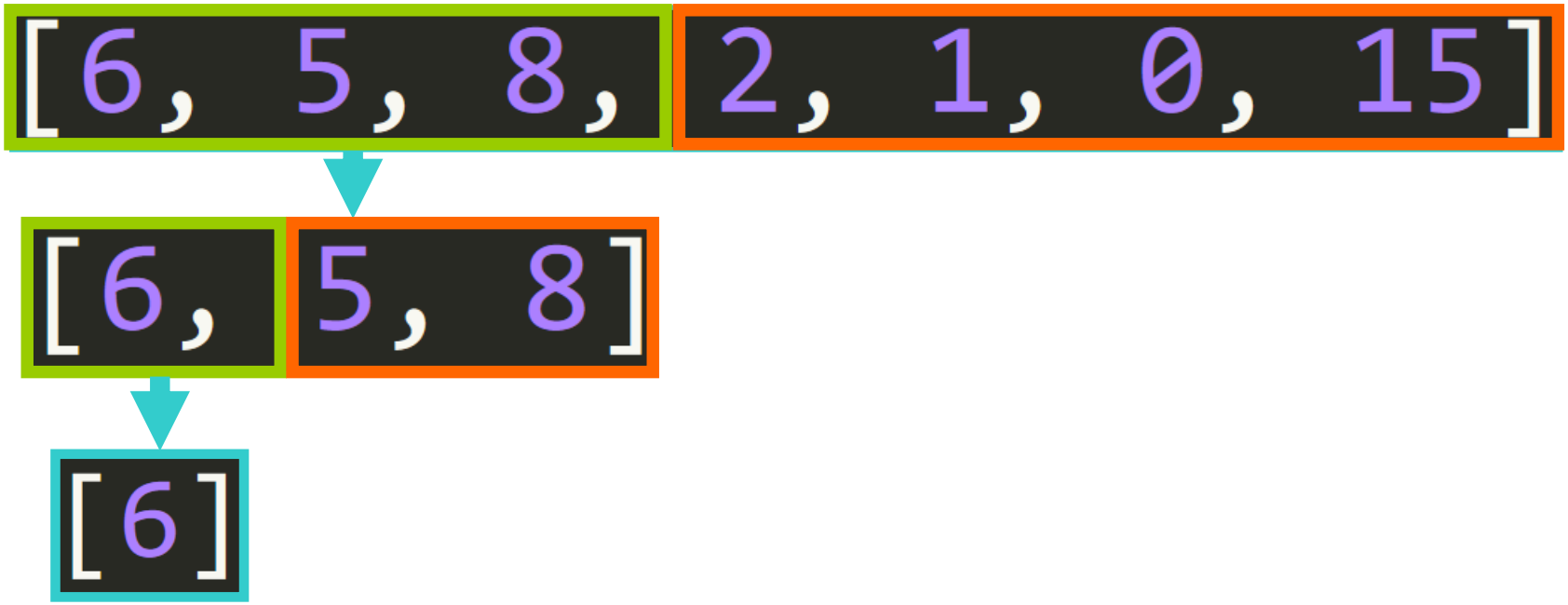
Merge Sort



Merge Sort

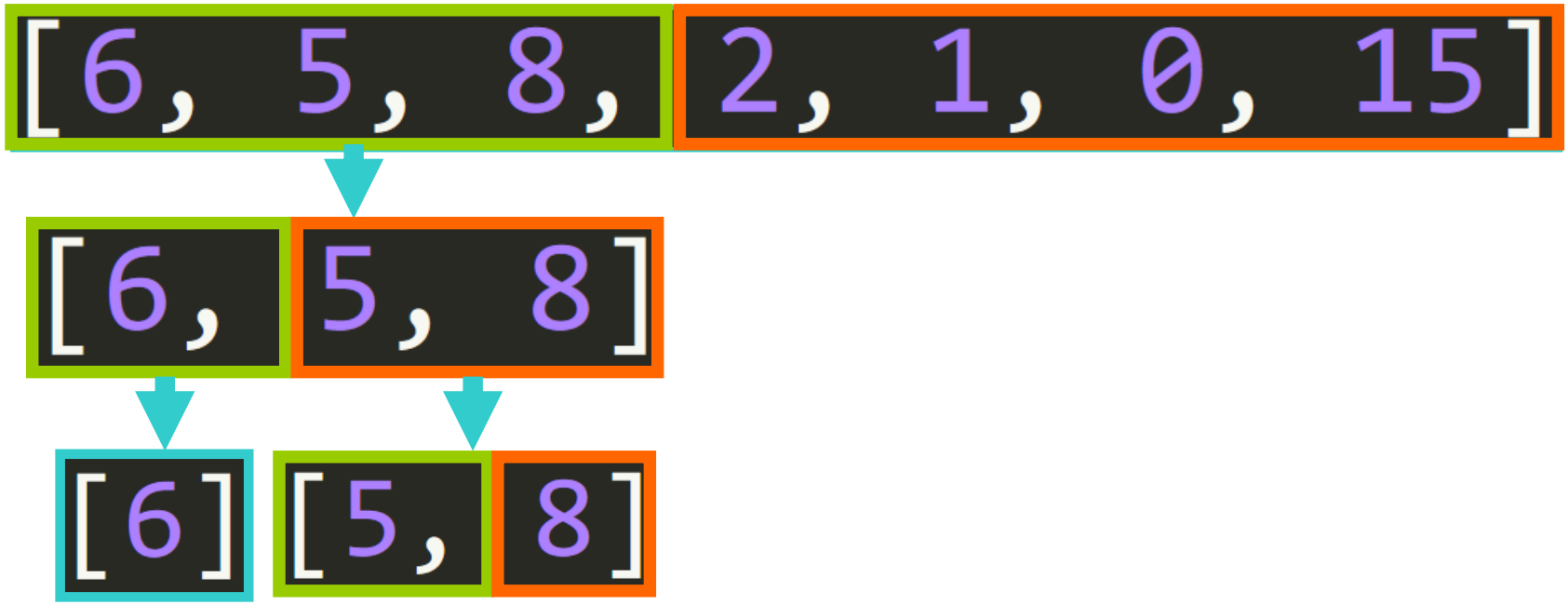


Merge Sort

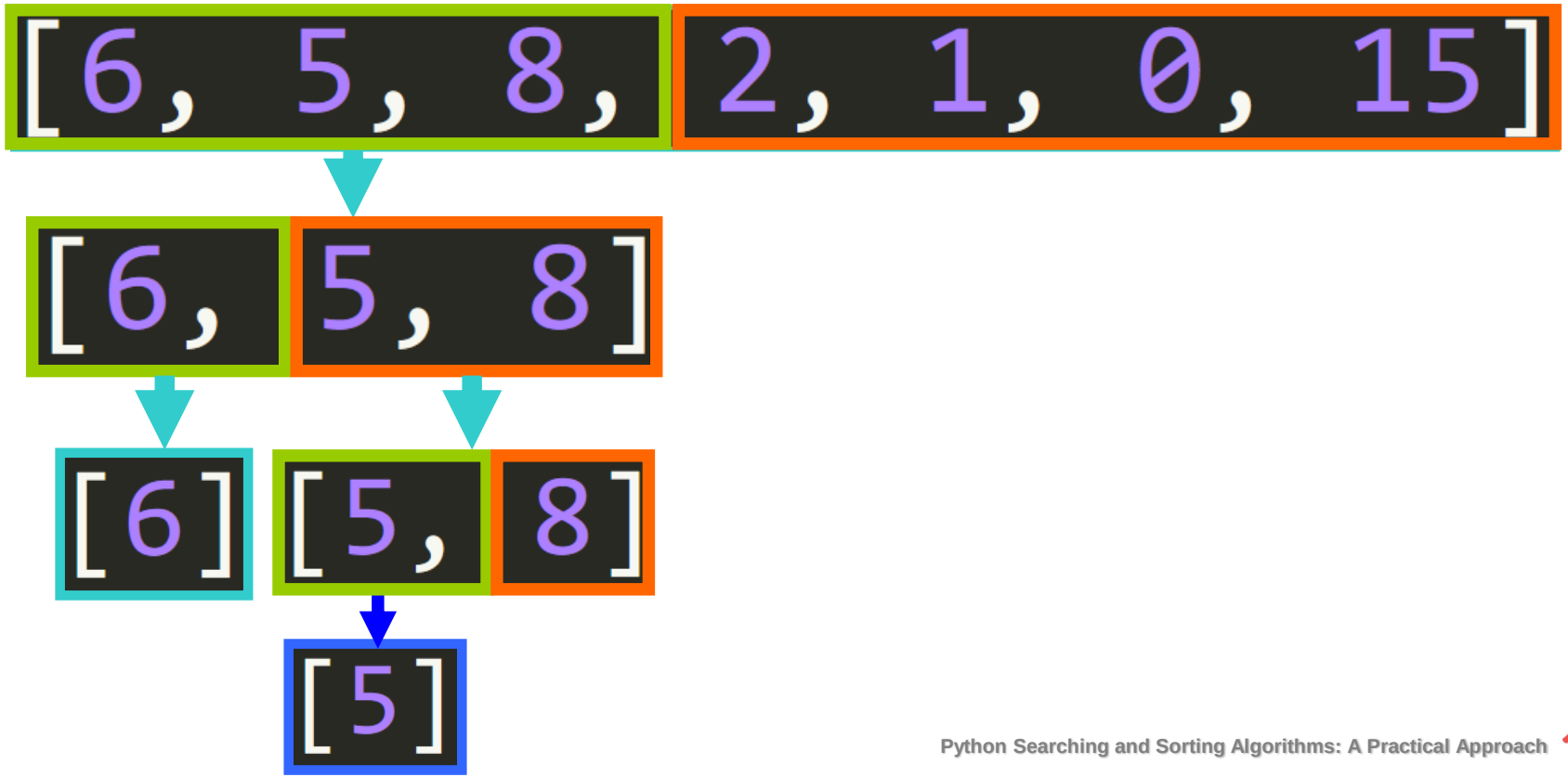


Merge Sort

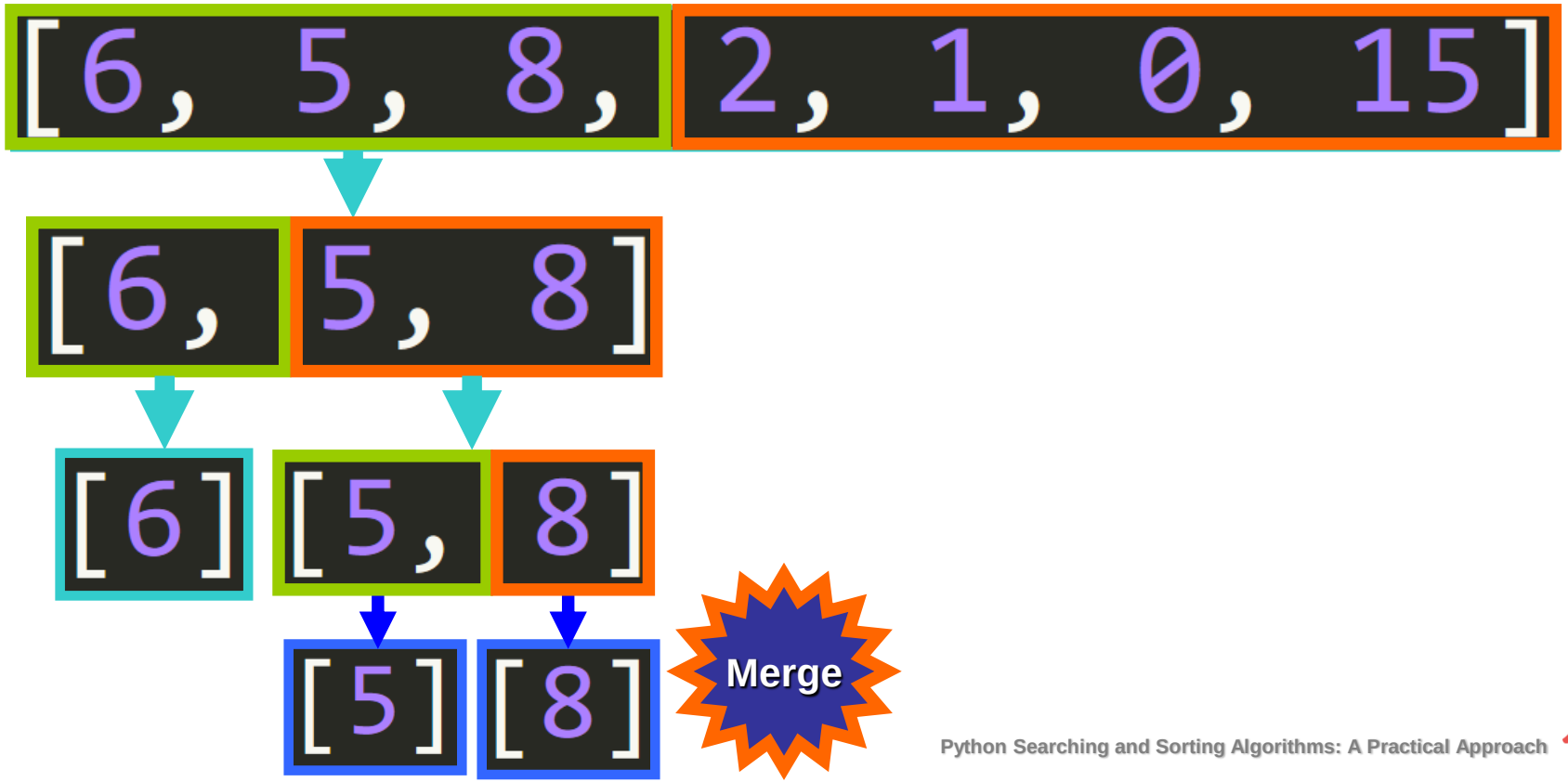
Merge Sort

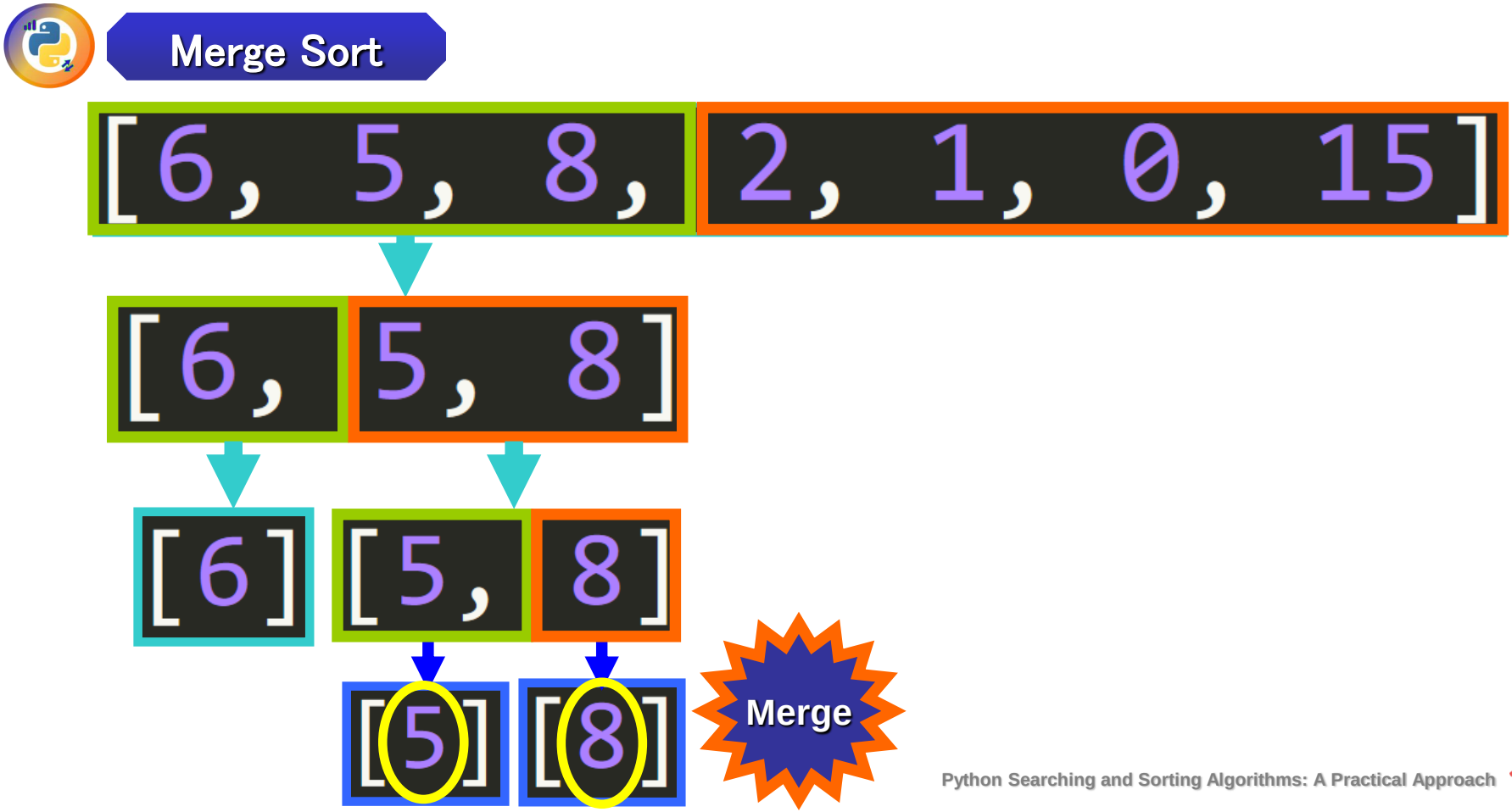


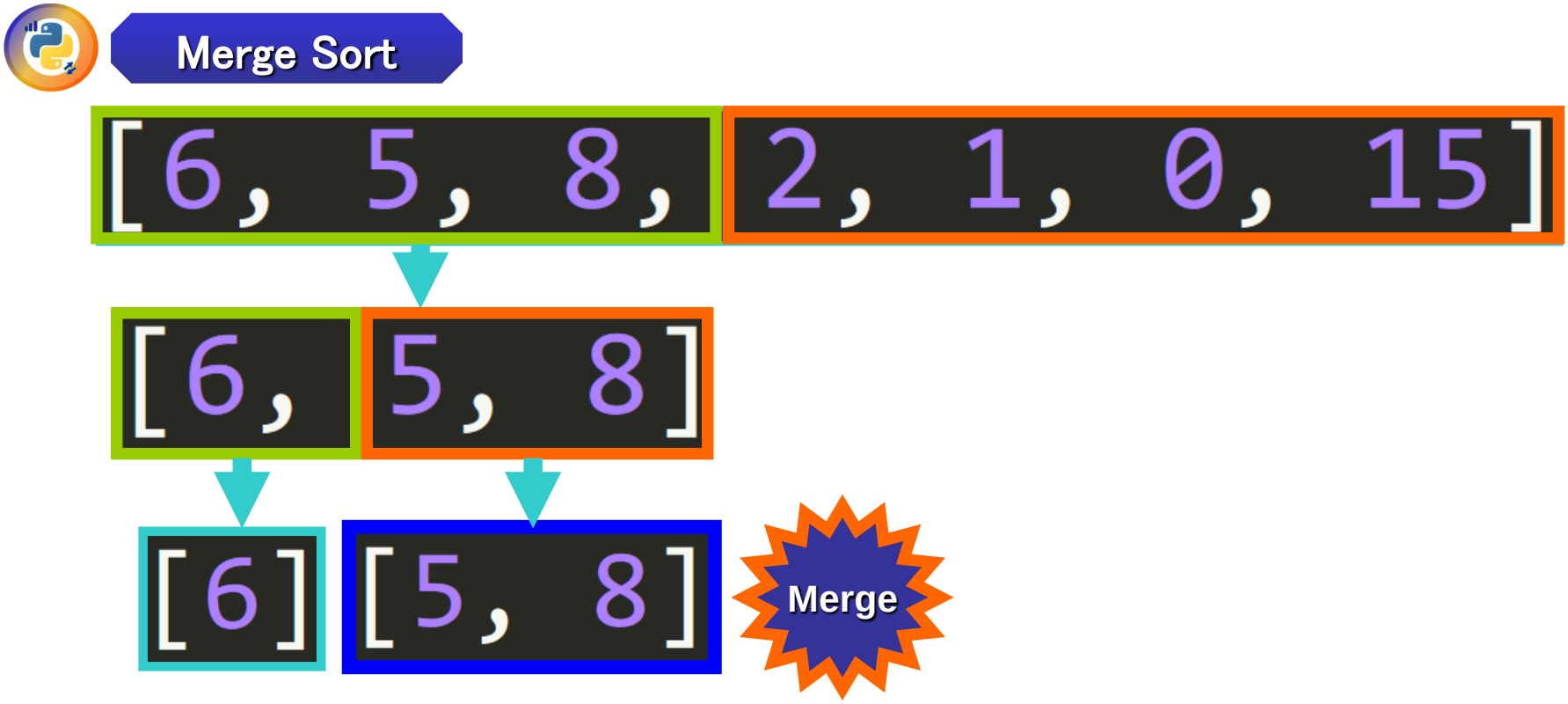
Merge Sort

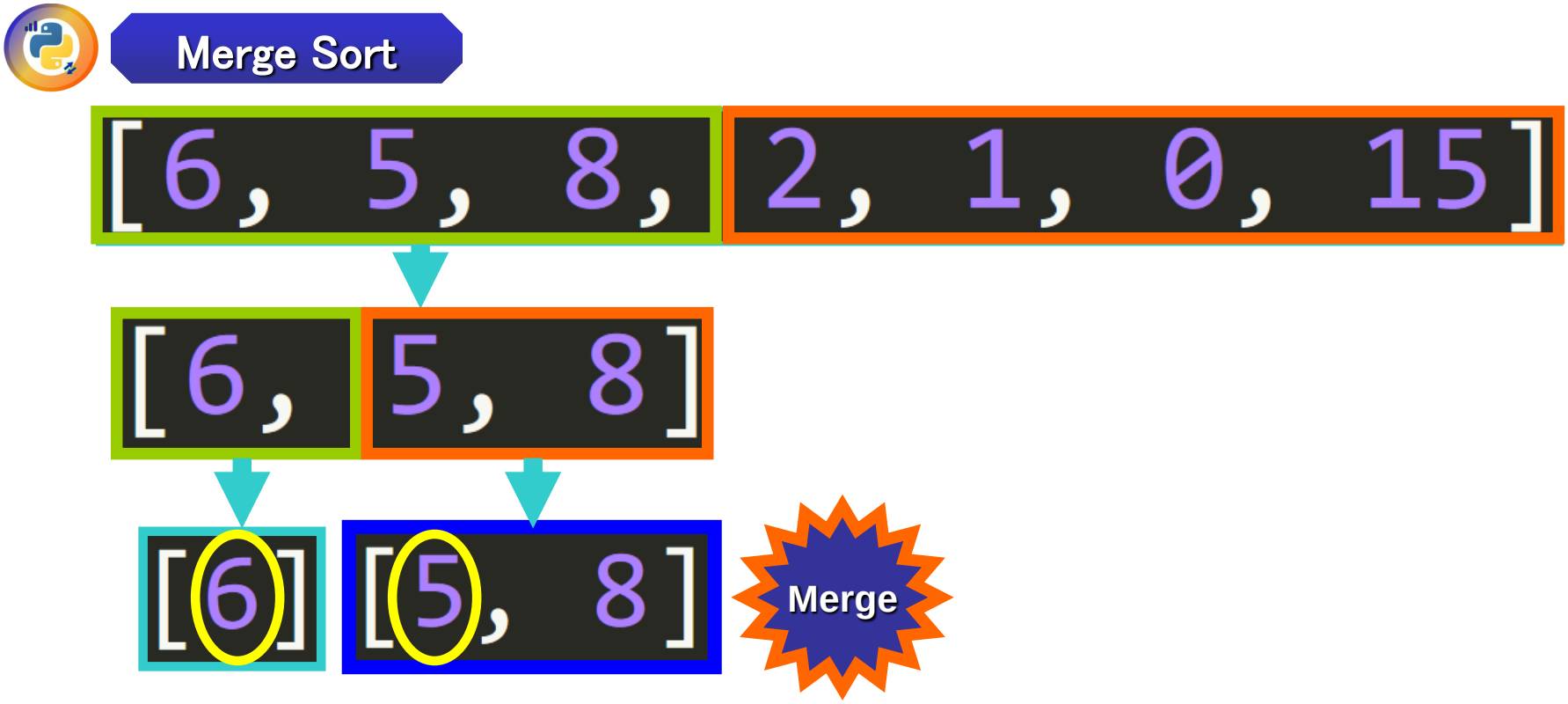


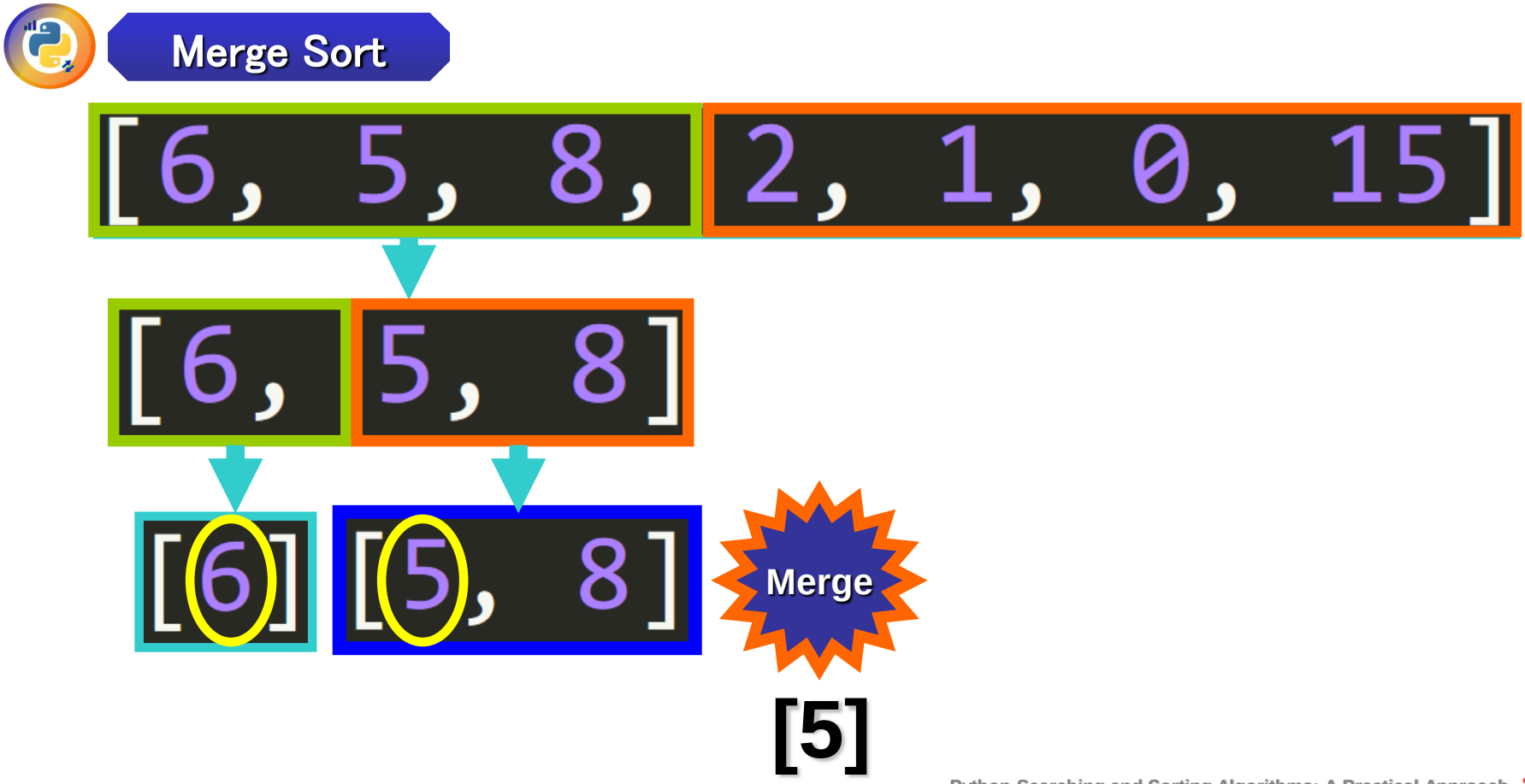
Merge Sort

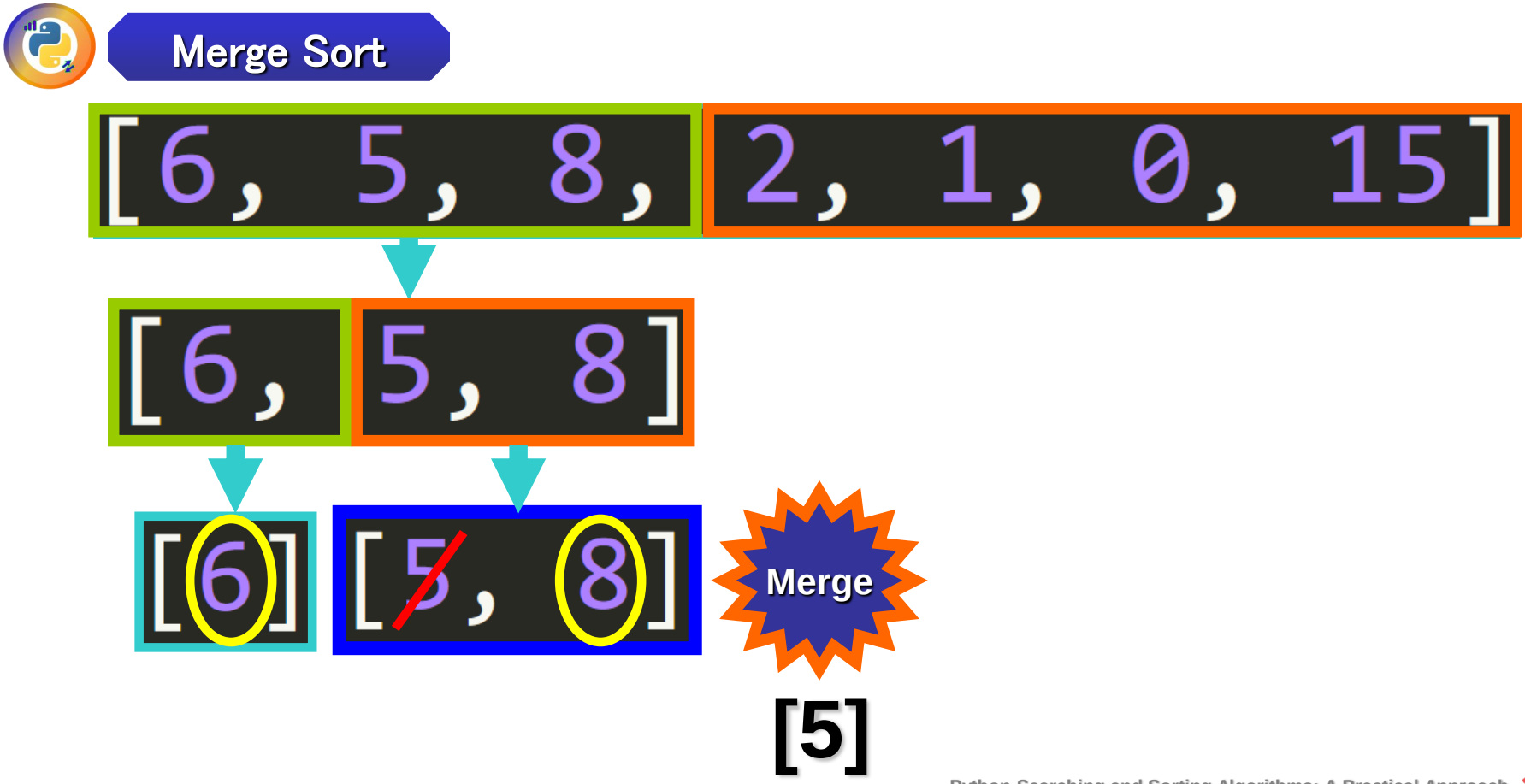


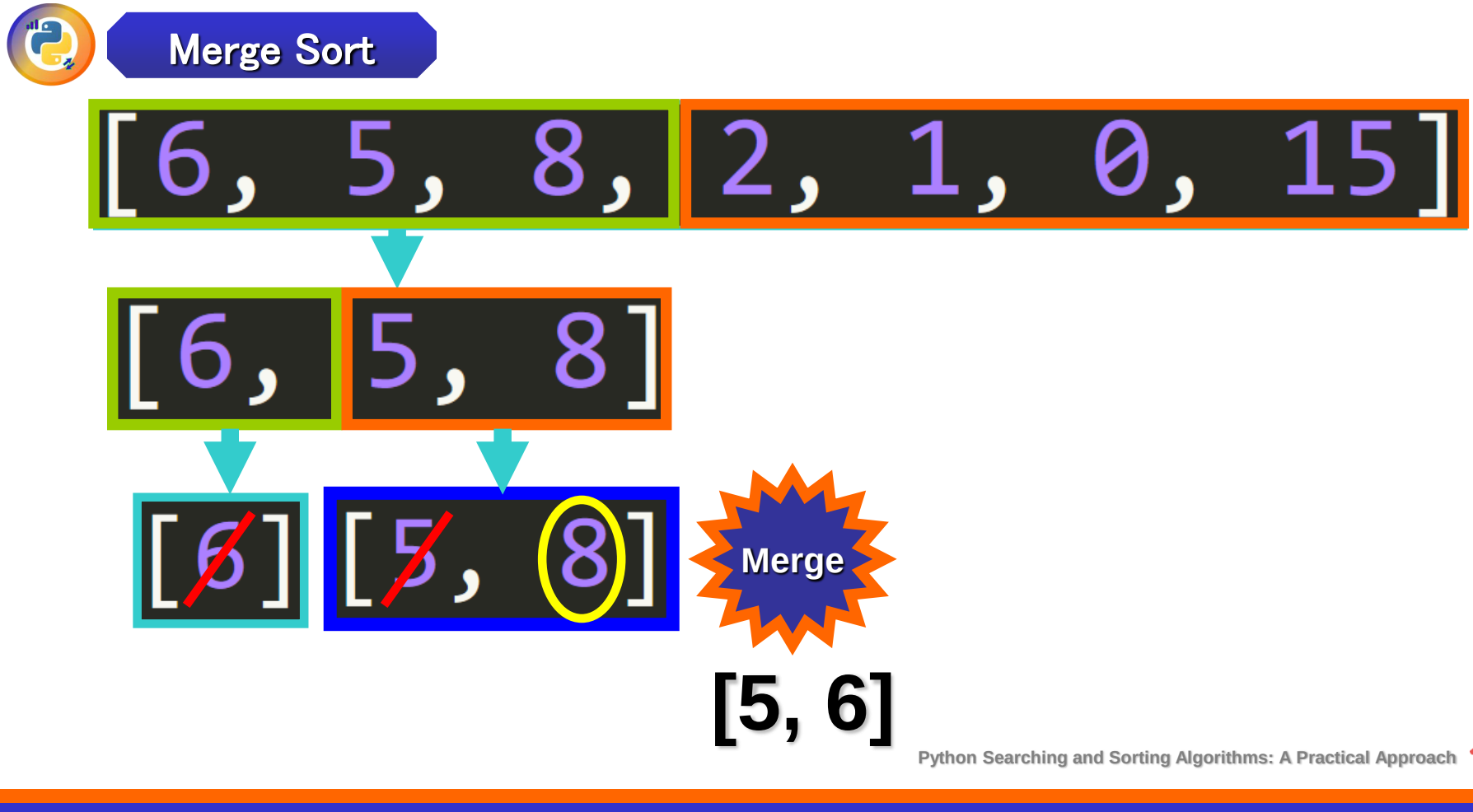

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[6, 5, 8]
