

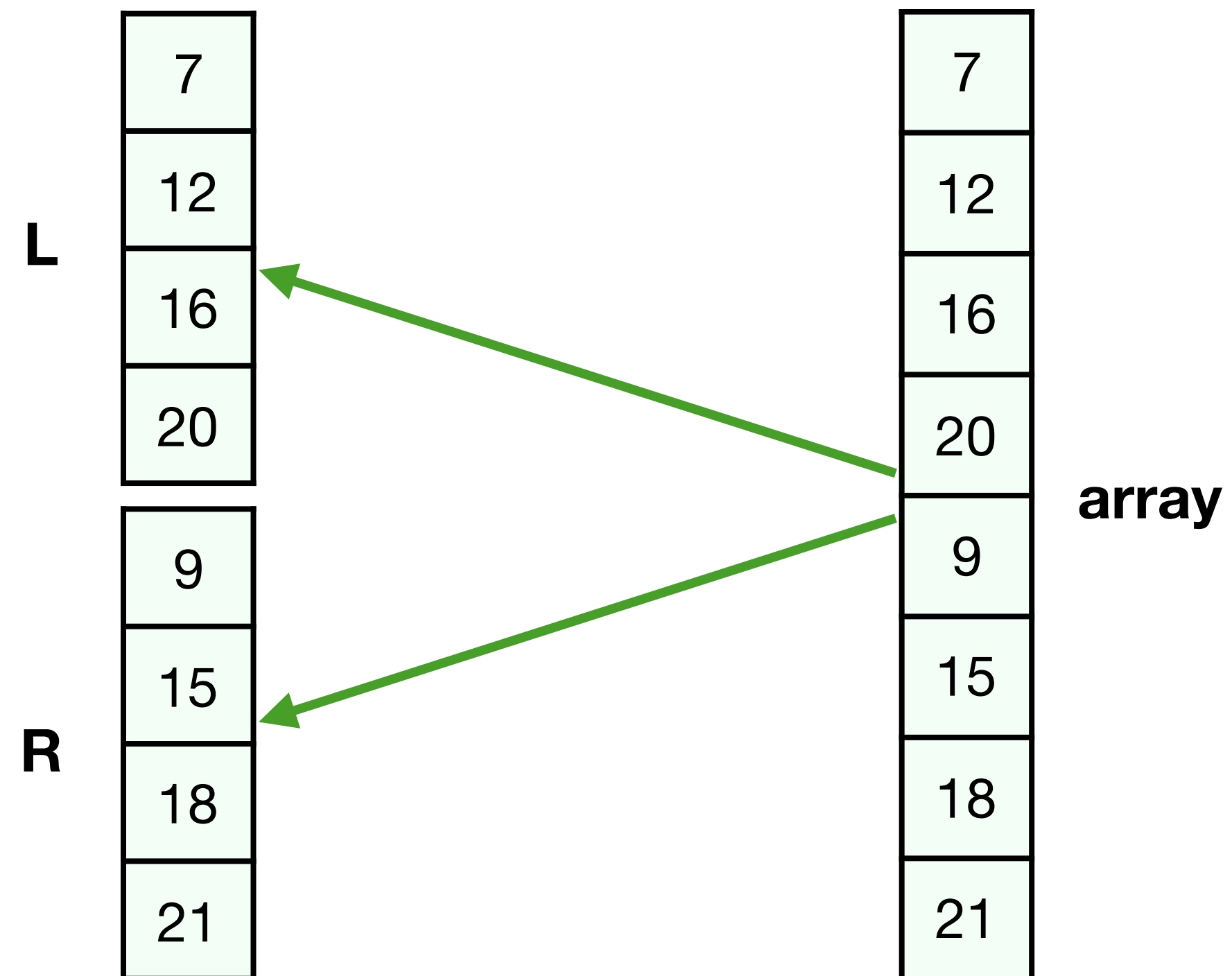
UCSC Silicon Valley Extension

Advanced C Programming

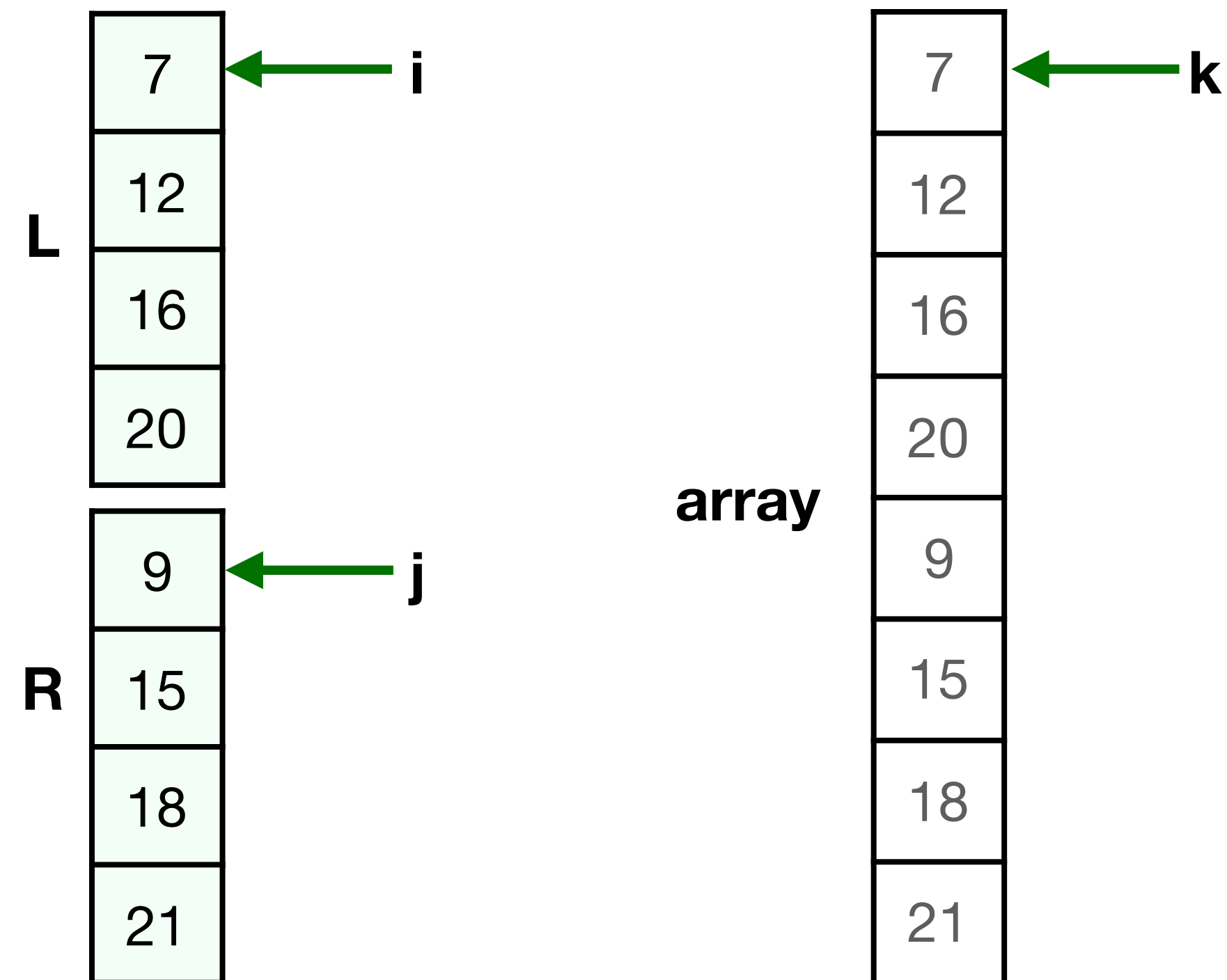
Mergesort

Instructor: Radhika Grover