[6]
[5, 8]
[5]
[8]
Merge


Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[6, 5, 8]
[6]
[5, 8]
Merge


Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[6, 5, 8]
[6]
[5, 8]
Merge

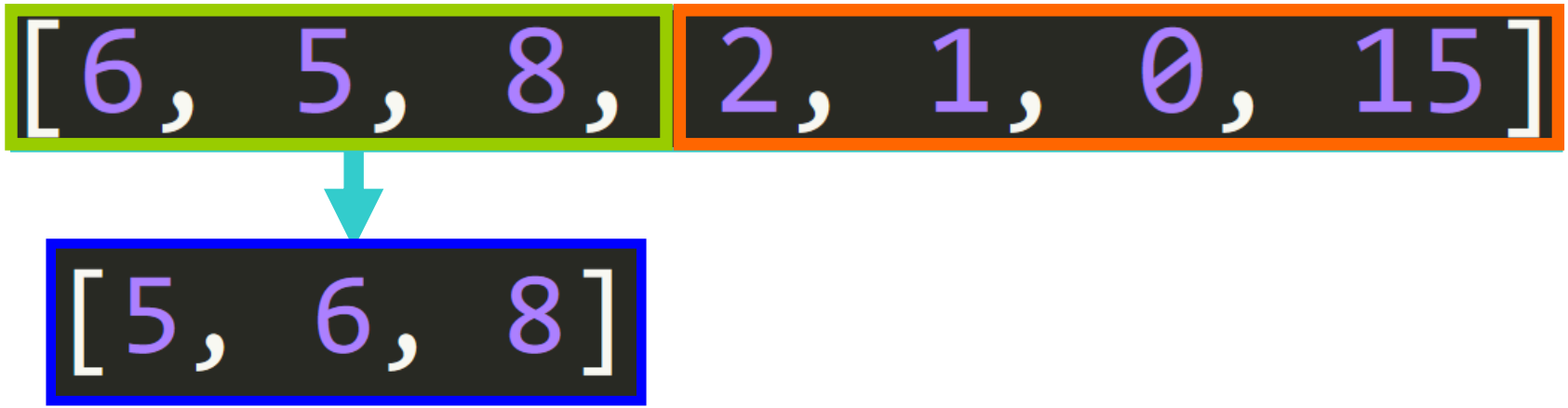

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[6, 5, 8]
[6]
[5, 8]
Merge
[5]


Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[6, 5, 8]
[6]
[5, 8]
Merge
[5]

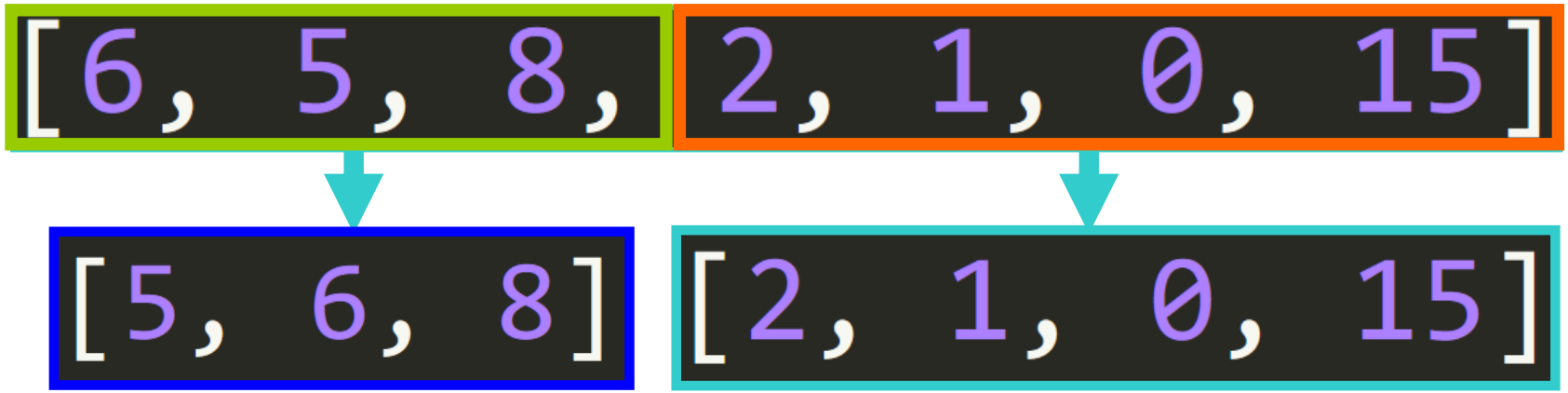

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[6, 5, 8]
[6]
[5, 8]
Merge
[5, 6]