Merge sort worst case

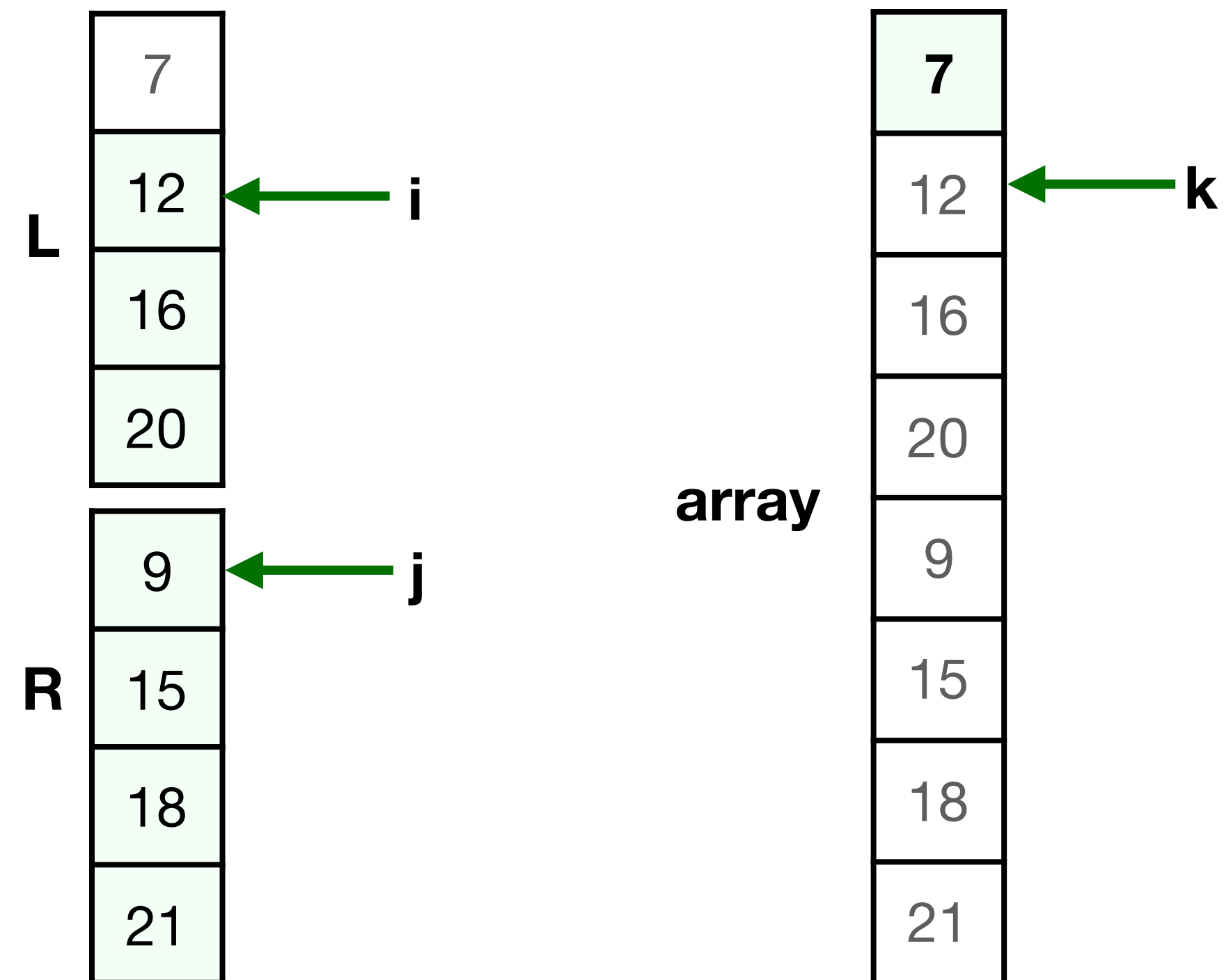


Merge sort worst case



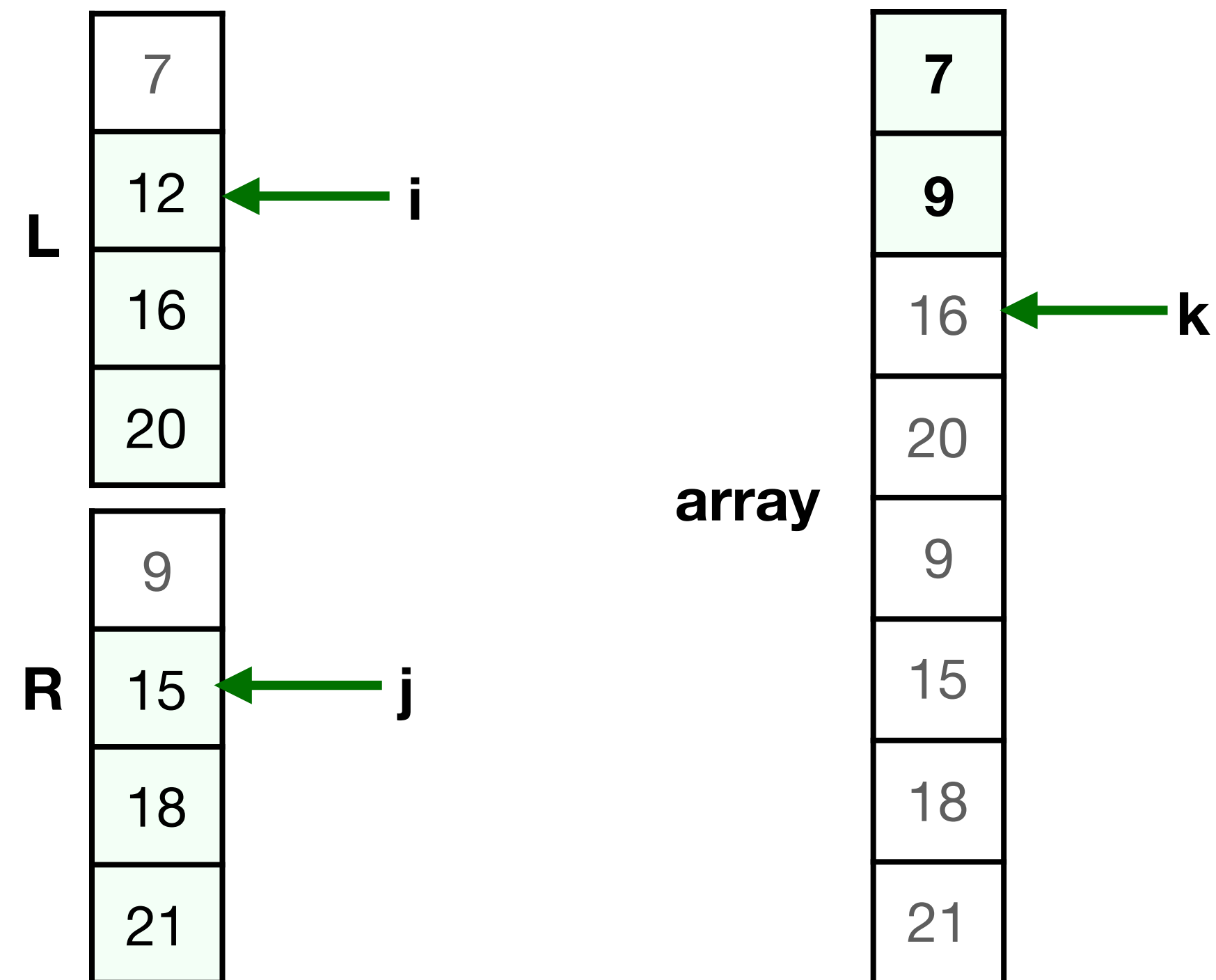
`if(L[i] <= R[j]) ? Yes`

Merge sort worst case