Merge Sort

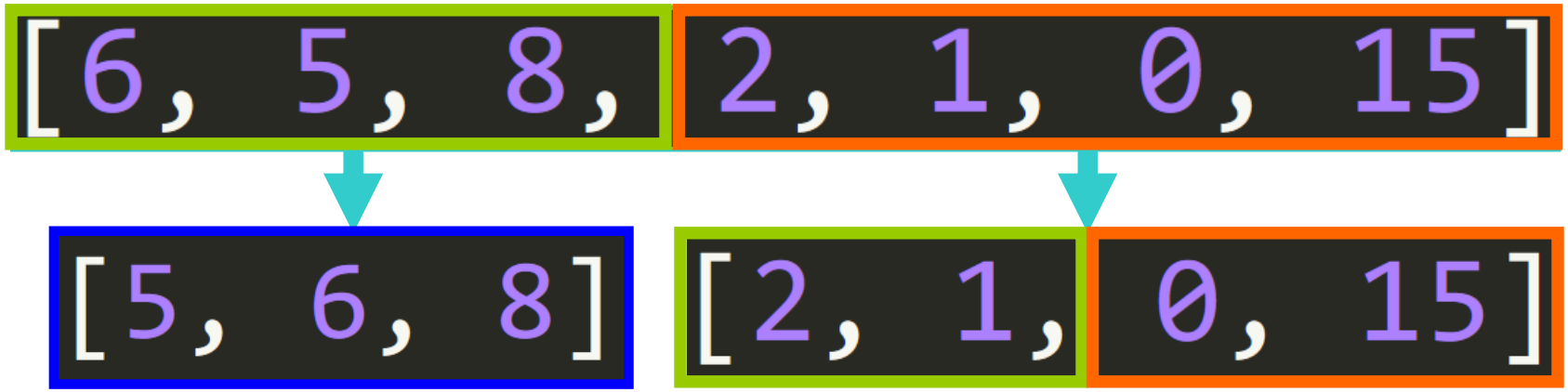
Merge Sort



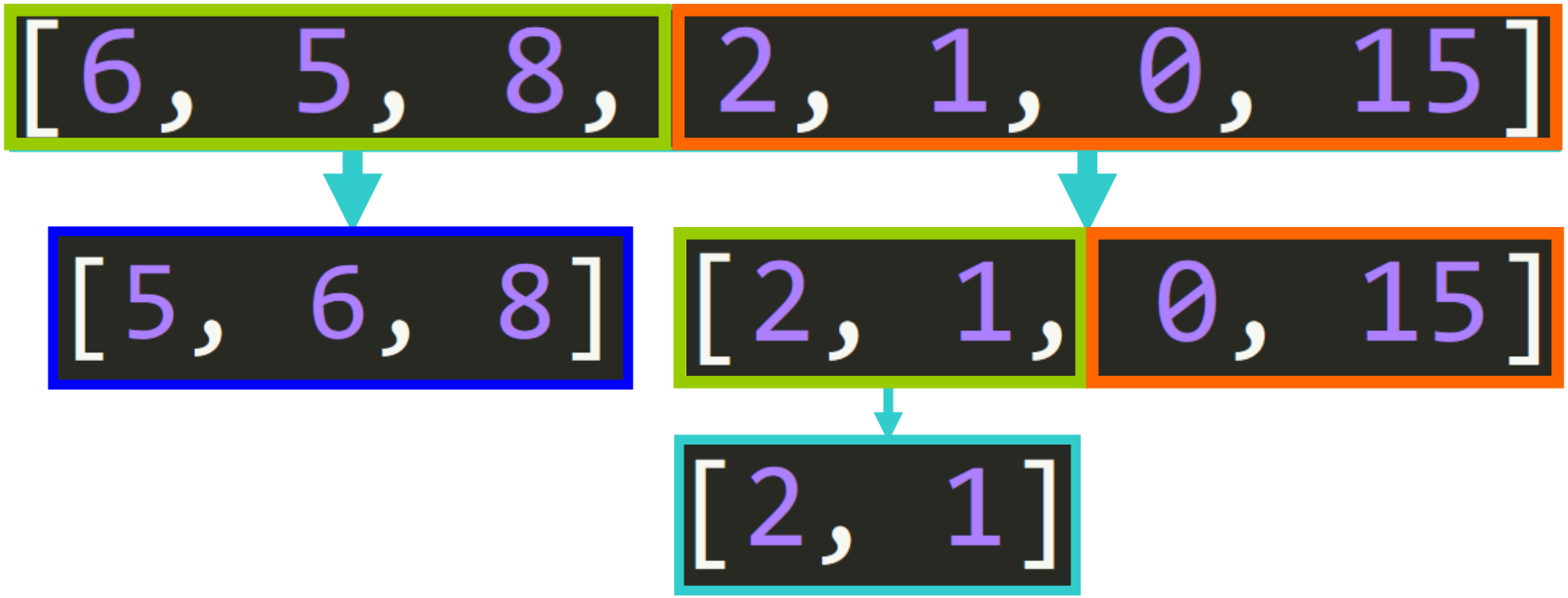
Merge Sort



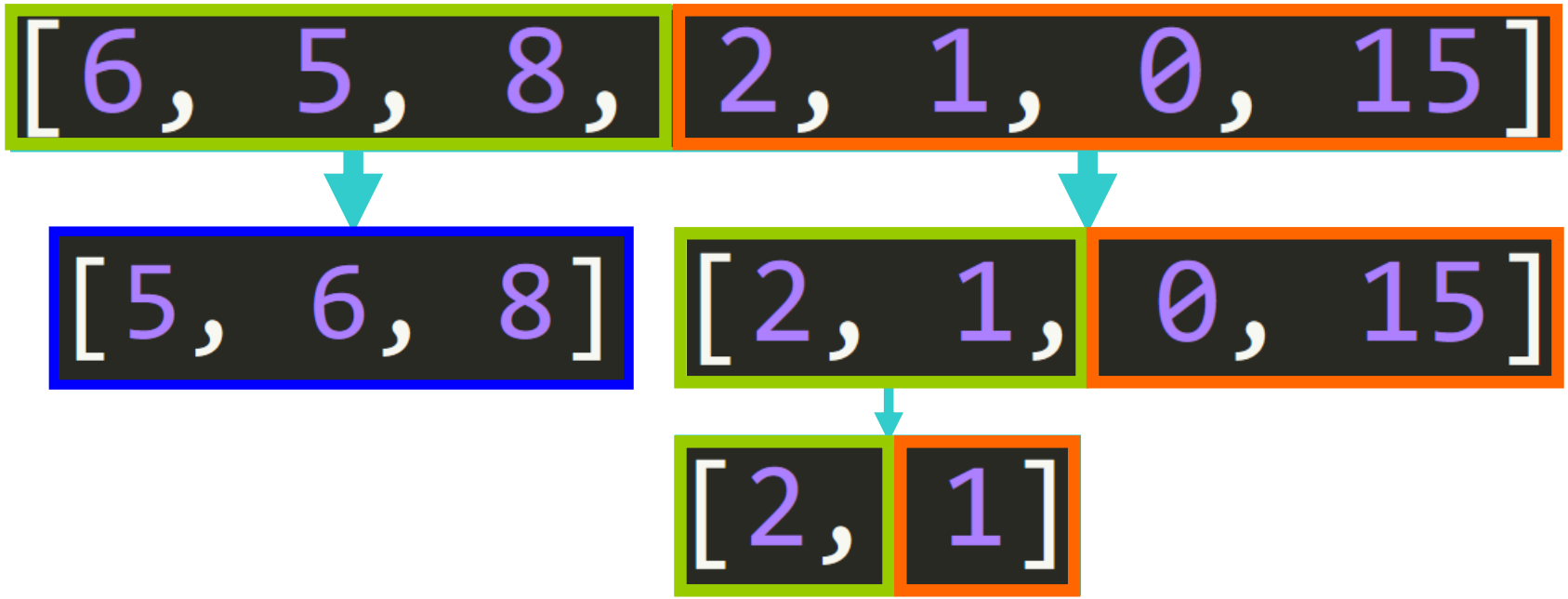
Merge Sort



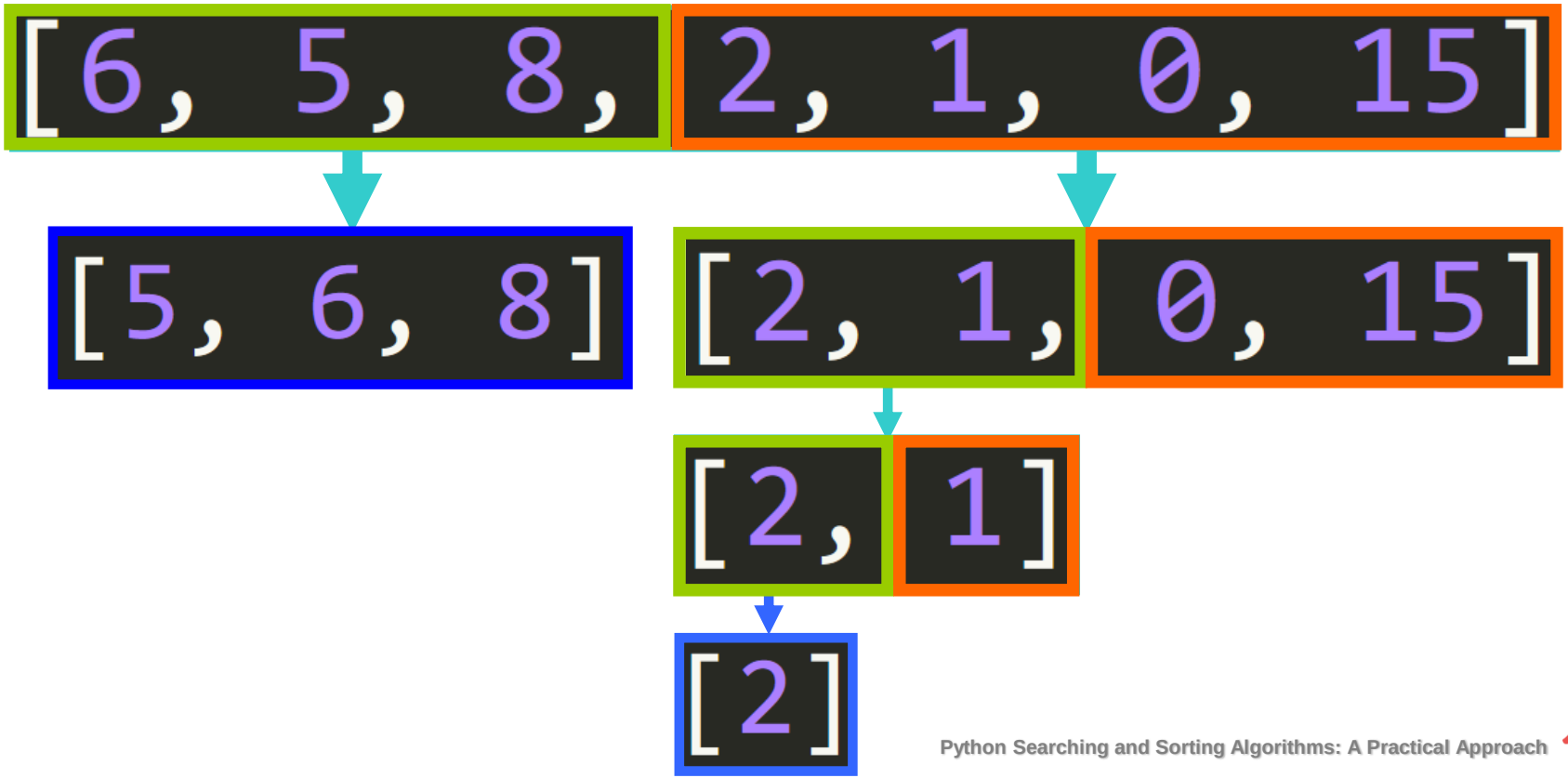
Merge Sort



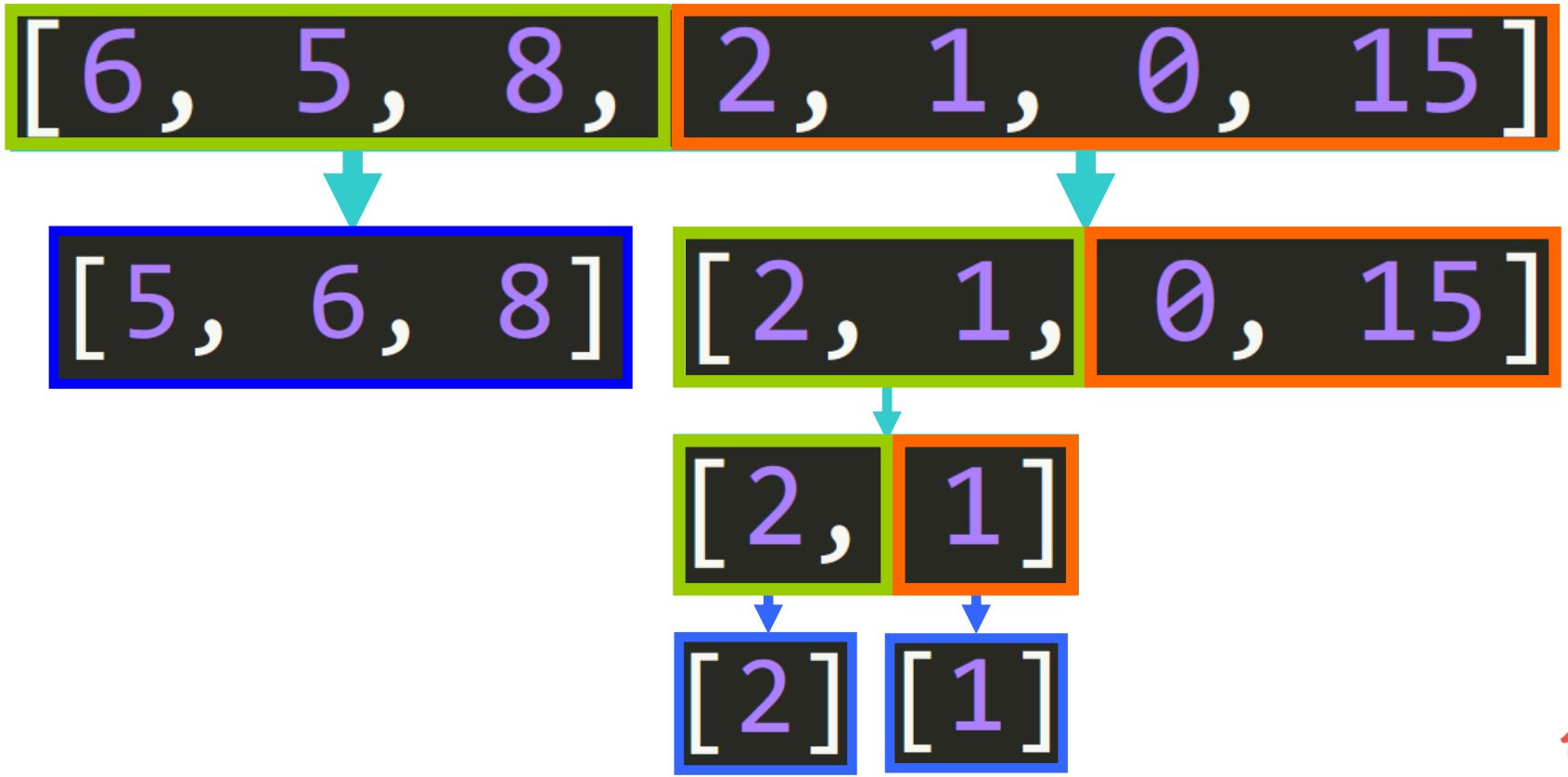
Merge Sort



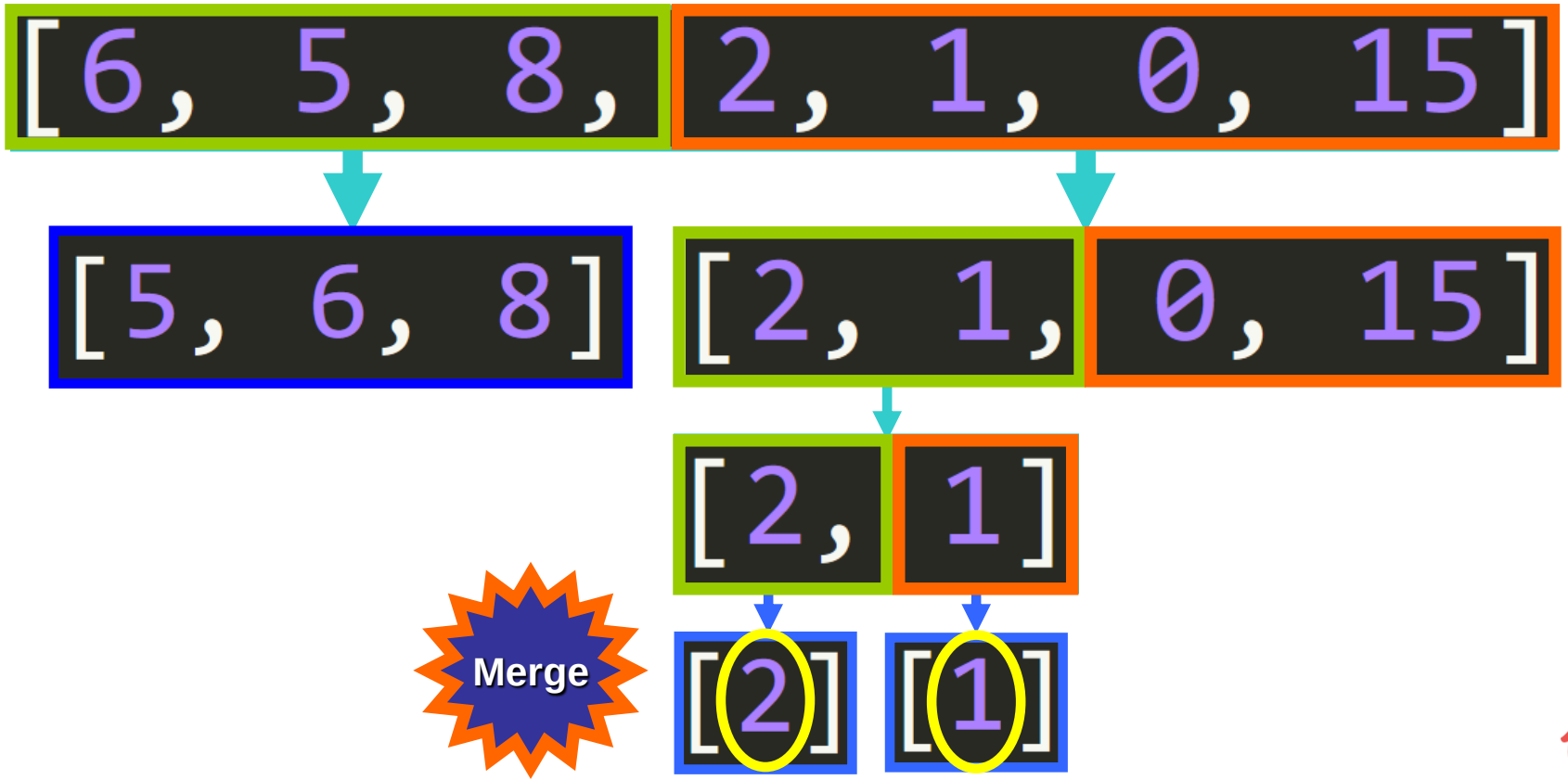
Merge Sort



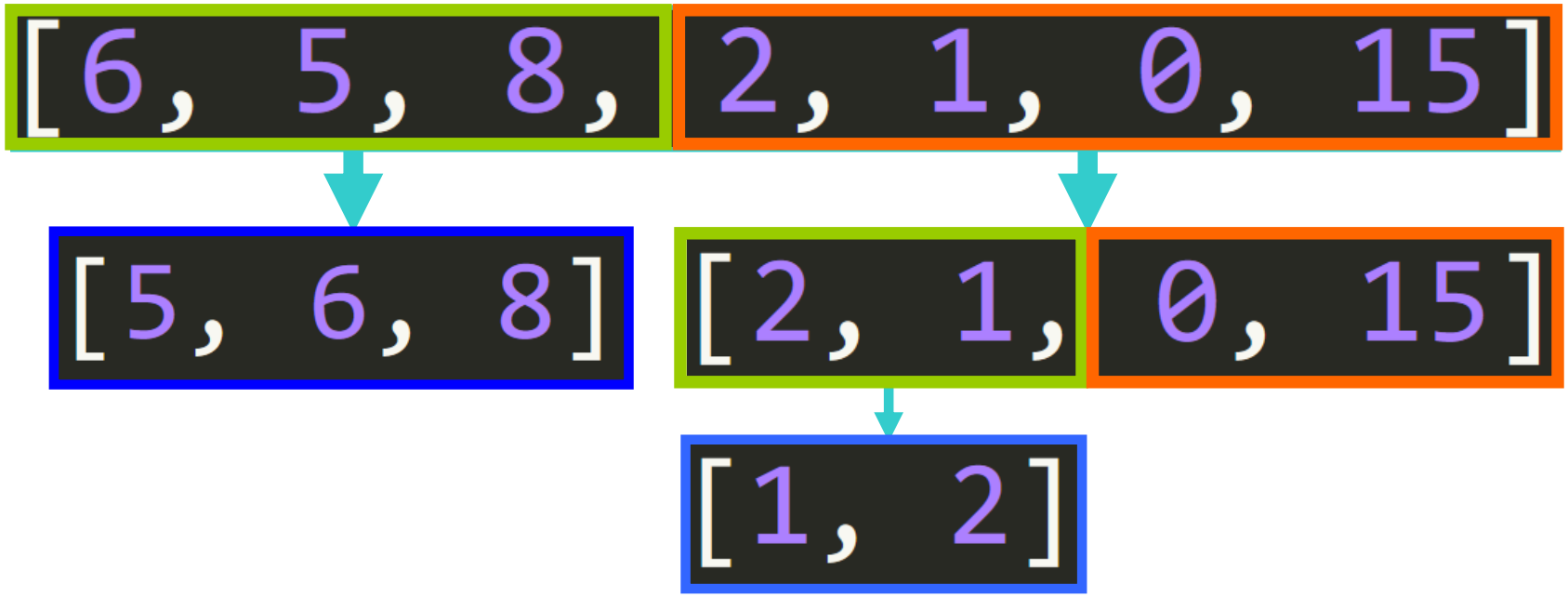
Merge Sort

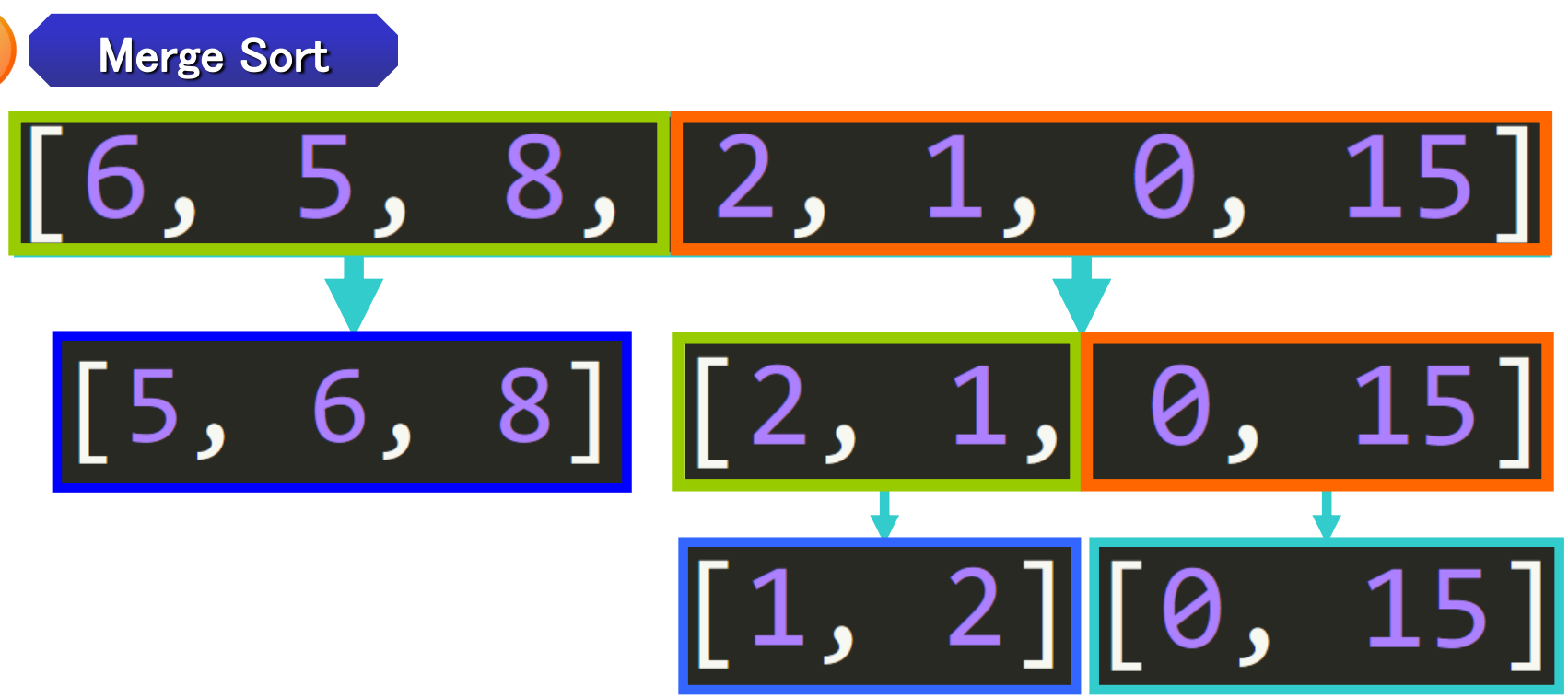


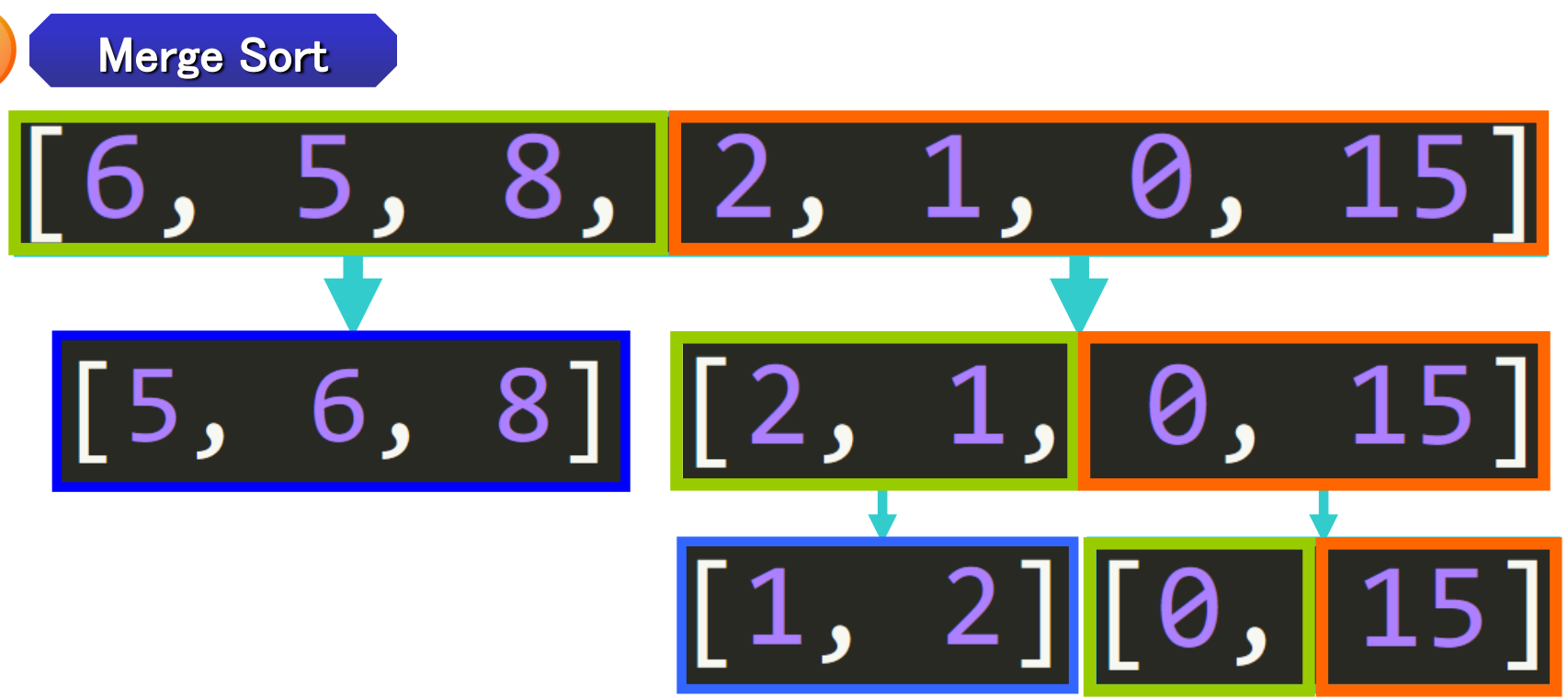