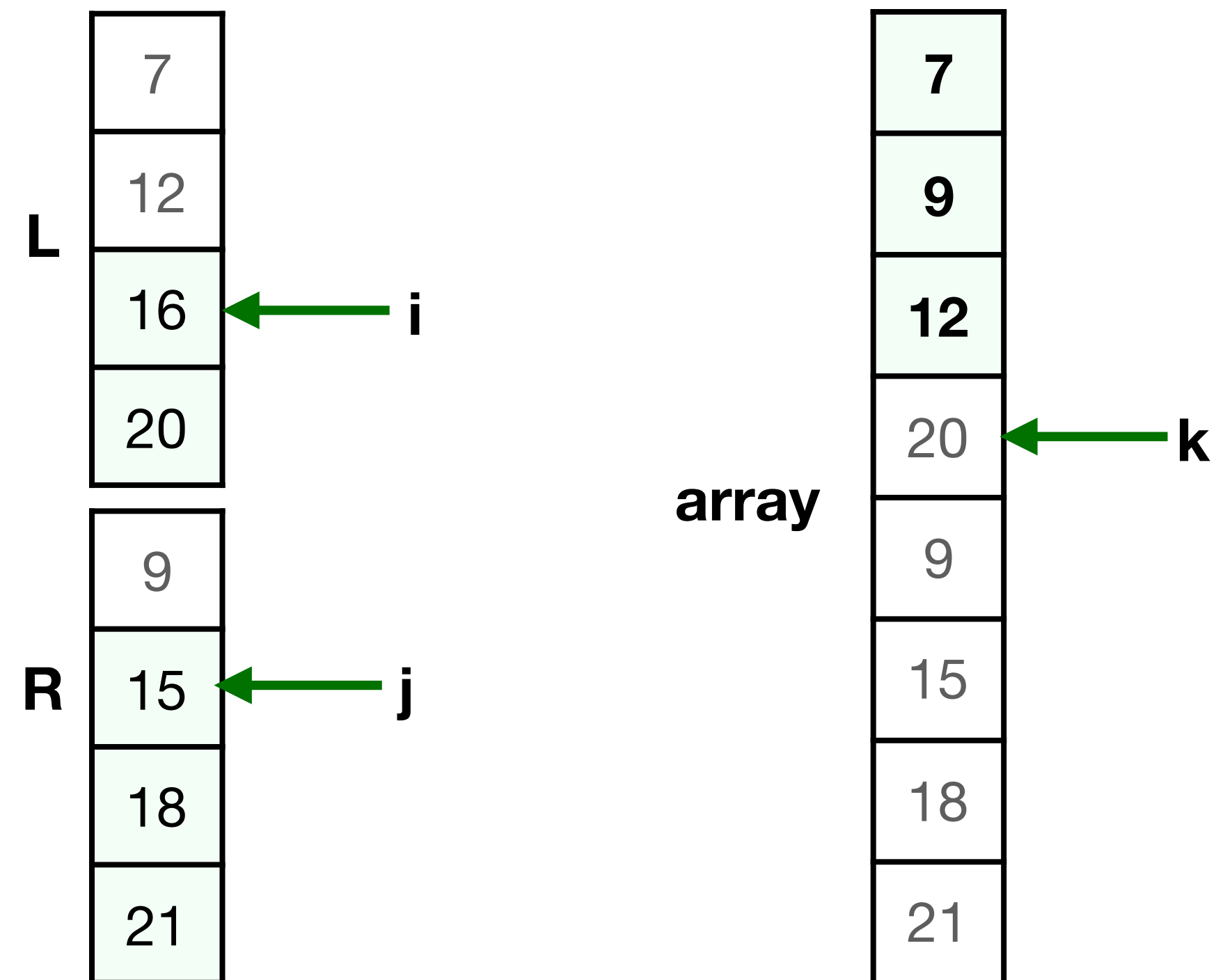
```
array[k] = L[i]
if(L[i] <= R[j]) ? No
```

Merge sort worst case



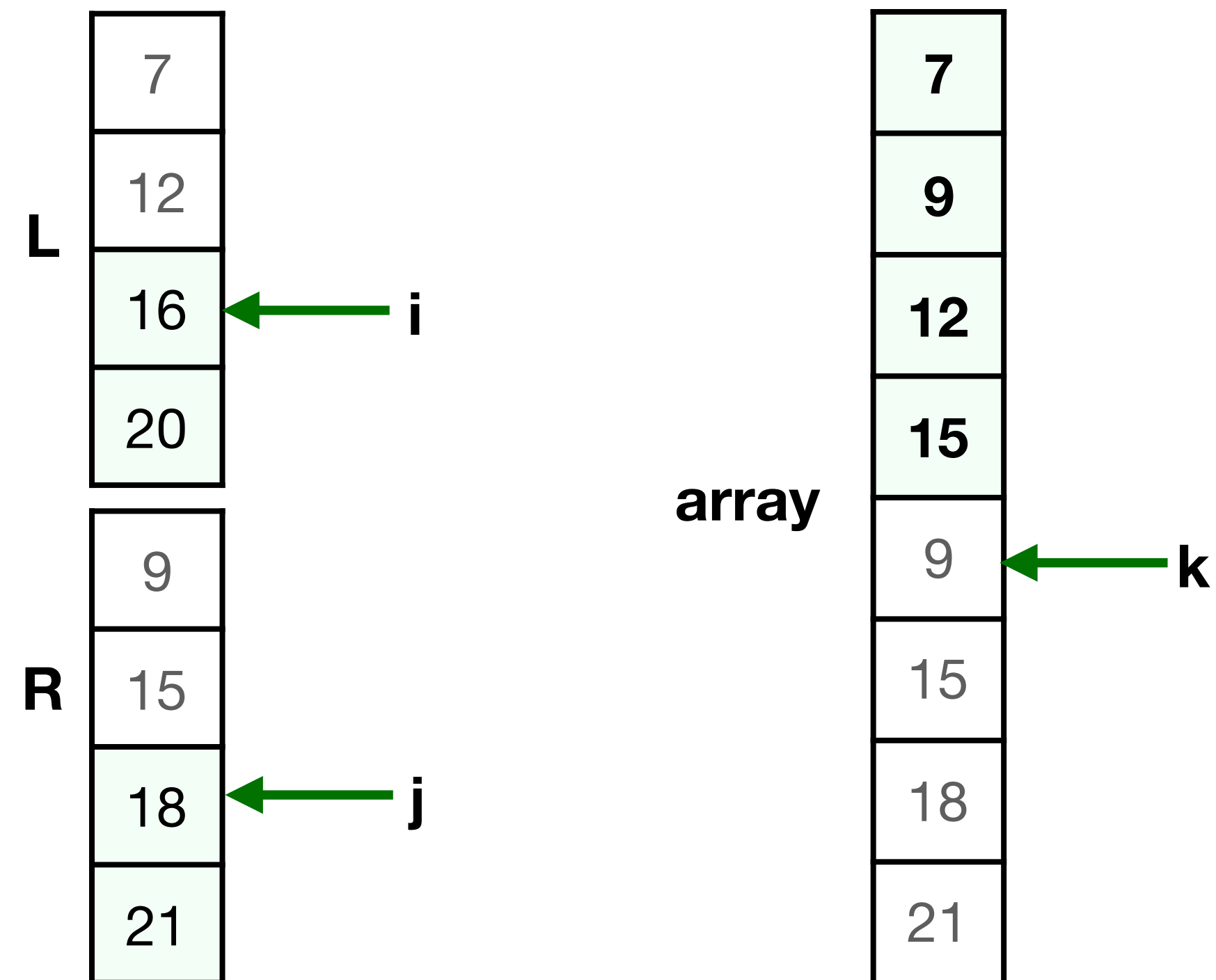
```
array[k] = R[j]
if(L[i] <= R[j]) ? Yes
```

Merge sort worst case



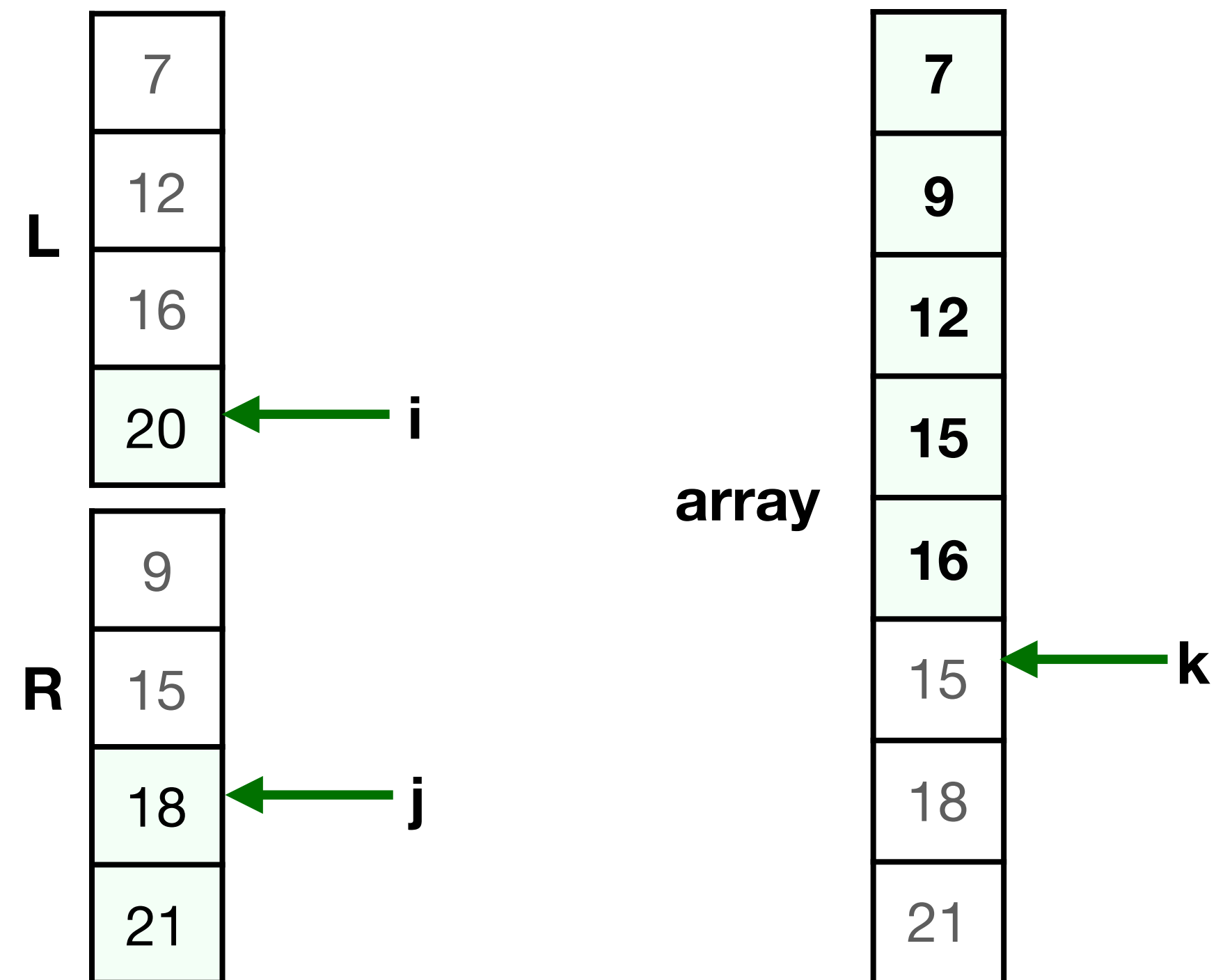
```
array[k] = L[j]
if(L[i] <= R[j]) ? No
```

Merge sort worst case



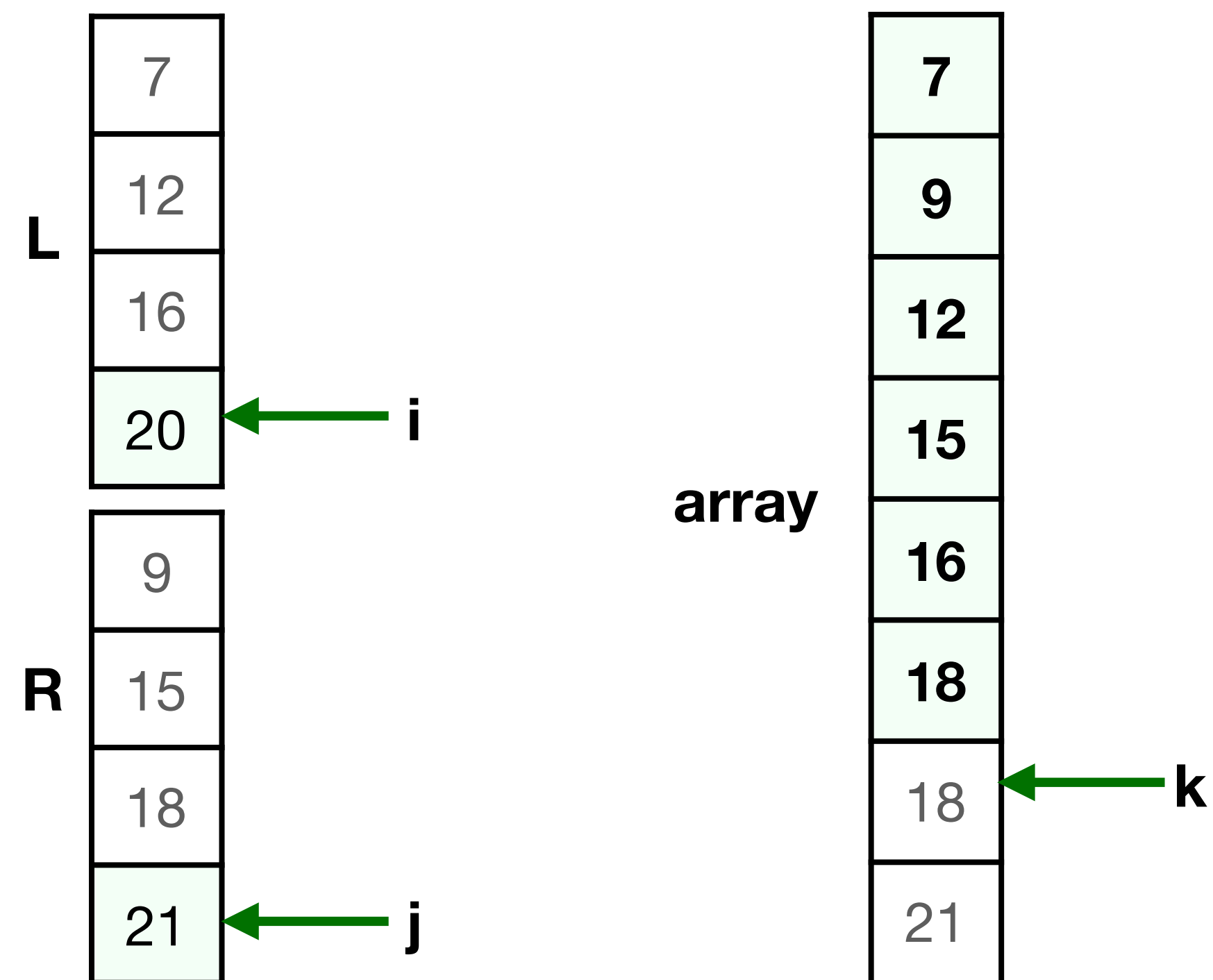
```
array[k] = R[j]
if(L[i] <= R[j]) ? yes
```

Merge sort worst case



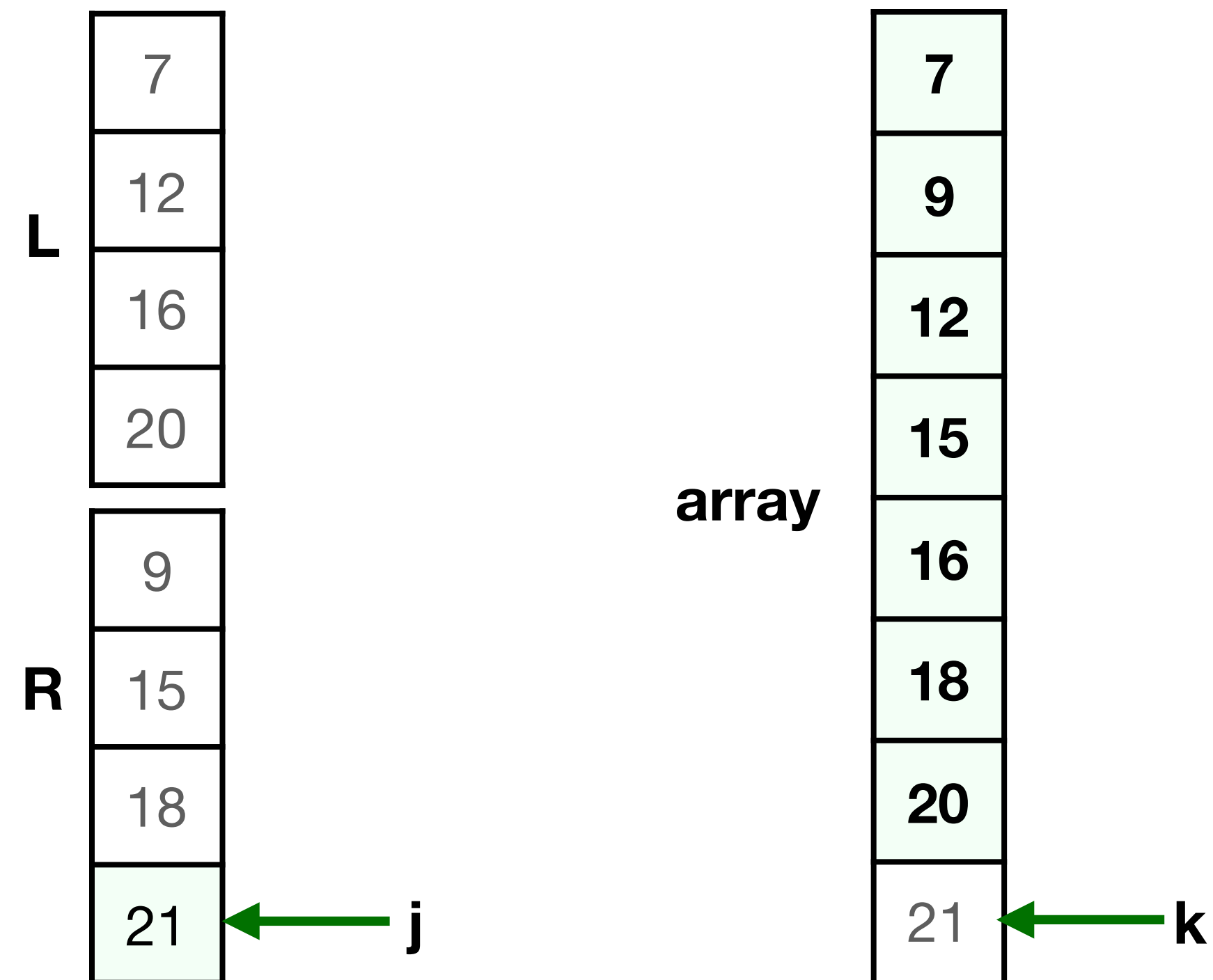
```
array[k] = L[j]
if(L[i] <= R[j]) ? No
```