Merge Sort



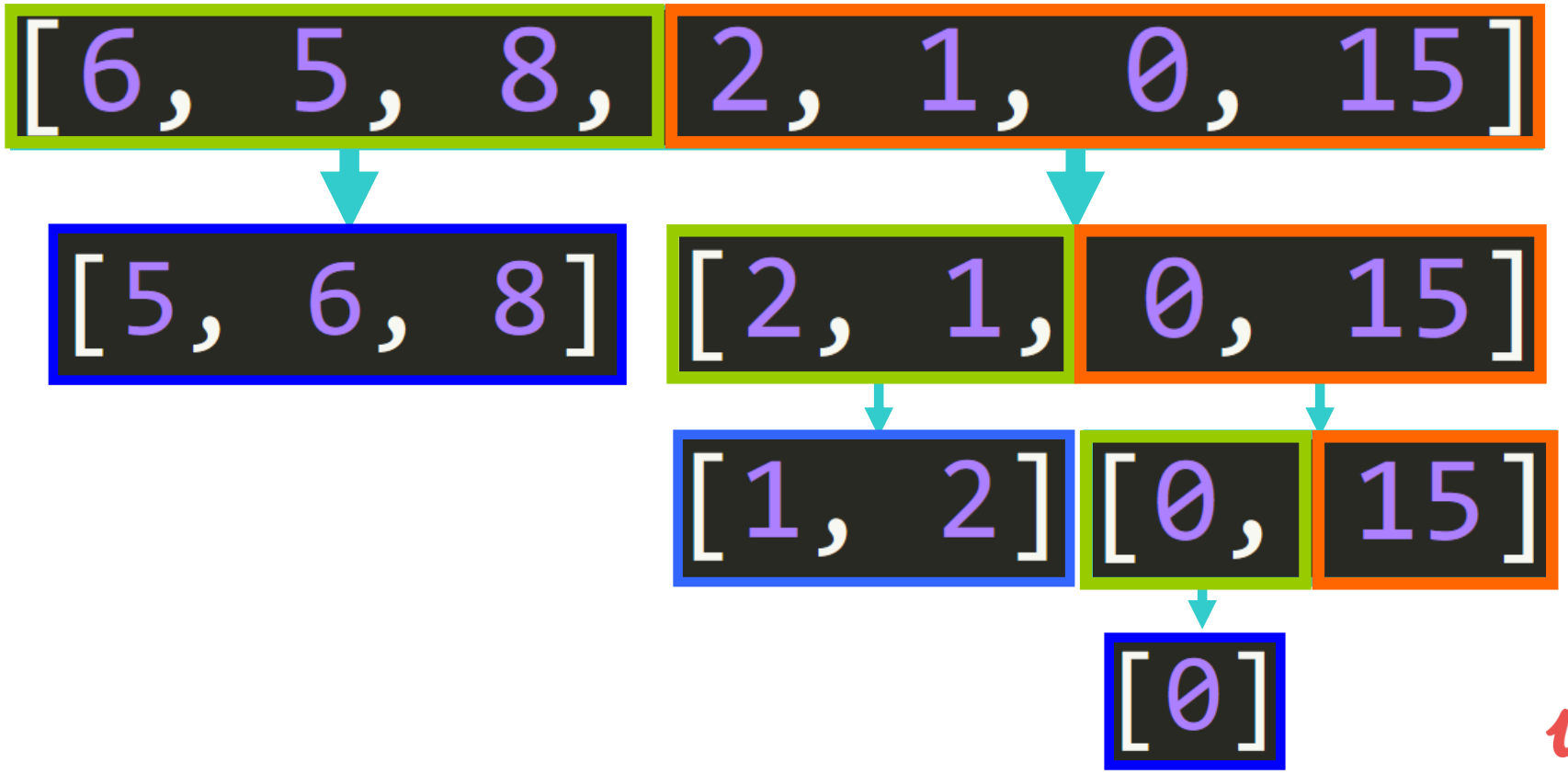
Merge Sort

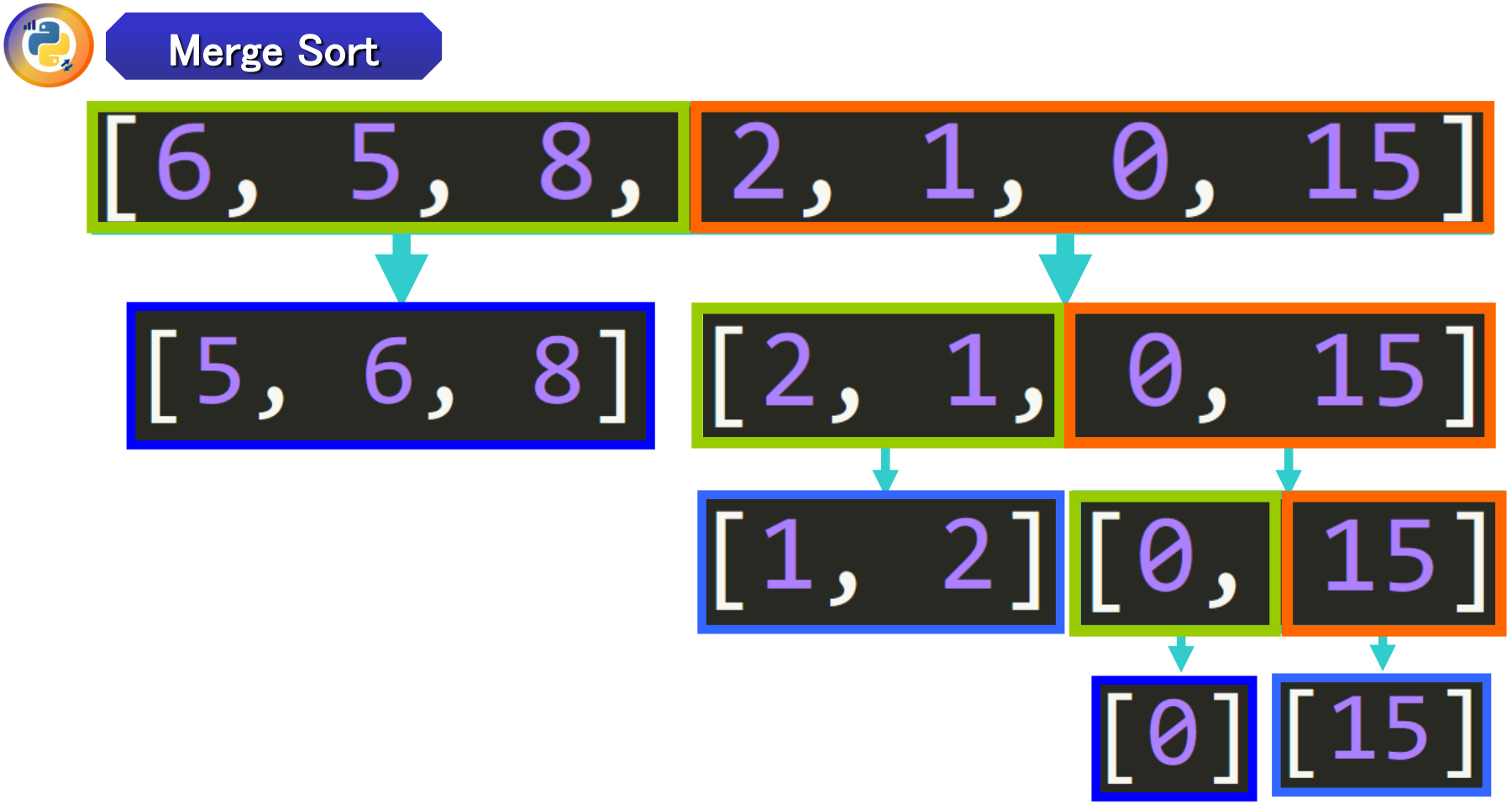



Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[2, 1,
0, 15]
[1, 2]
[0, 15]

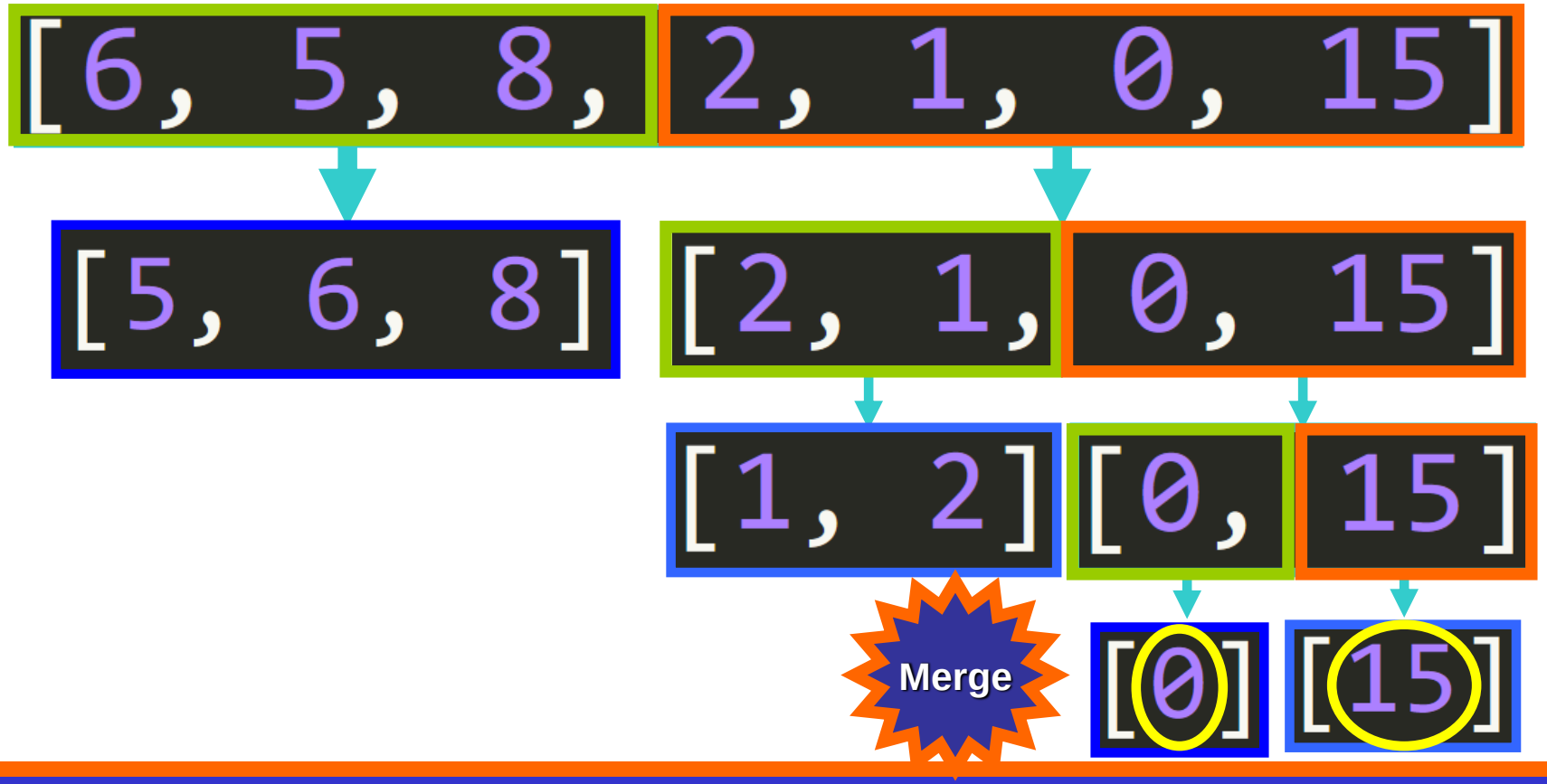

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
[1, 2]
[0, 15]

Merge Sort

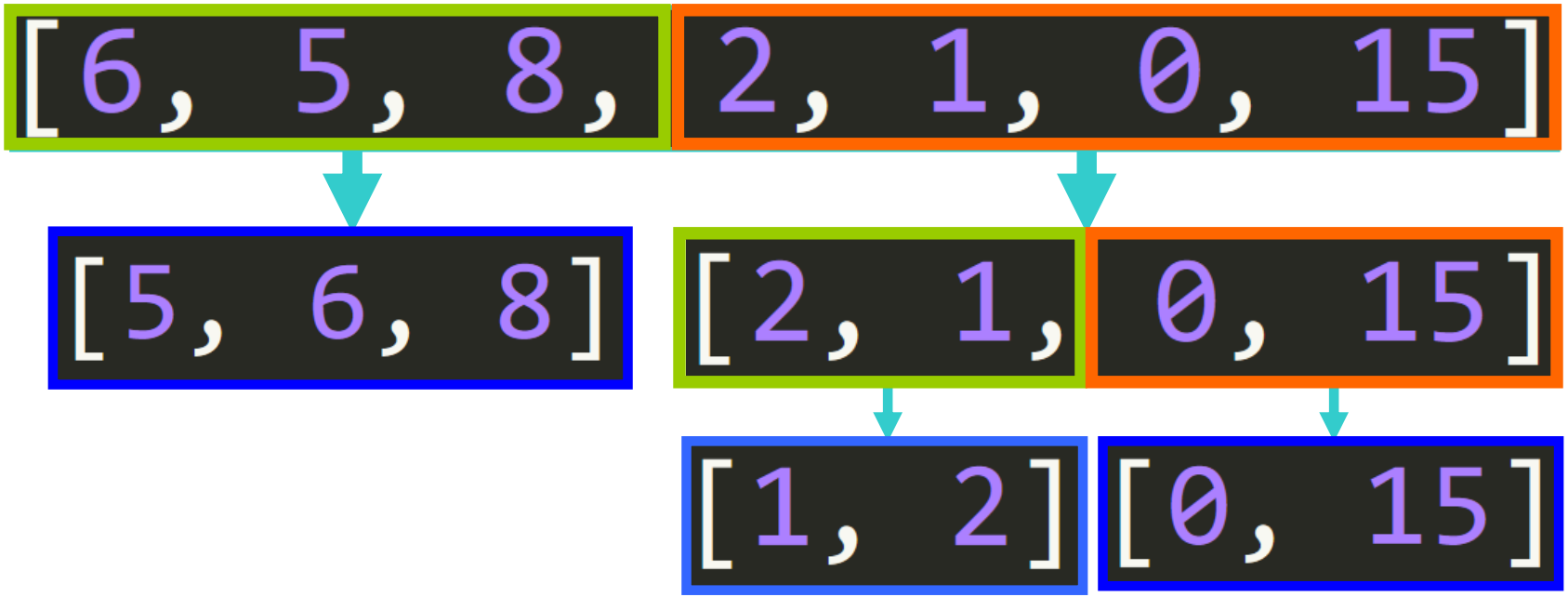



Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
[1, 2]
[0, 15]
[0]
[15]