Merge sort worst case



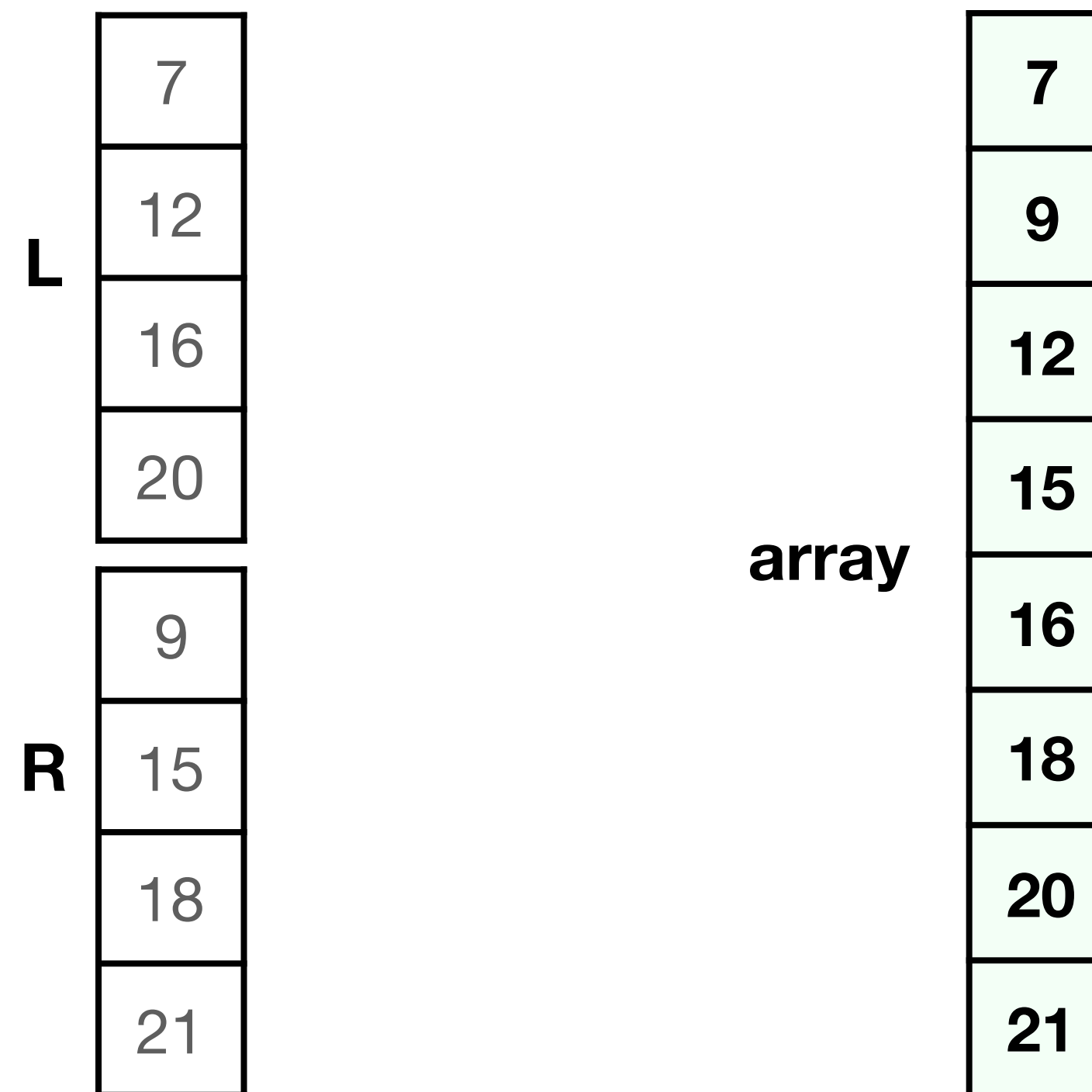
```
array[k] = R[j]
if(L[i] <= R[j]) ? yes
```

Merge sort worst case



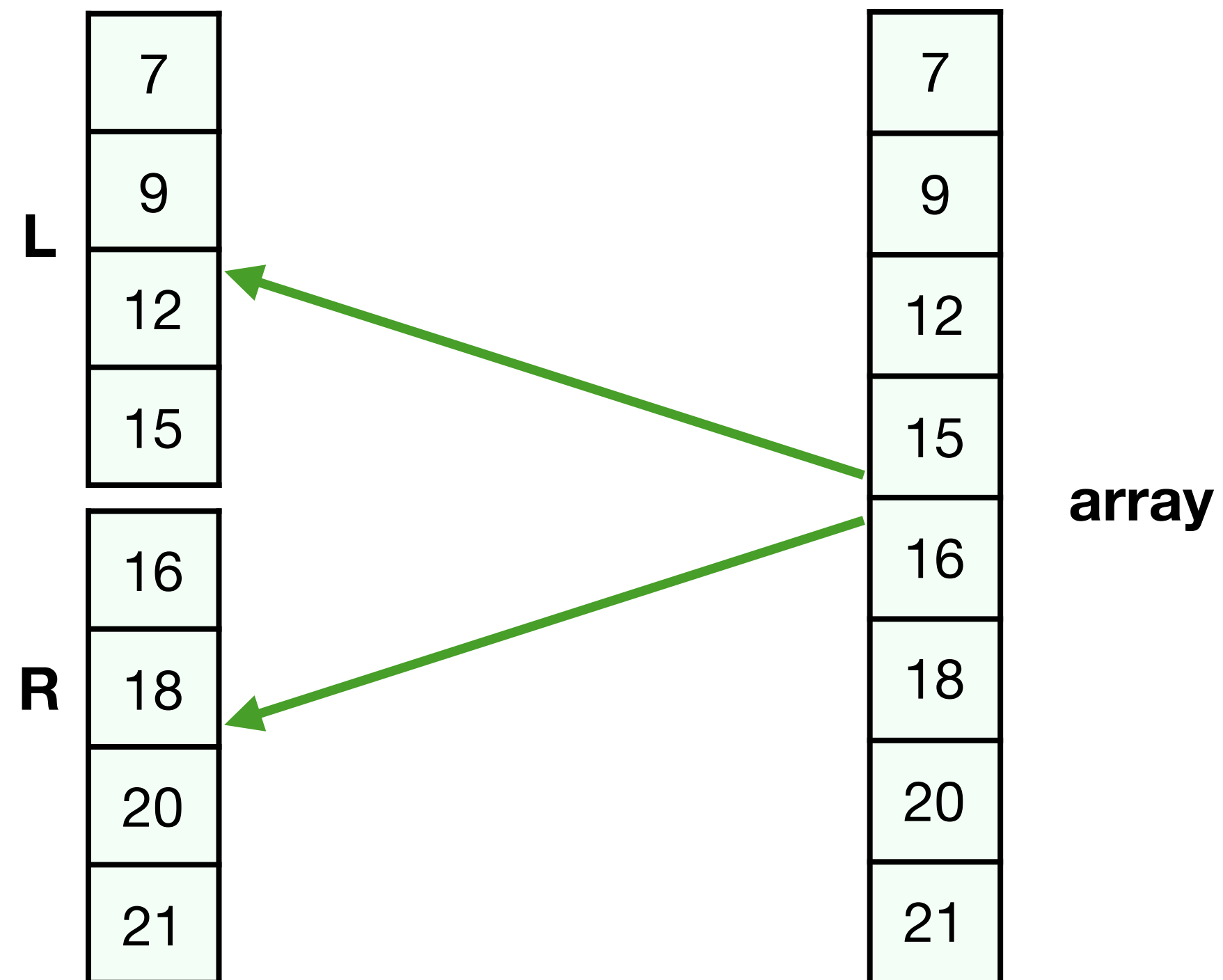
$$\text{array}[k] = L[j]$$

Merge sort worst case

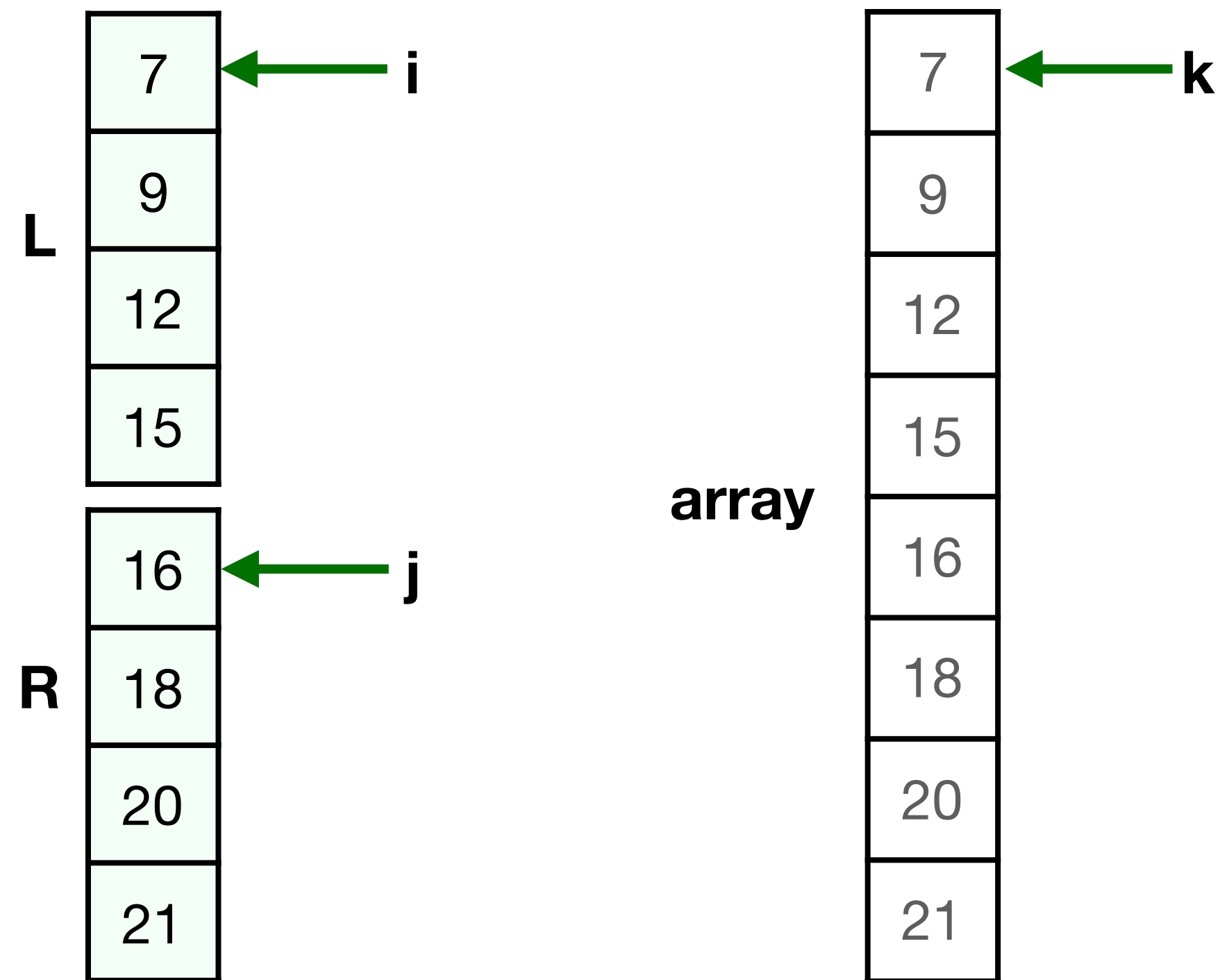


$\text{array}[k] = R[j]$

Merge sort best case