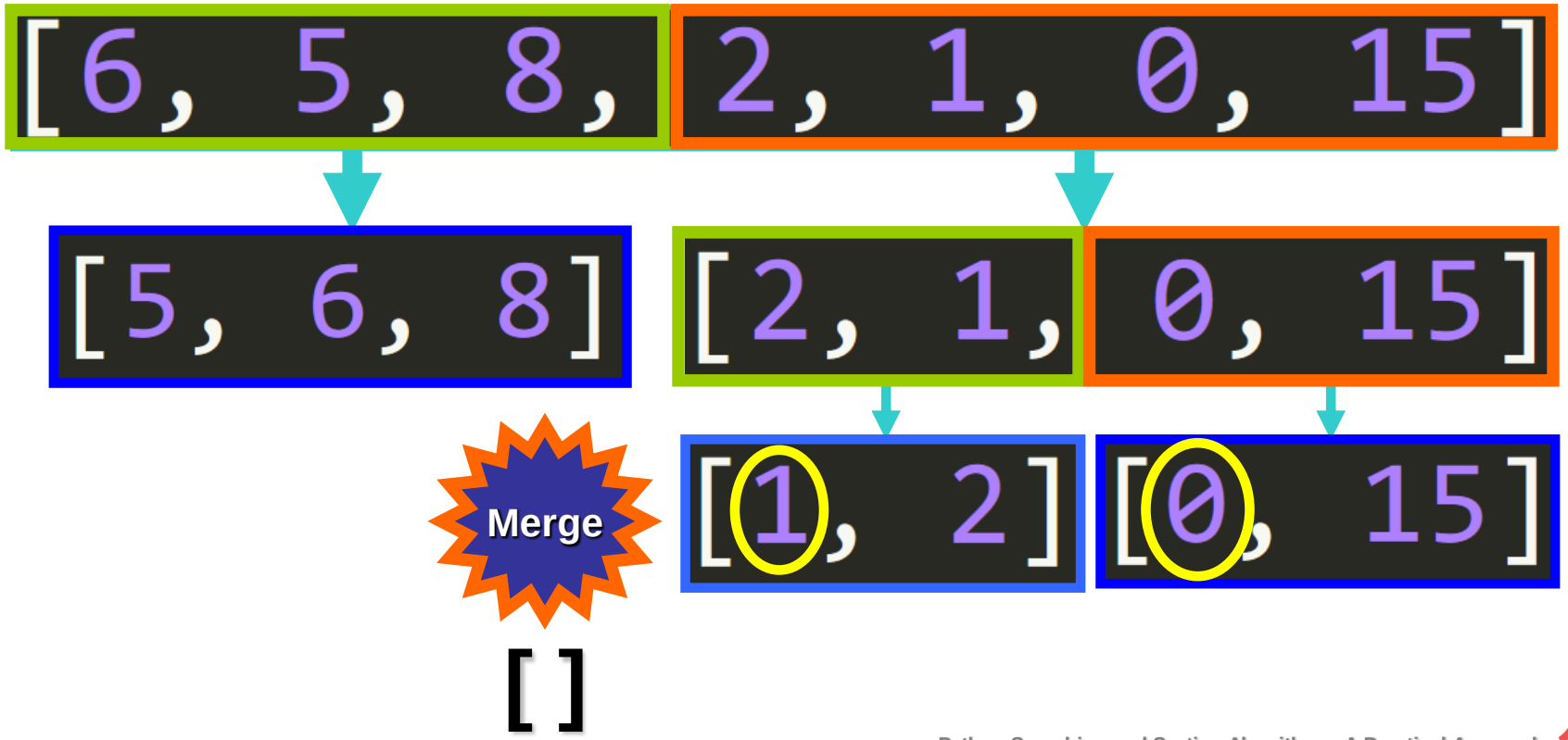
Merge Sort

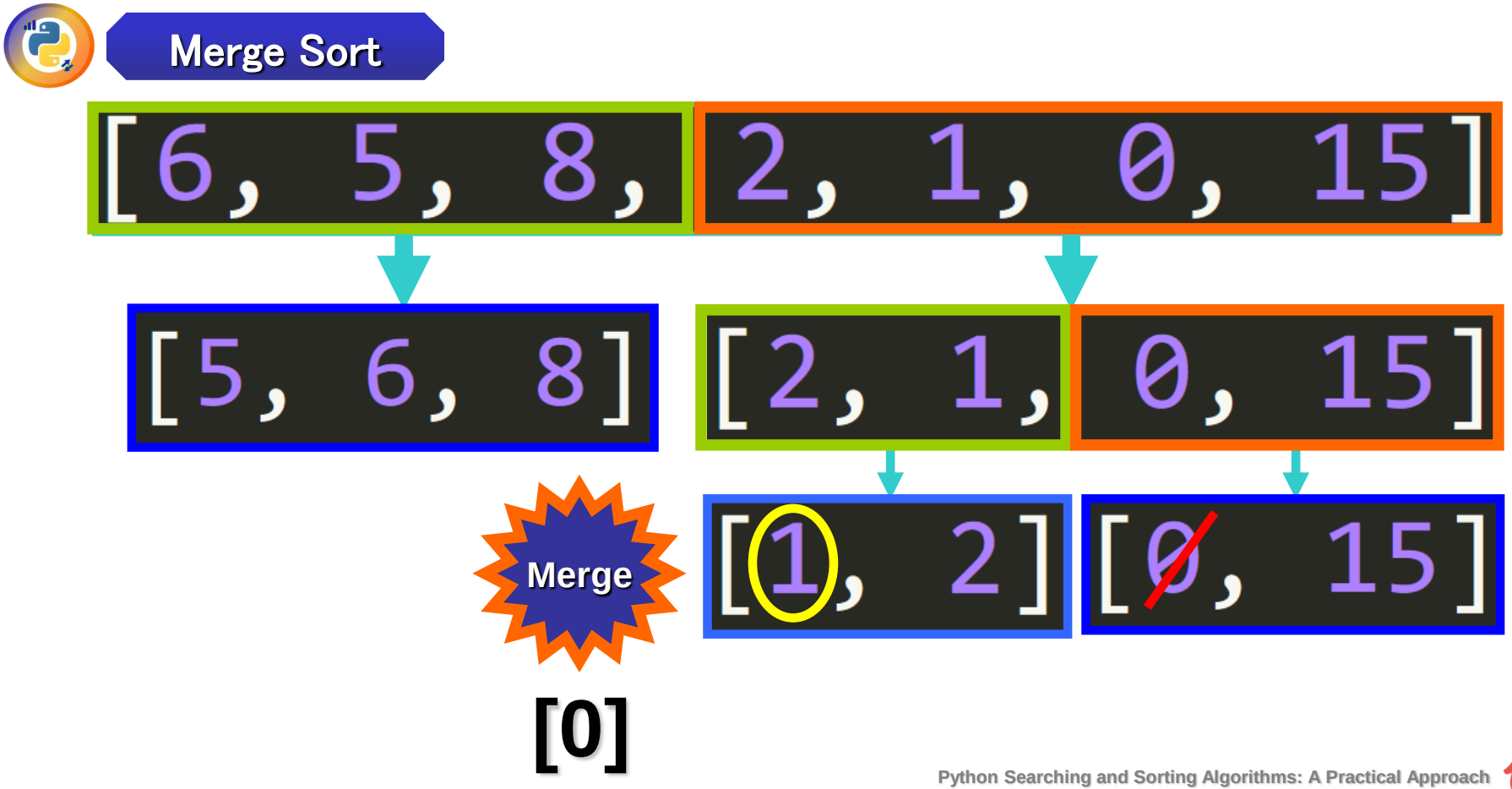


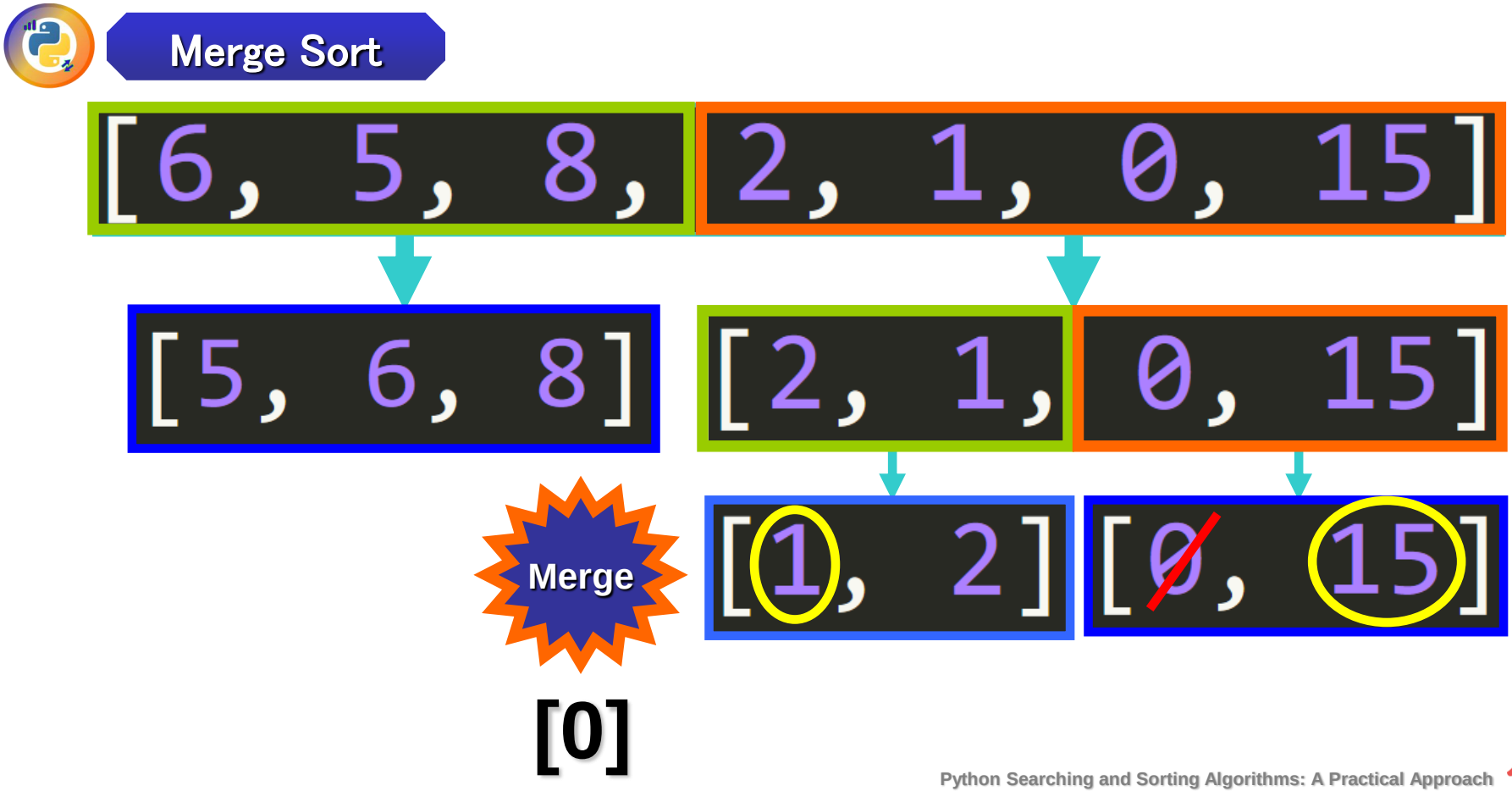
Merge Sort

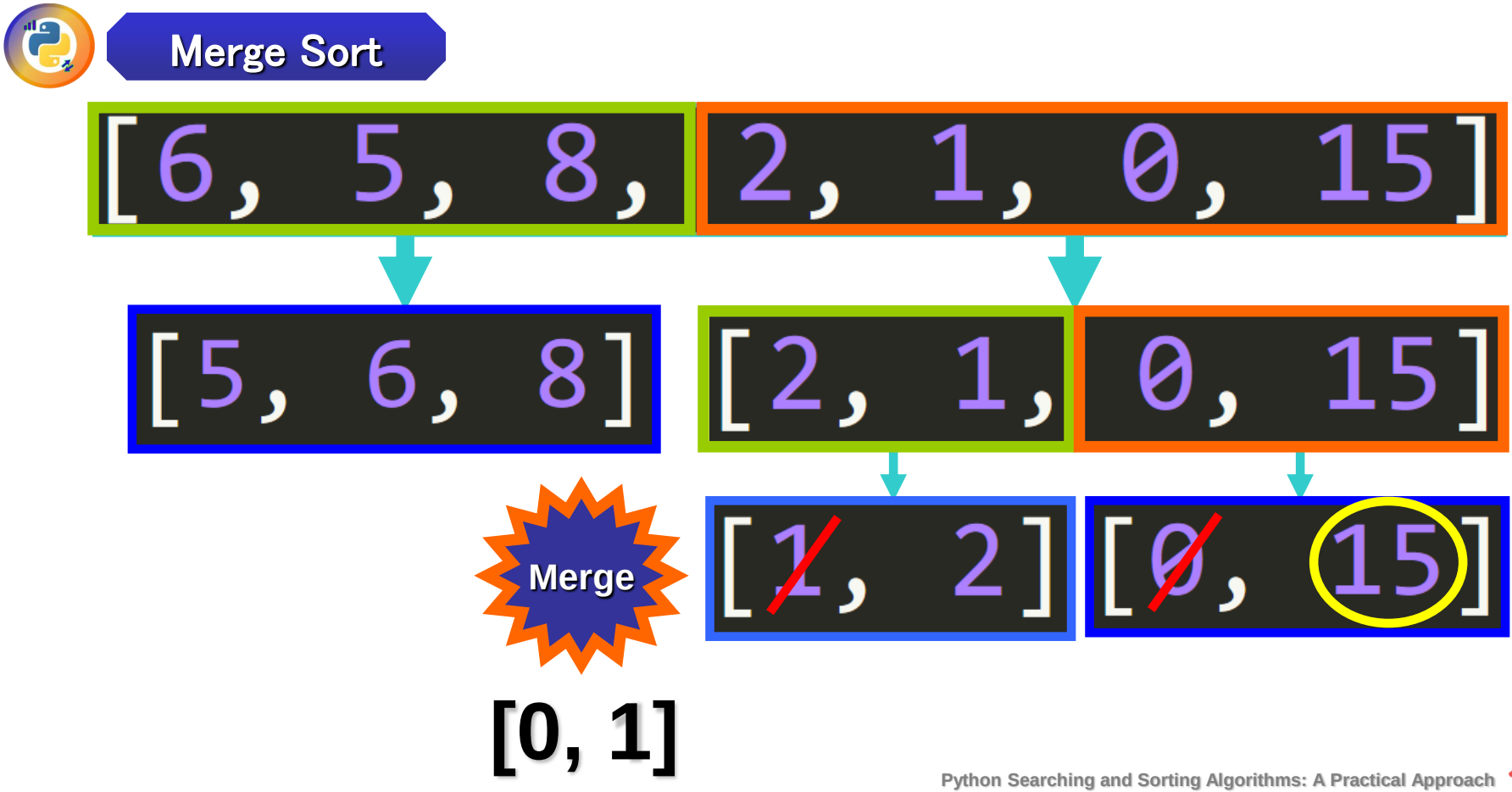


Merge Sort

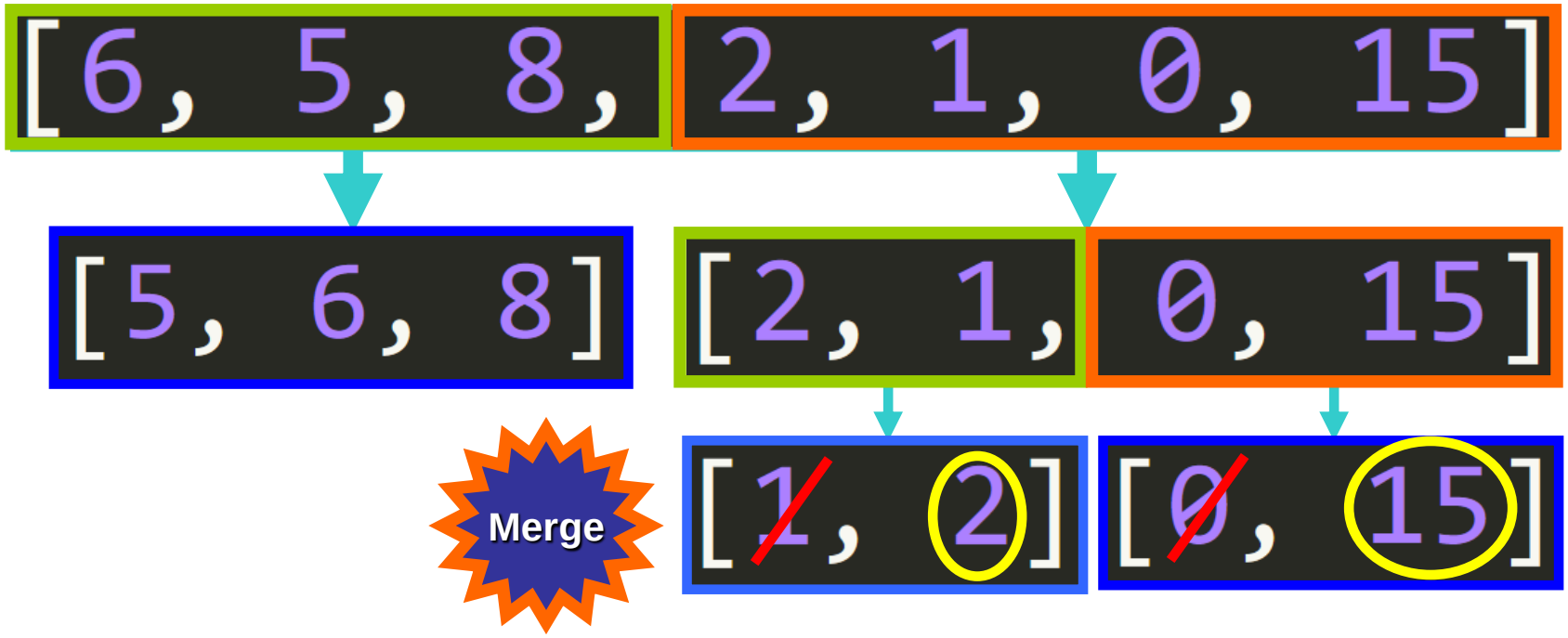



Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
Merge
[1, 2]
[0, 15]
[0]

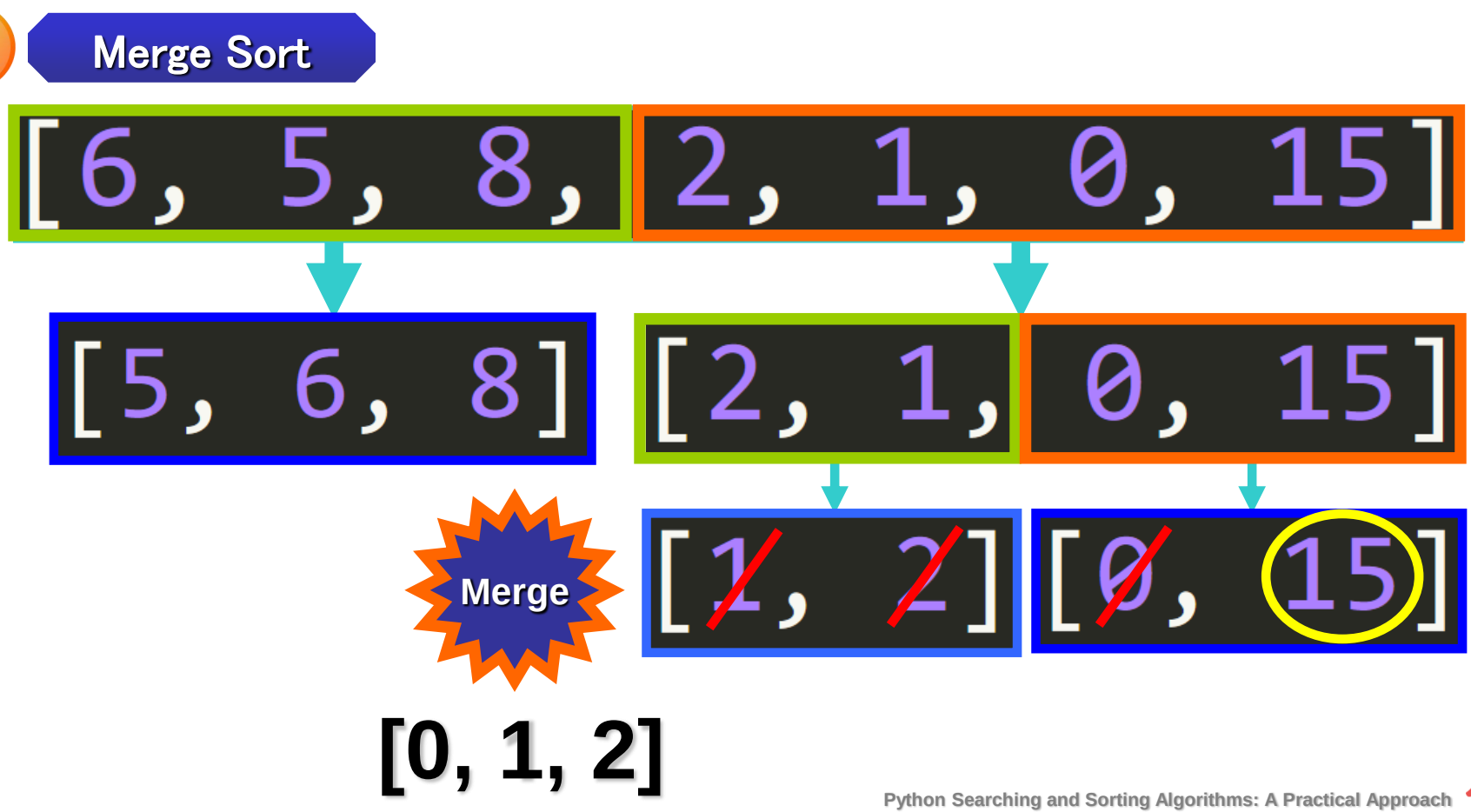

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
Merge
[1, 2]
[0, 15]
[0]