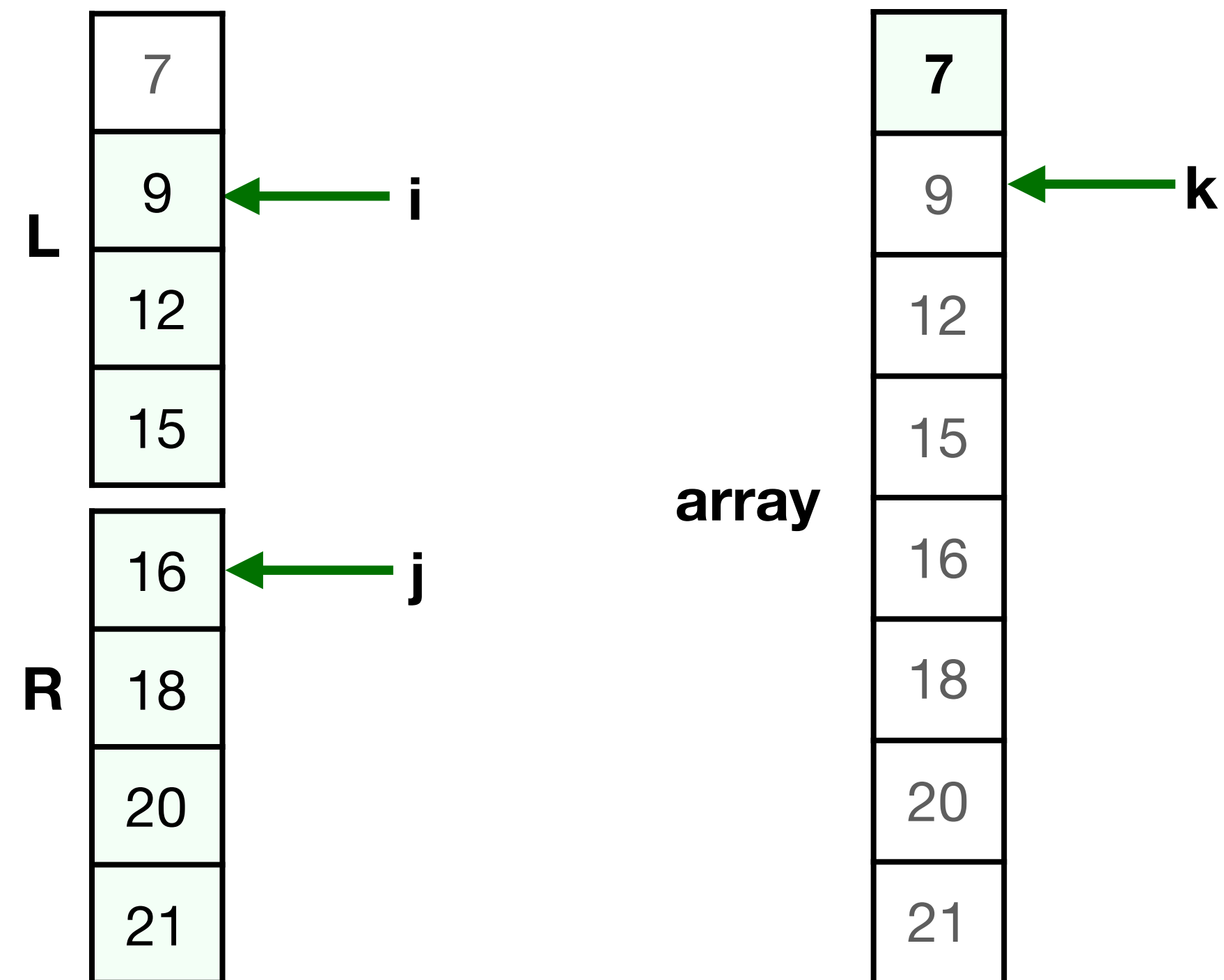


Merge sort best case



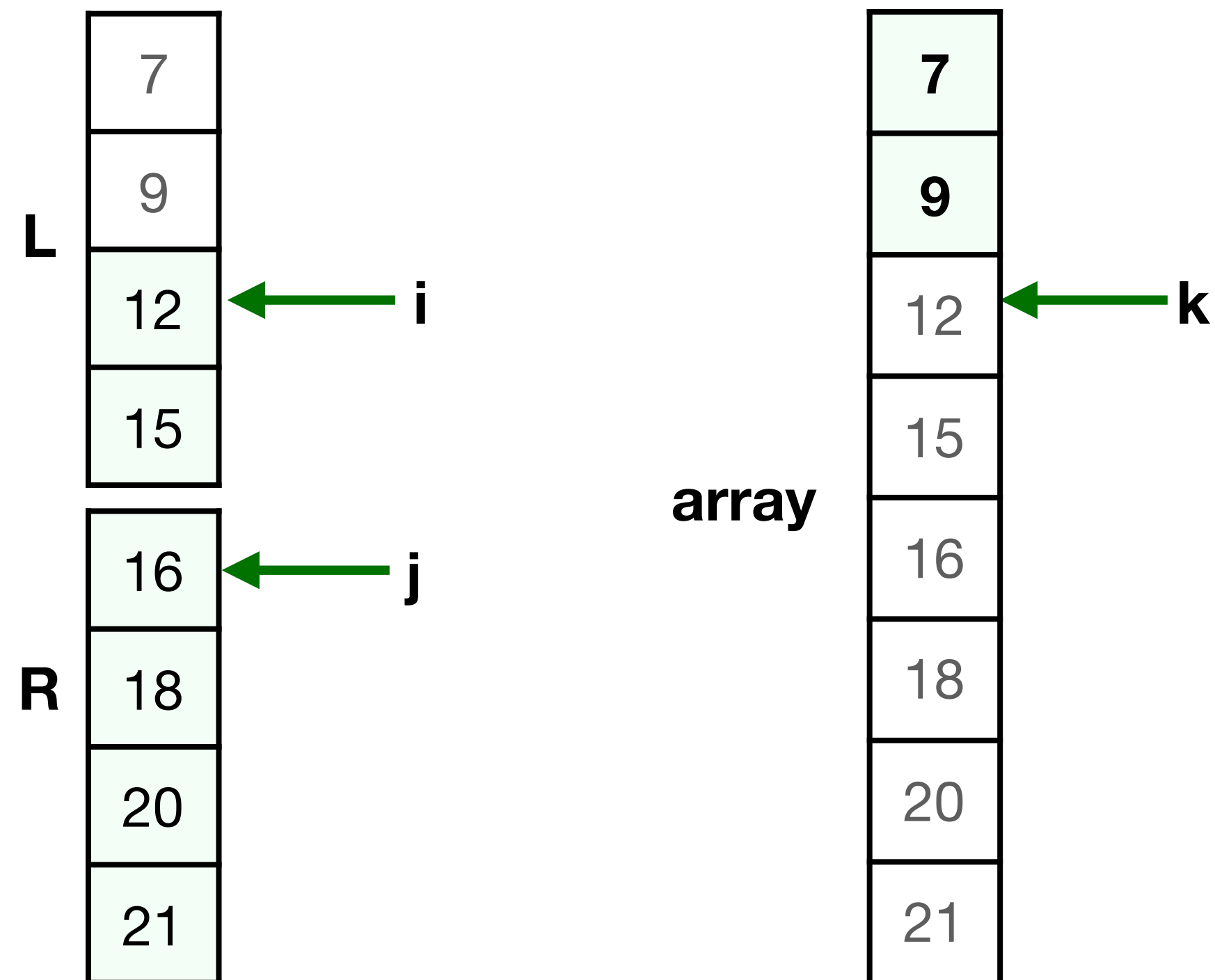
`if(L[i] <= R[j]) ? Yes`

Merge sort best case



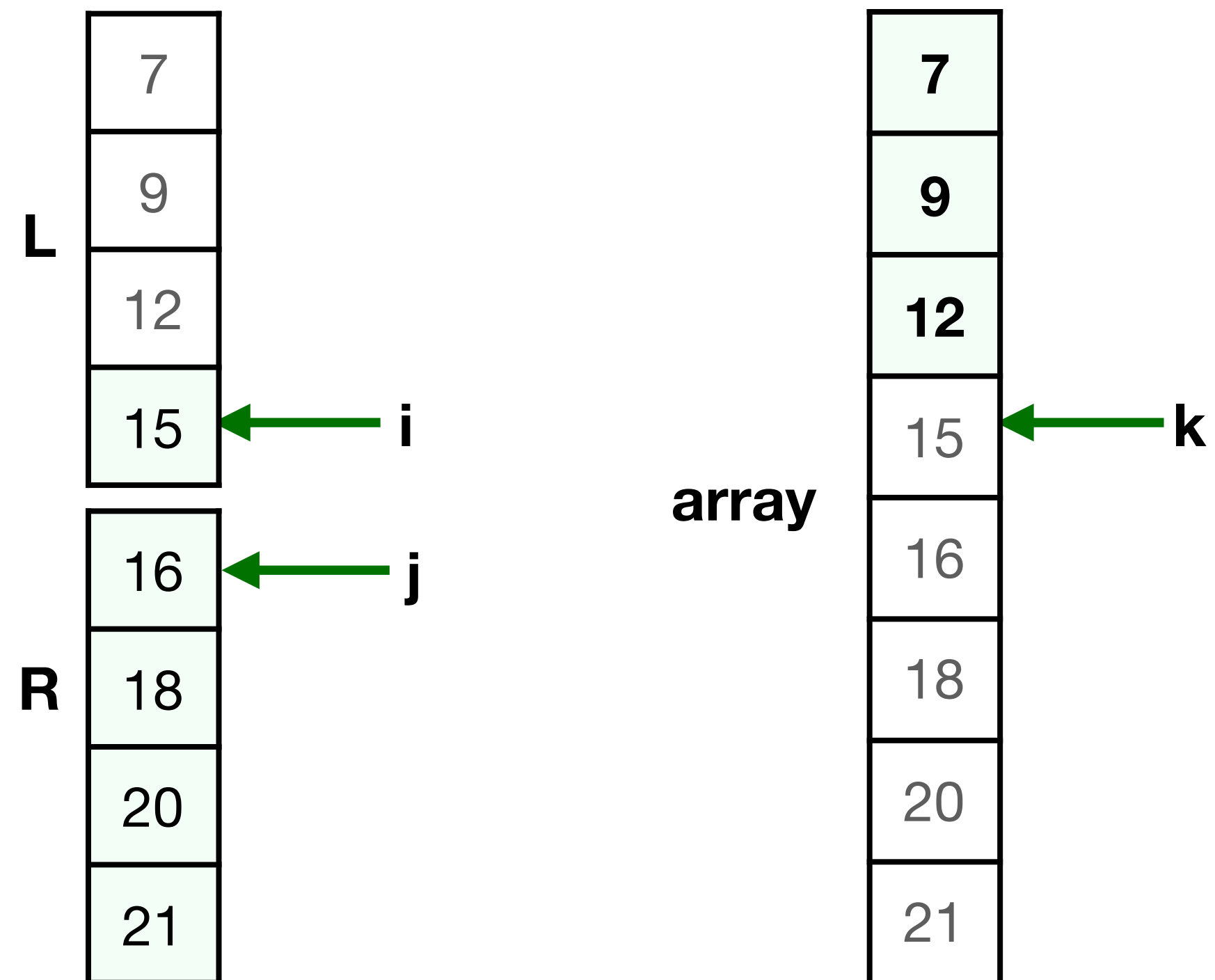
```
array[k] = L[i]
if(L[i]<= R[j]) ? Yes
```