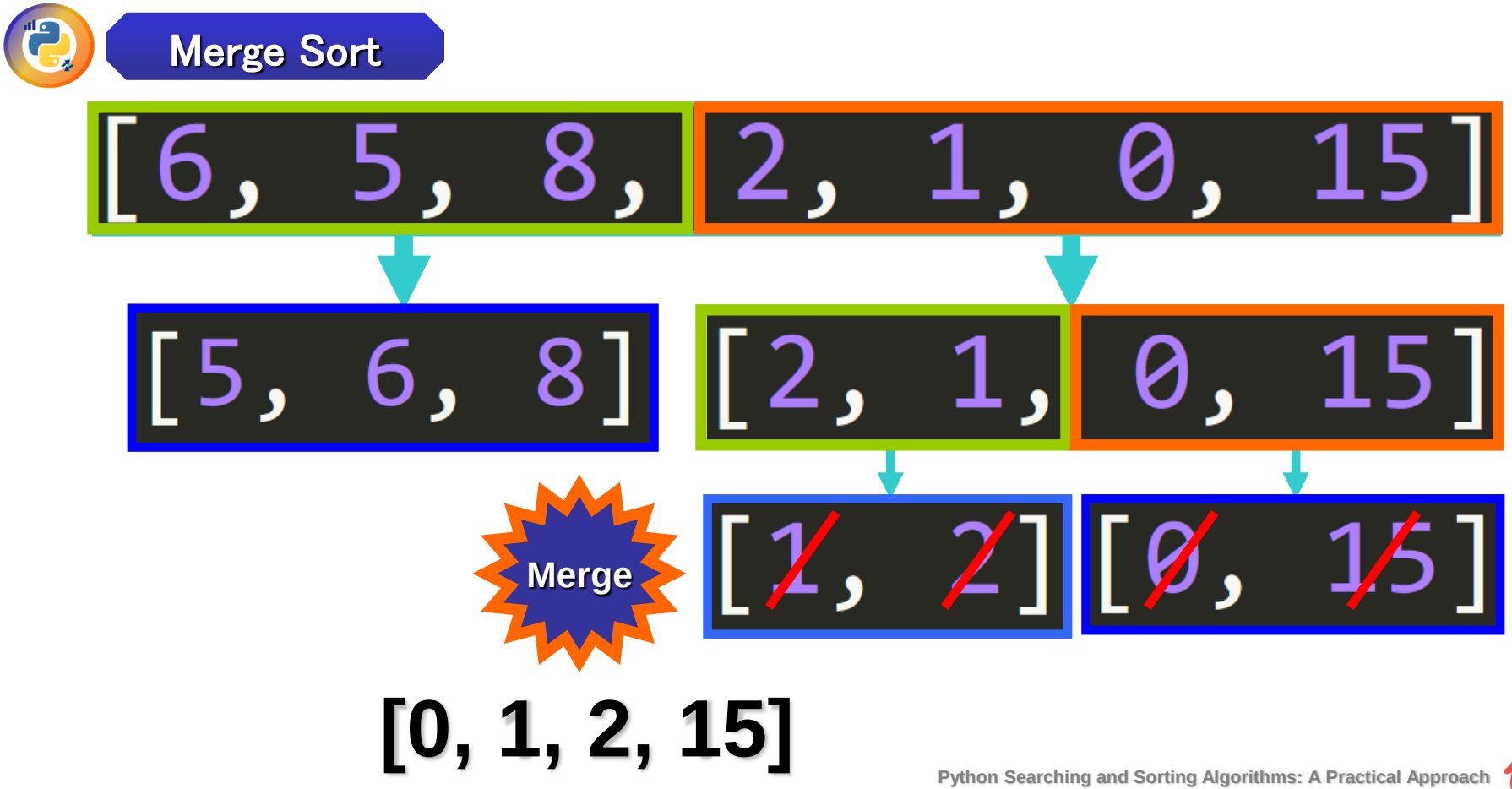

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
[1, 2]
[0, 15]
Merge
[0, 1]

Merge Sort

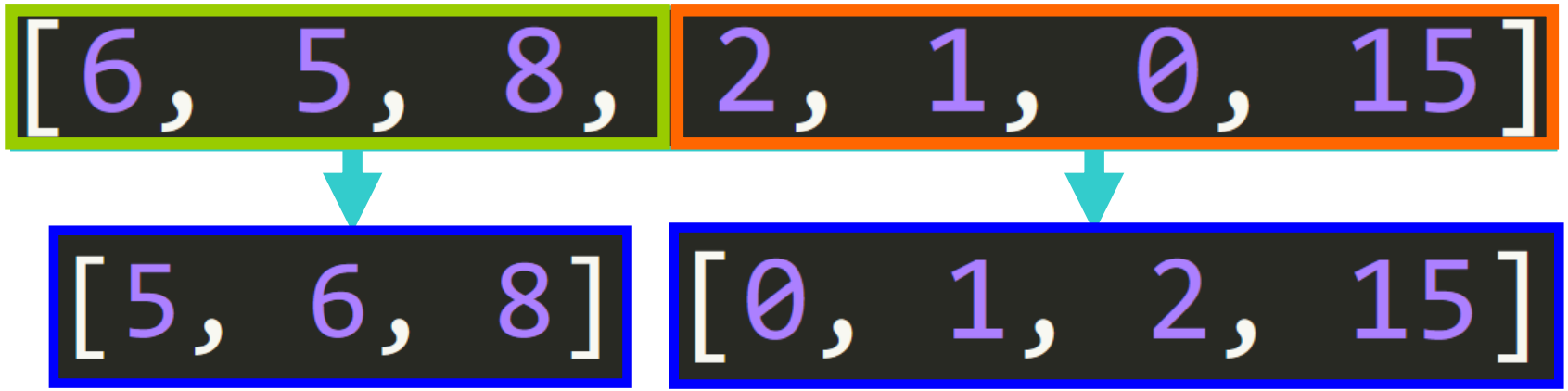


[0, 1]

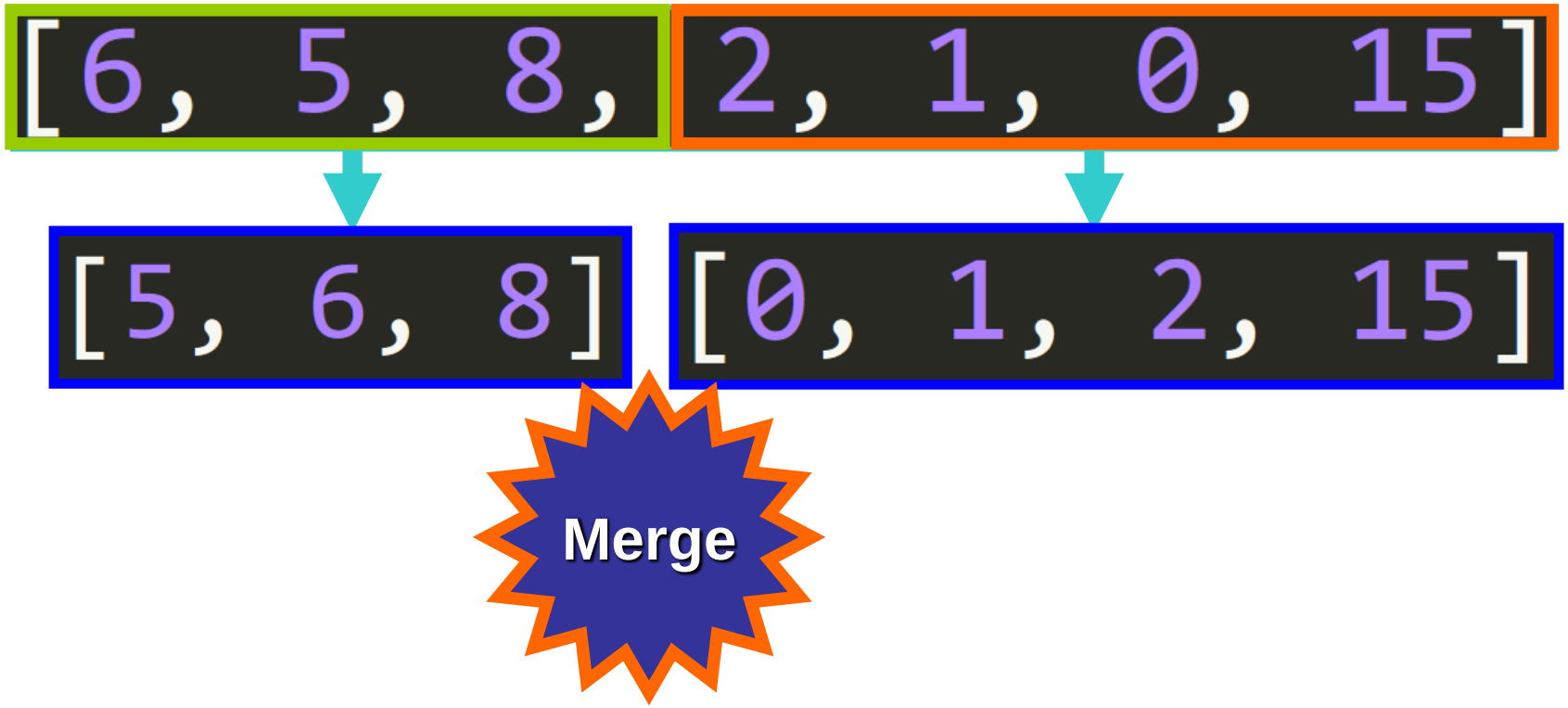

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
Merge
[1, 2]
[0, 15]
[0, 1, 2]

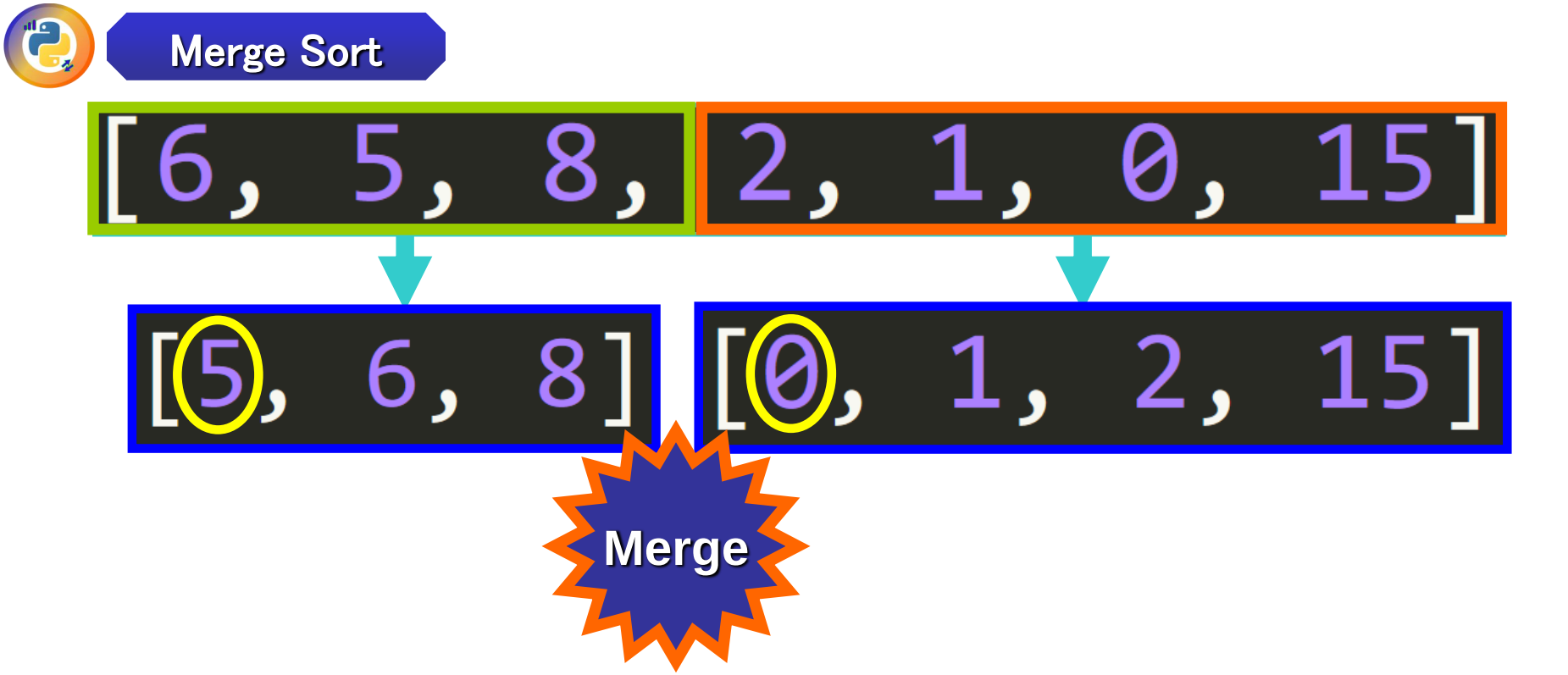

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[2, 1, 0, 15]
Merge
[1, 2]
[0, 15]
[0, 1, 2, 15]