Merge sort best case



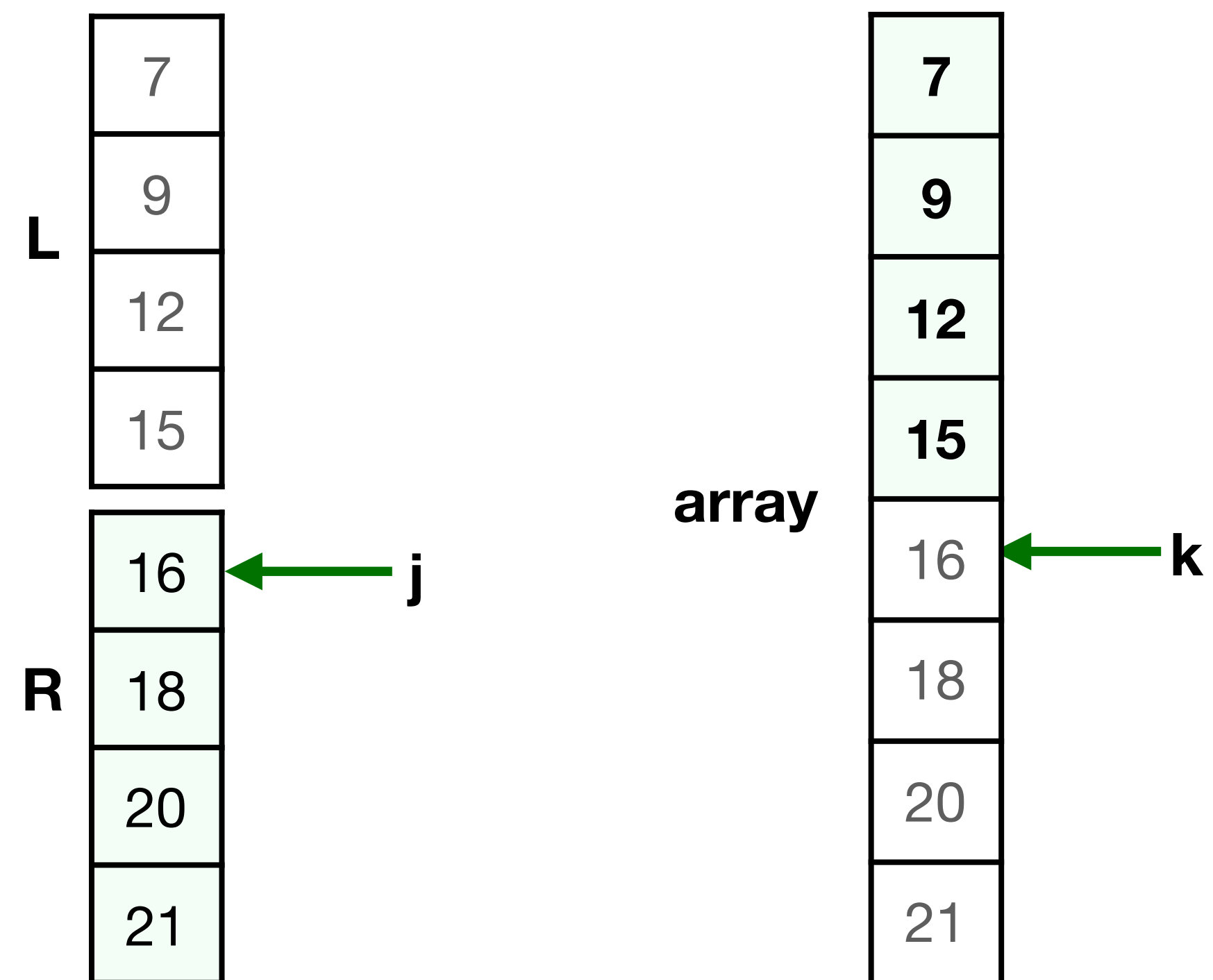
```
array[k] = L[i]
if(L[i] <= R[j]) ? Yes
```

Merge sort best case



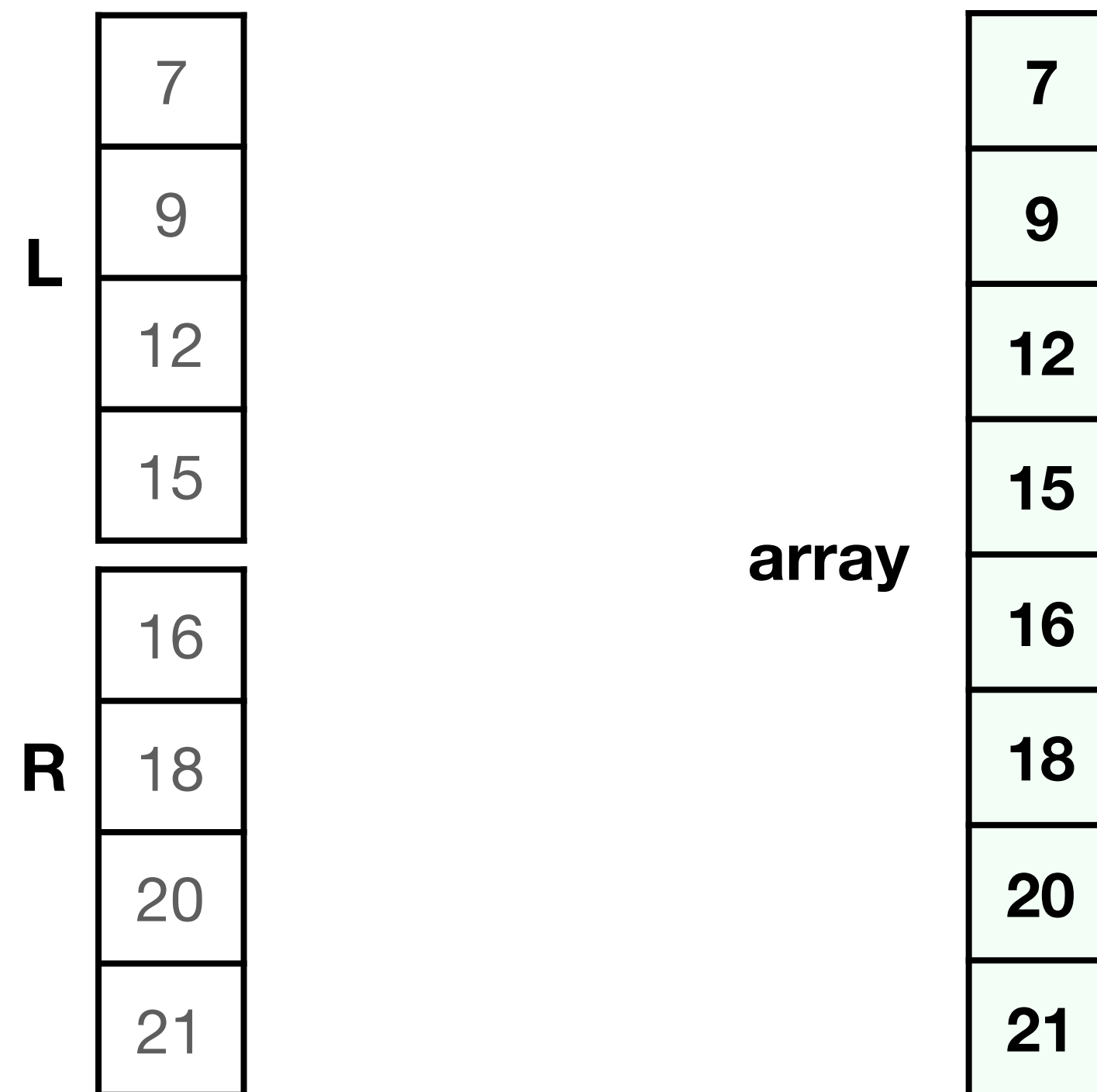
```
array[k] = L[i]
if(L[i] <= R[j]) ? Yes
```


Merge sort best case



$\text{array}[k] = L[i]$

Merge sort best case



$\text{array}[k] = R[j]$