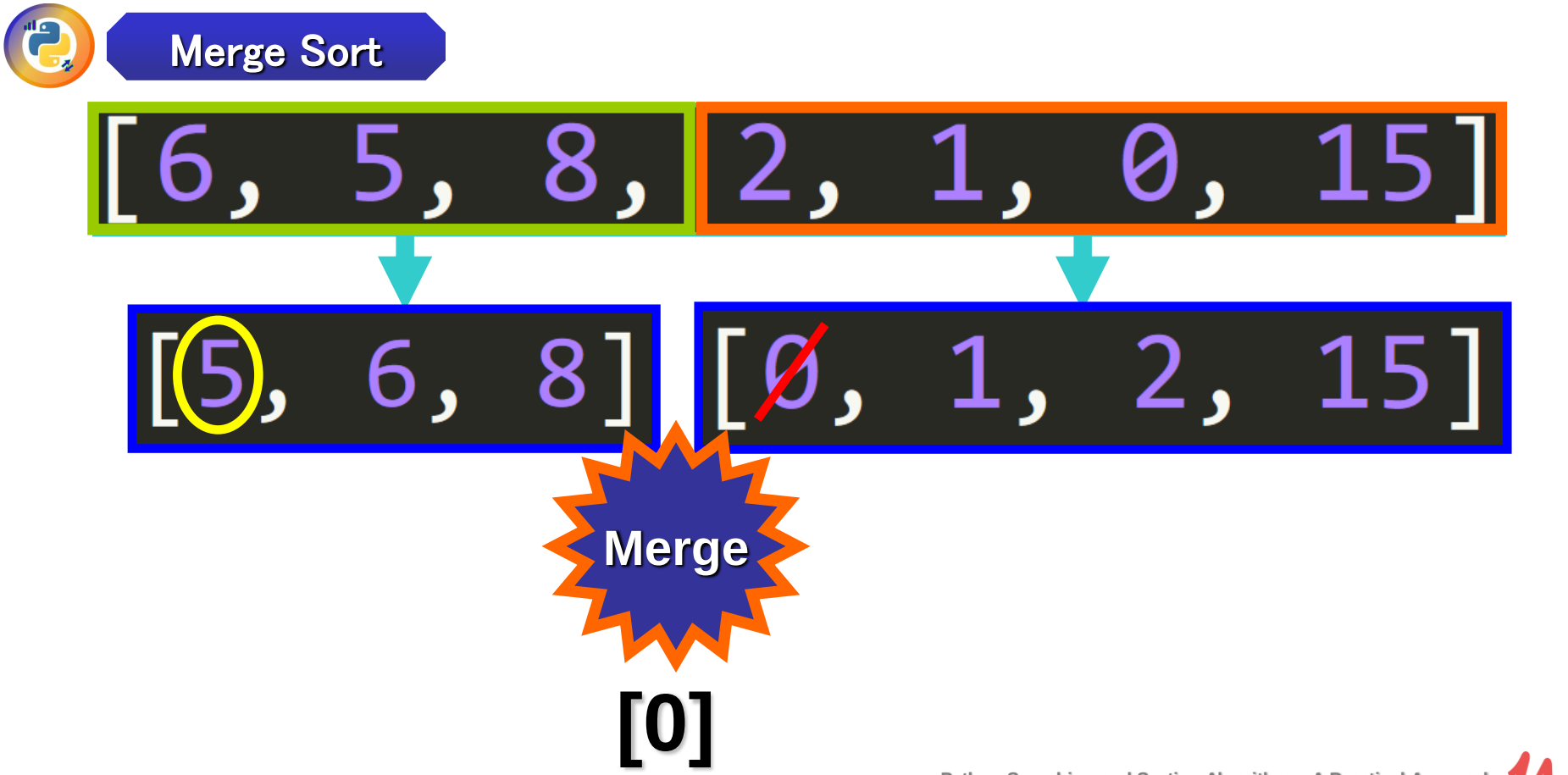
Merge Sort

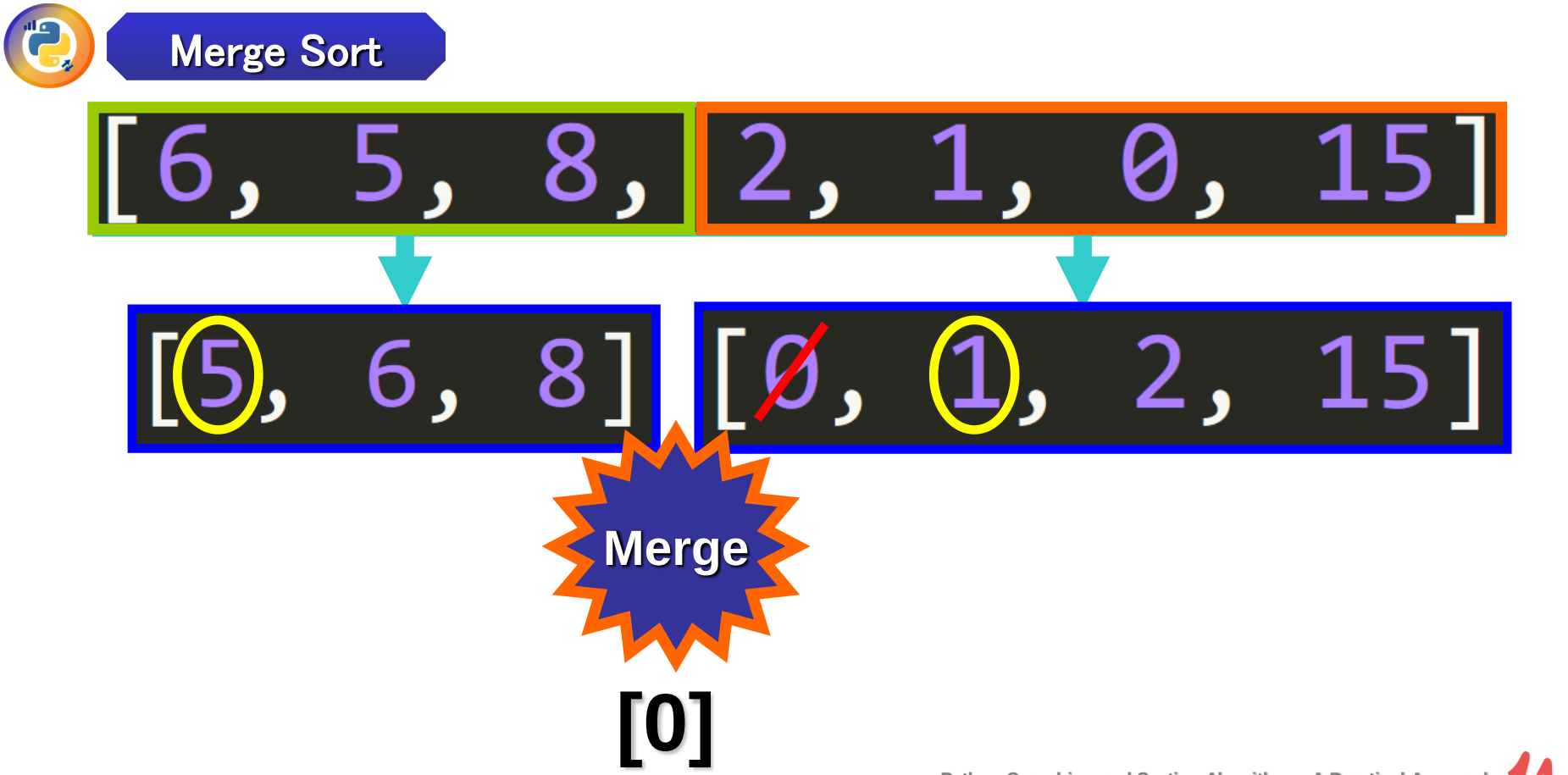


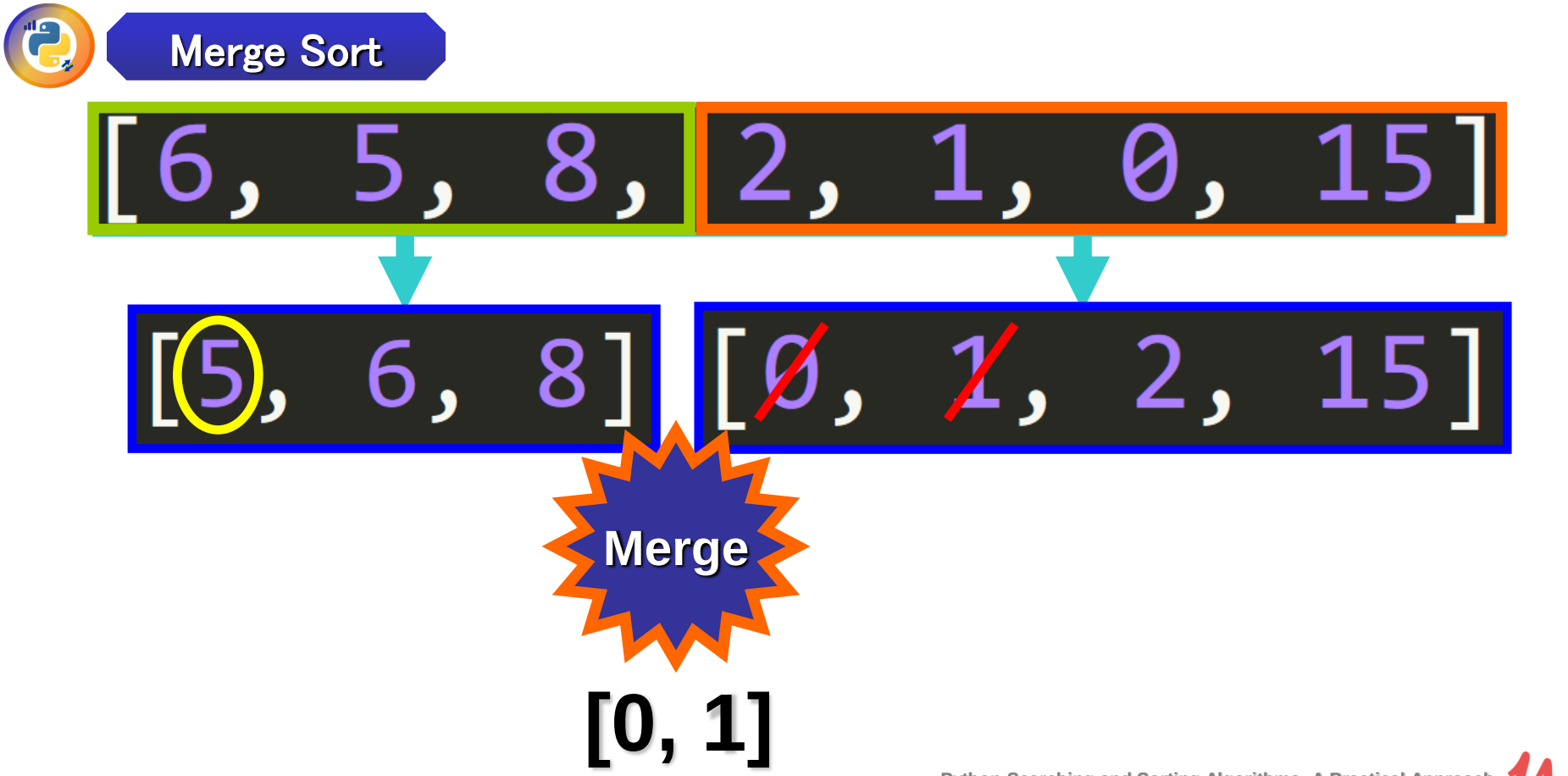
Merge Sort

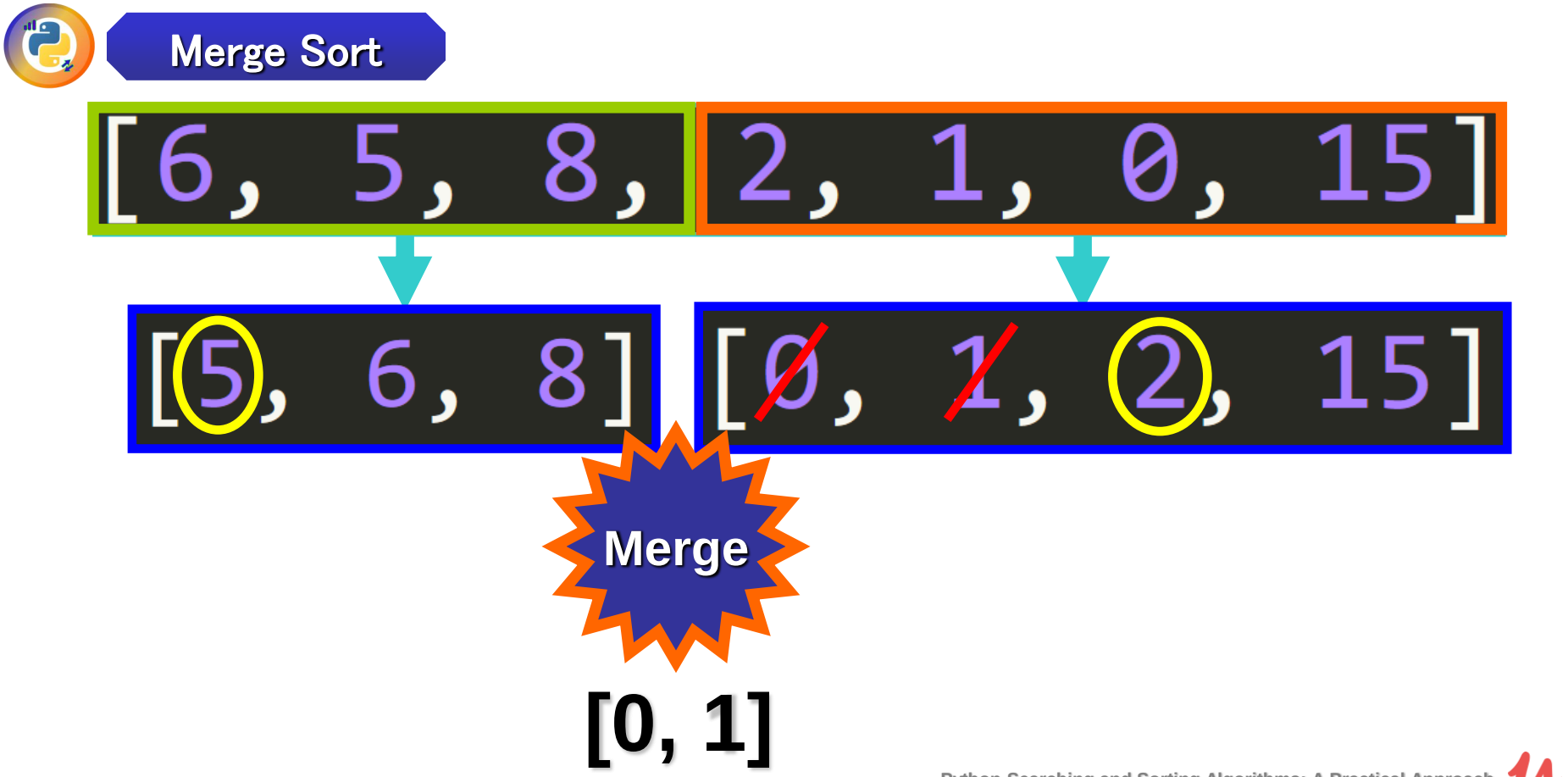


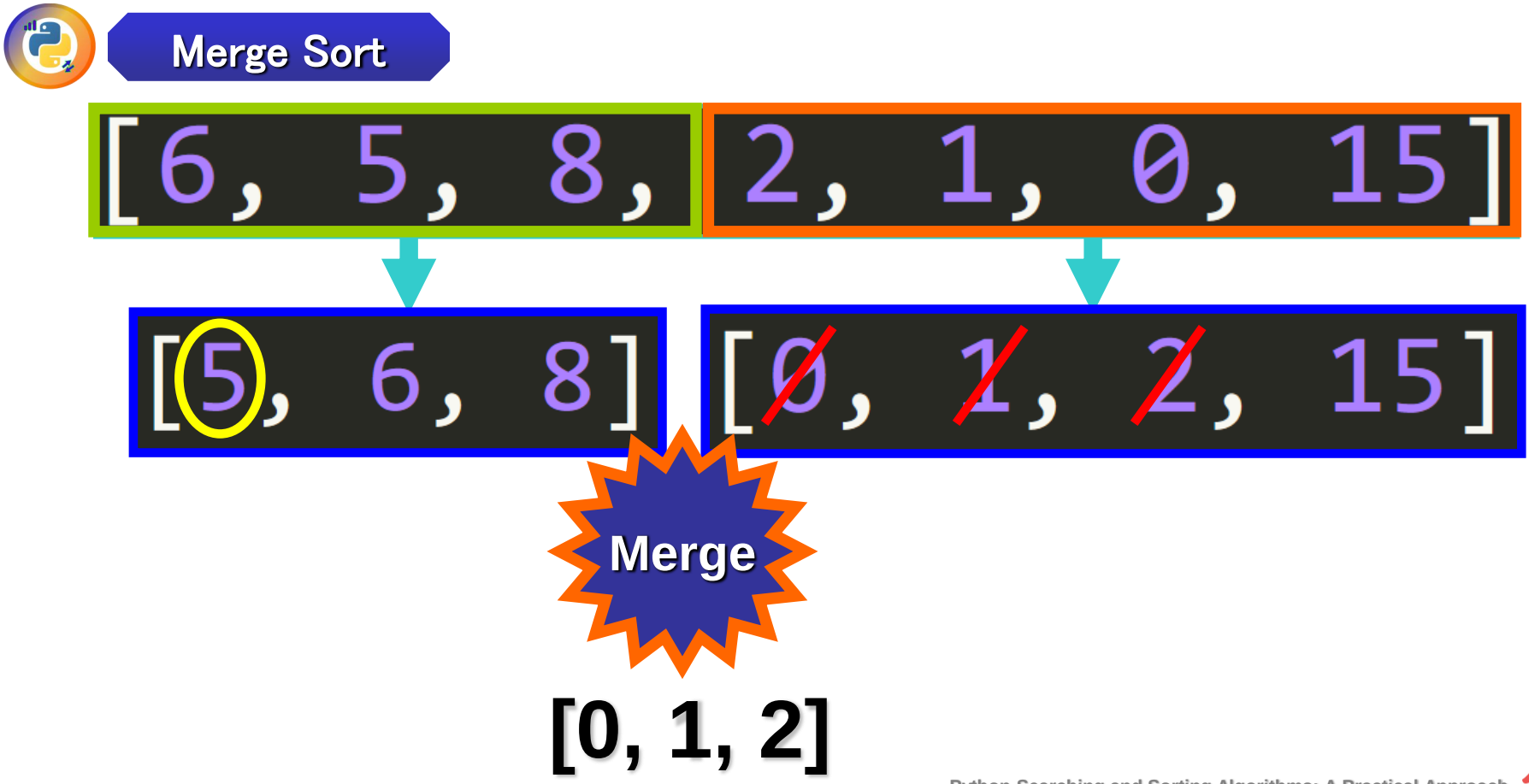

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge

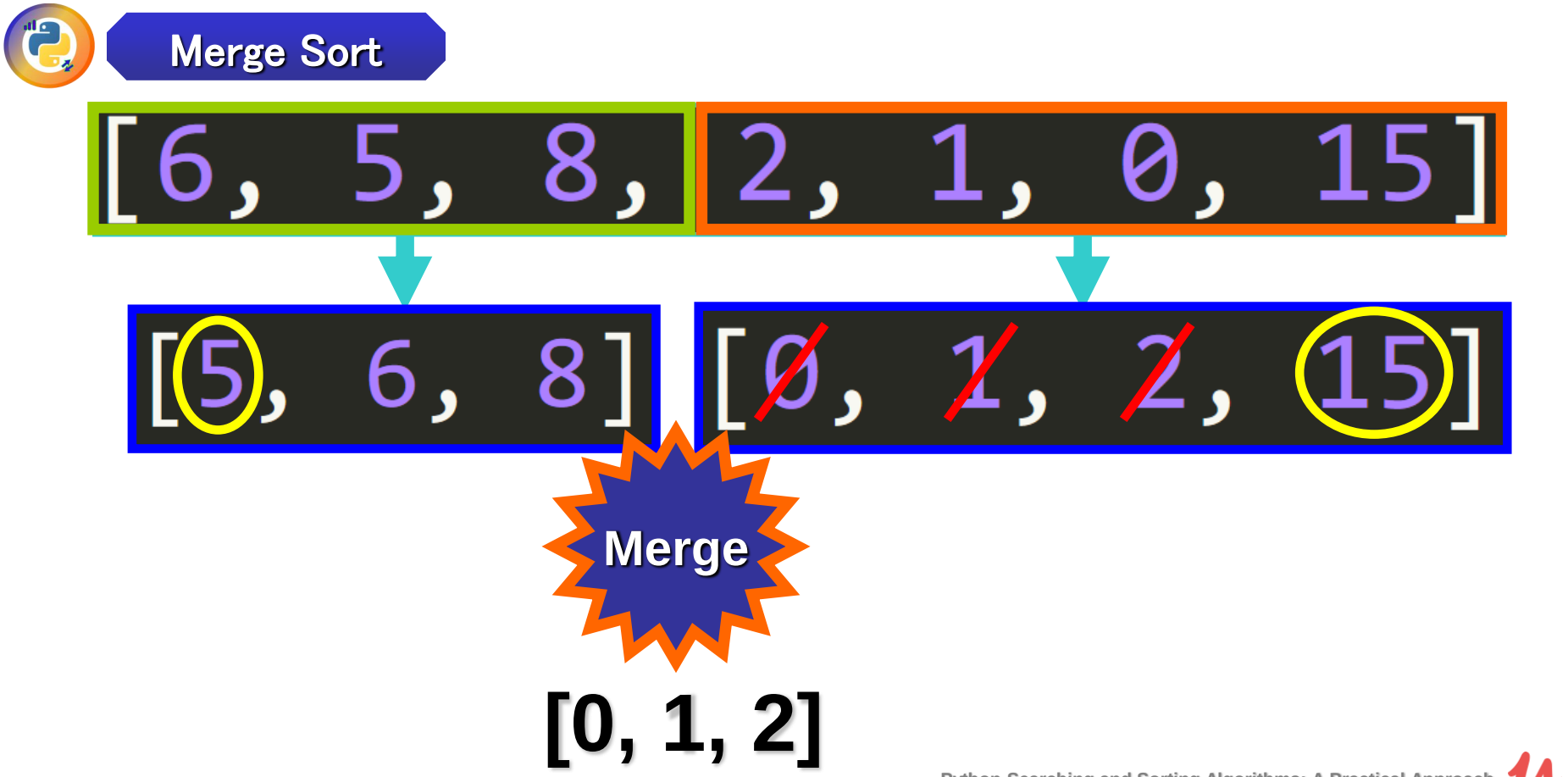

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0]

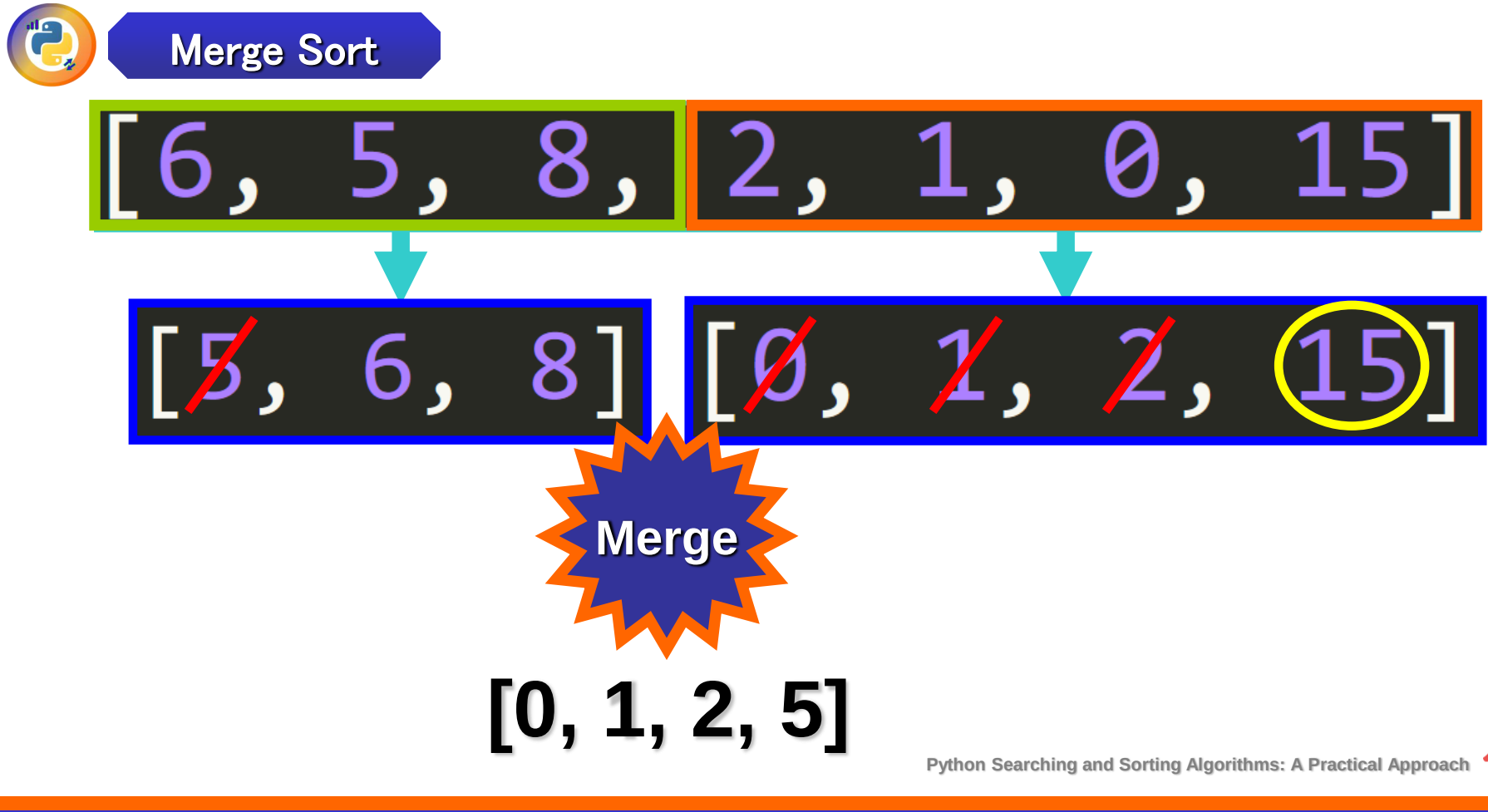

Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0]

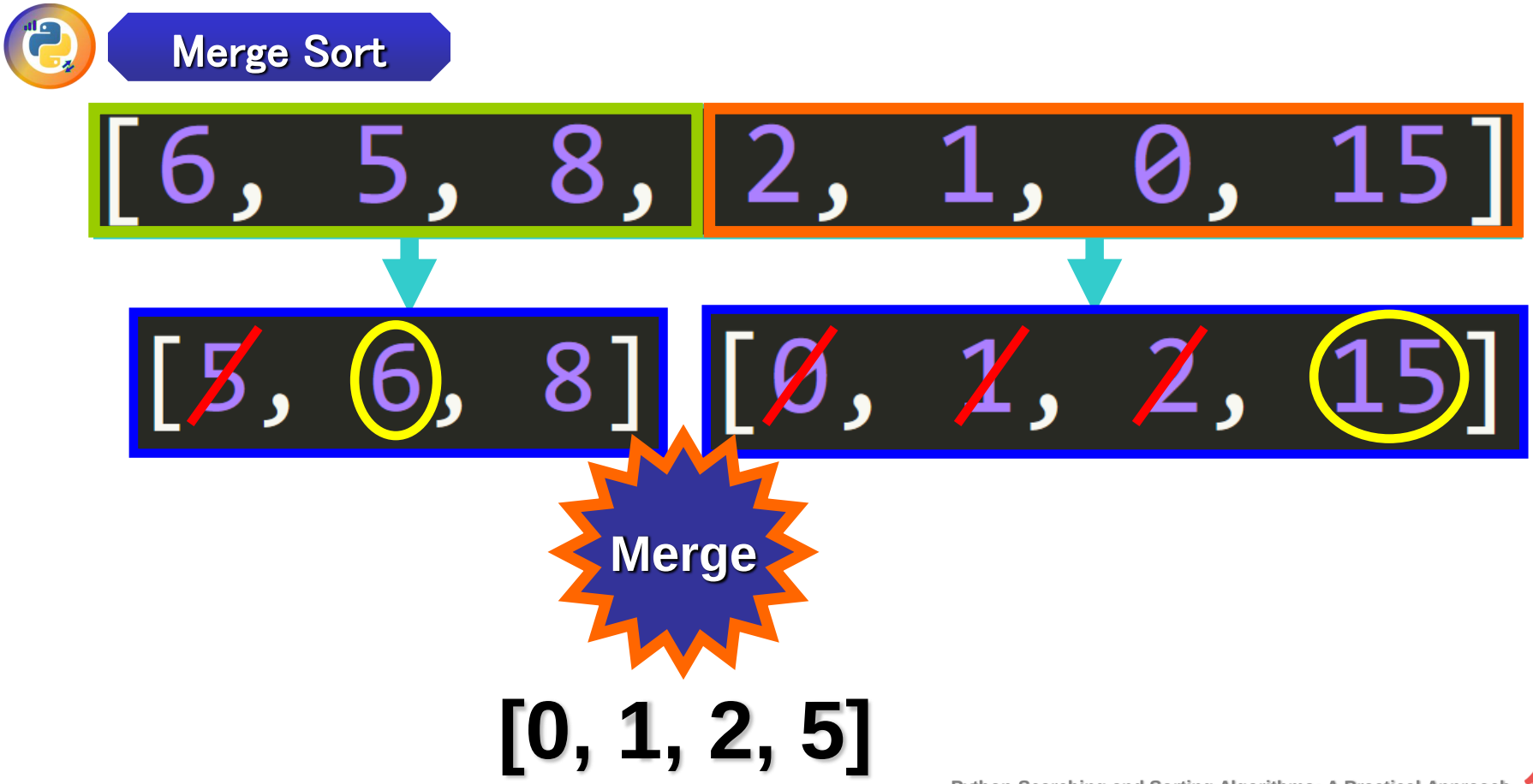

Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1]

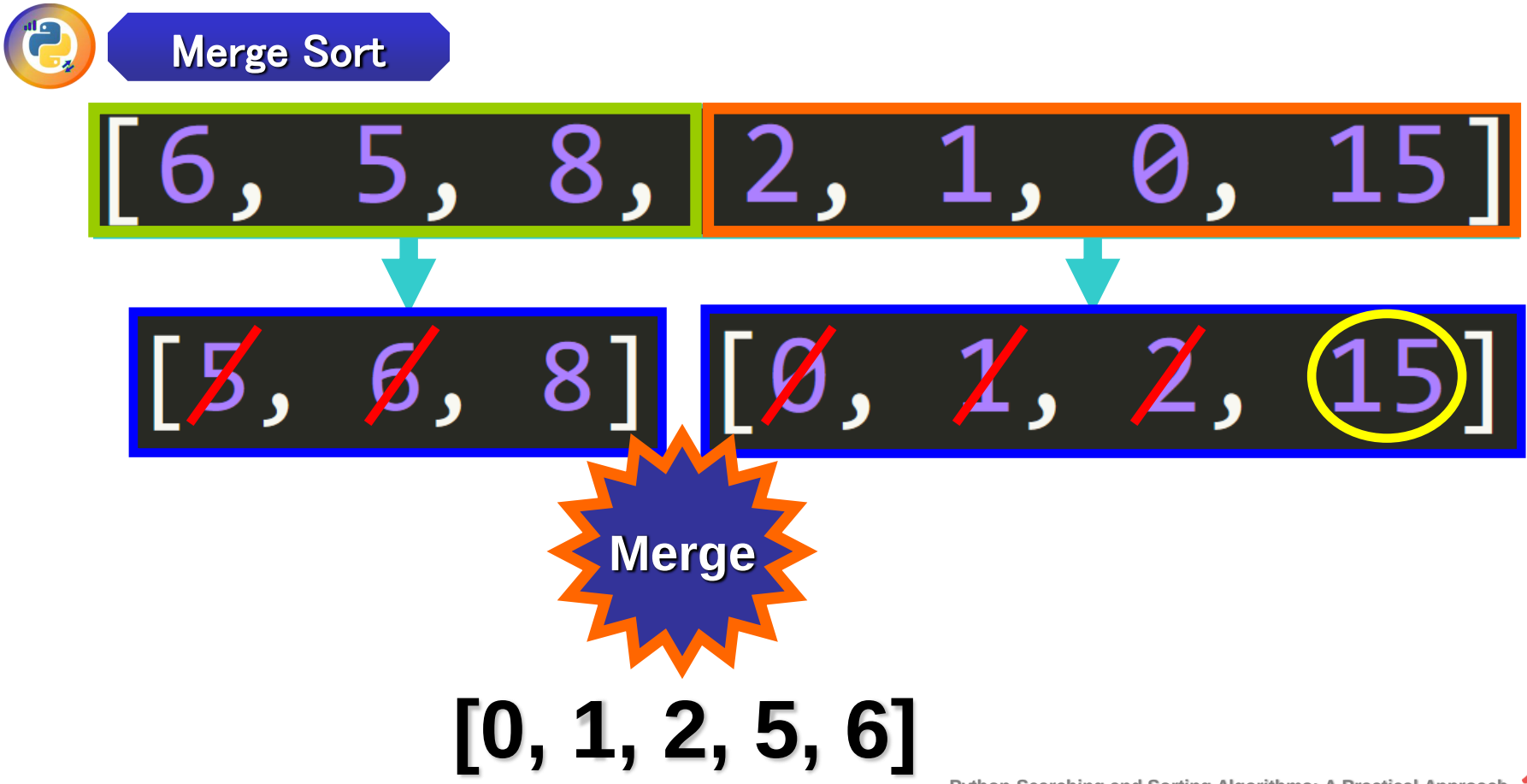

Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1]

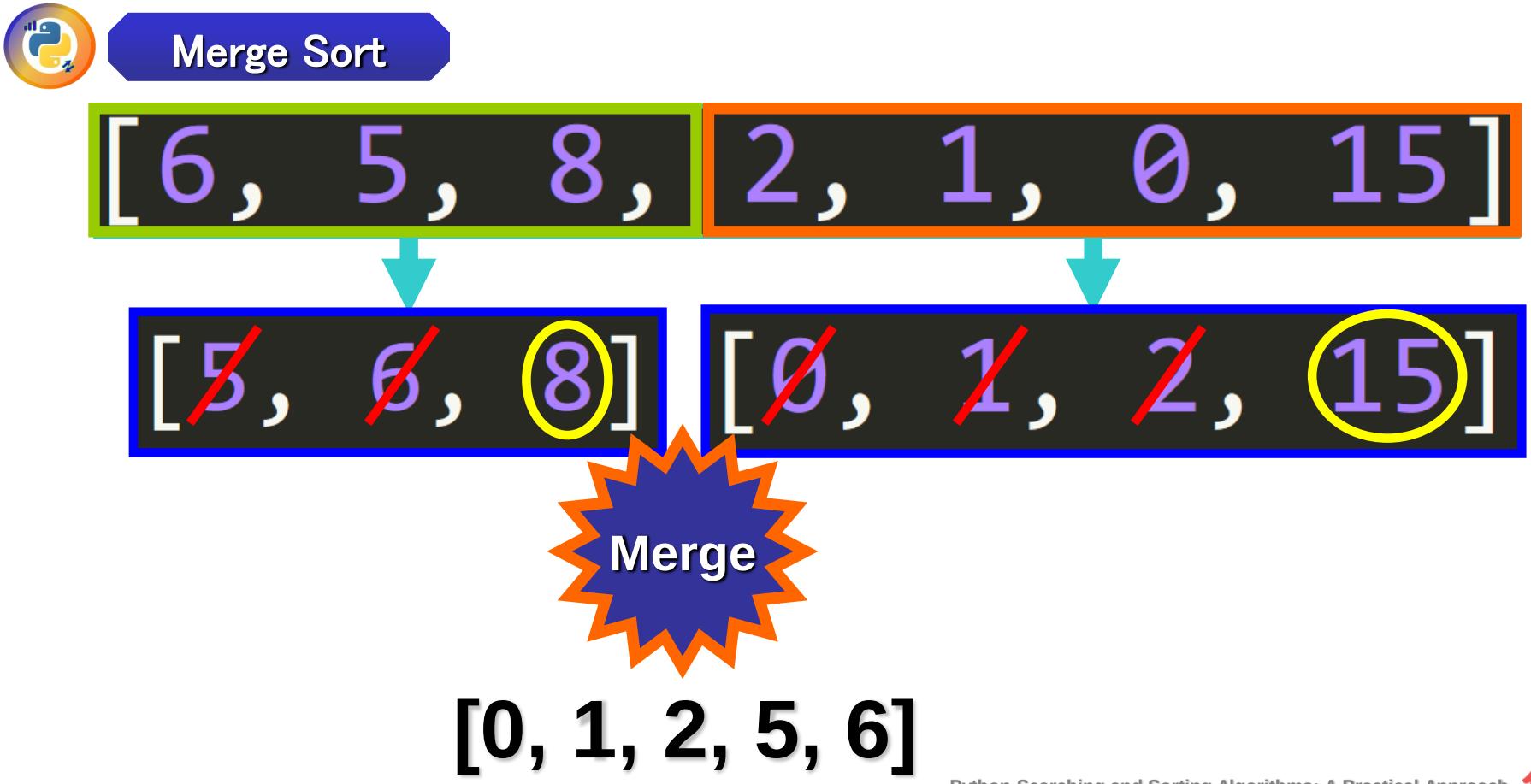

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2]

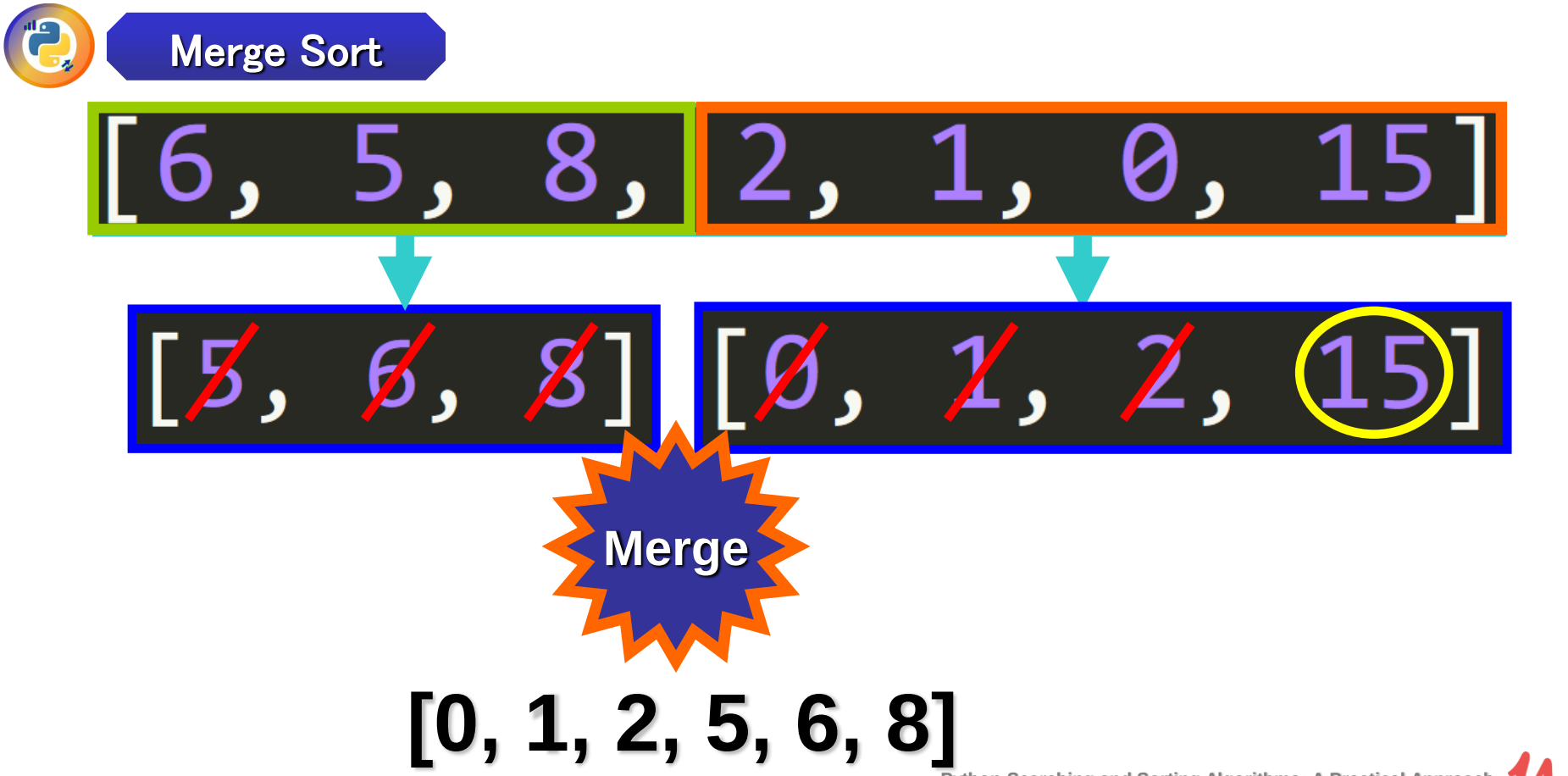

Merge Sort
[6, 5, 8, 2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2]

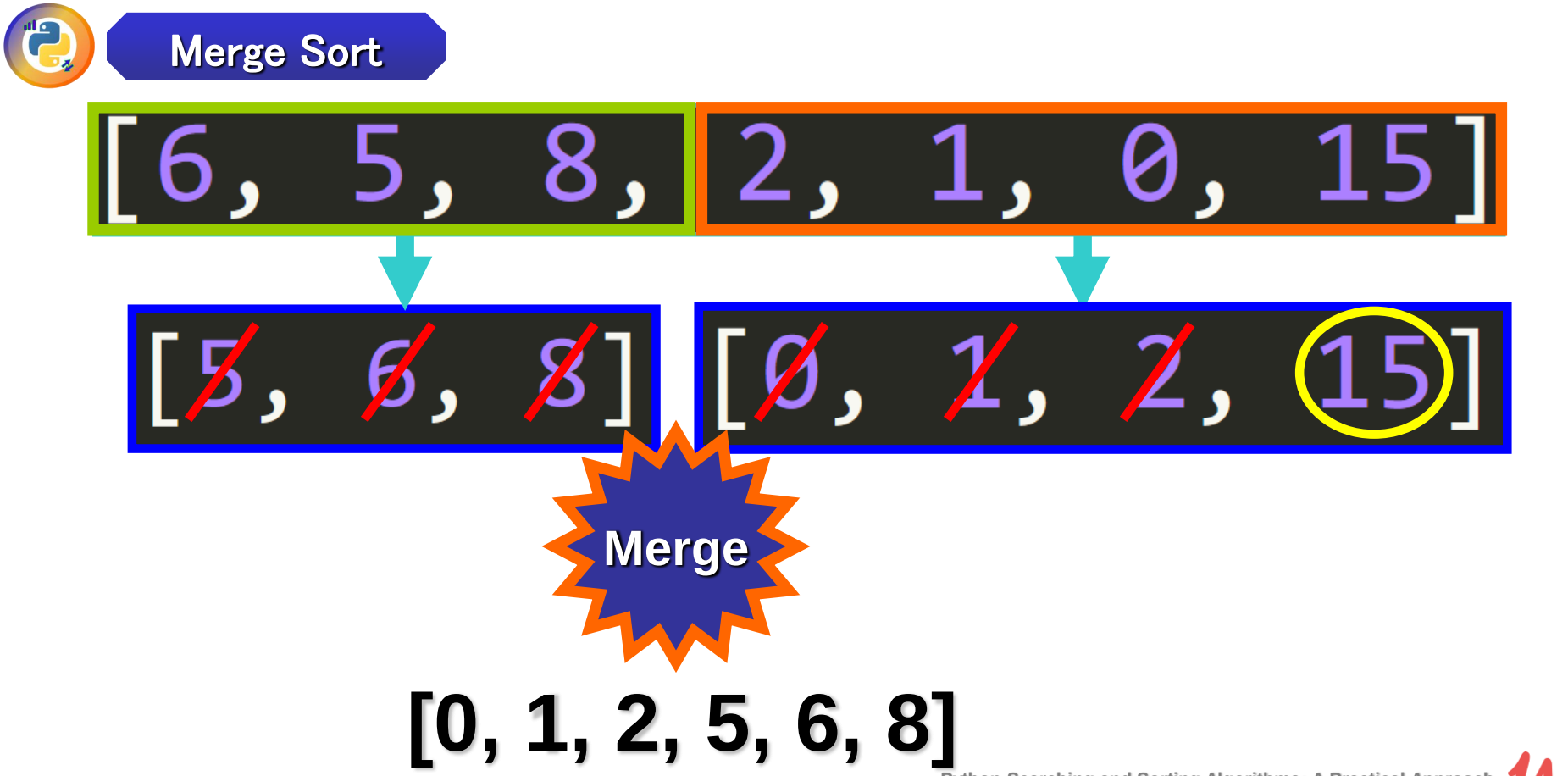

Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5]

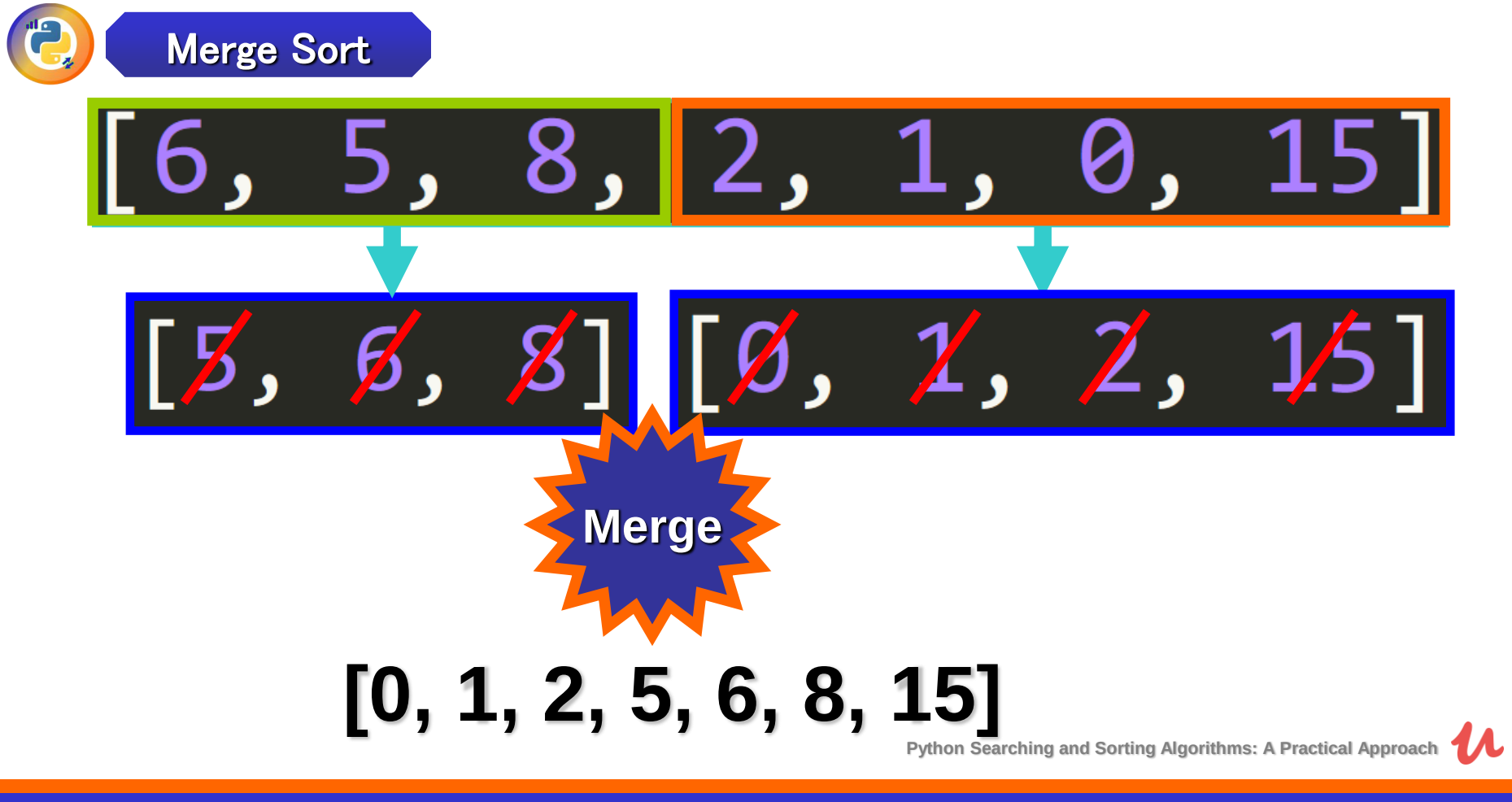

Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5]


Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5, 6]


Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5, 6]


Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5, 6, 8]


Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5, 6, 8]


Merge Sort
[6, 5, 8,
2, 1, 0, 15]
[5, 6, 8]
[0, 1, 2, 15]
Merge
[0, 1, 2, 5, 6, 8, 15]

Merge Sort

```
[0, 1, 2, 5, 6, 8, 15]
```

To the Code!