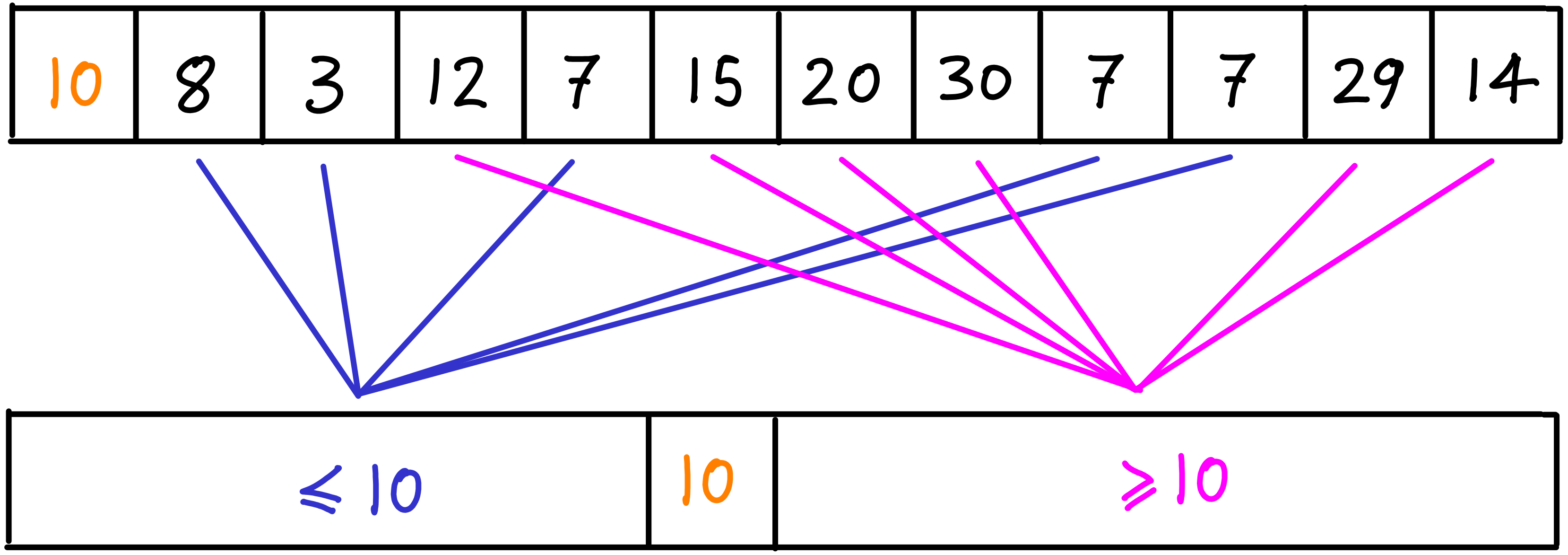
PARTITION

Input: an array (or linked list)

PIVOT - An element in the array.

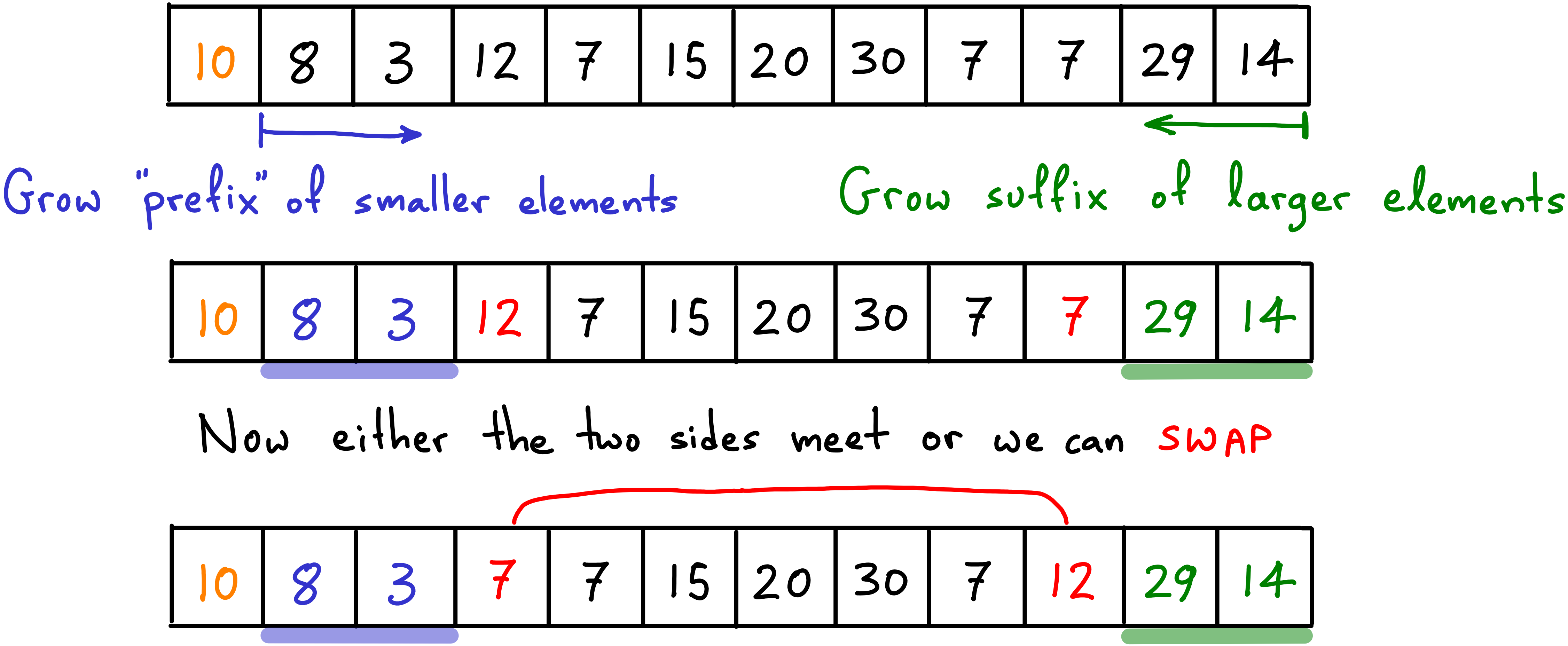
(either given as input, or random, depending on context)

wlog, assume pivot is leftmost



Output:

- Elements smaller than pivot go to its left
- Arbitrary handling of duplicates. (or use some consistent rule, e.g., <10 vs >=10)



10	8	3	12	7	15	20	30	7	7	29	14

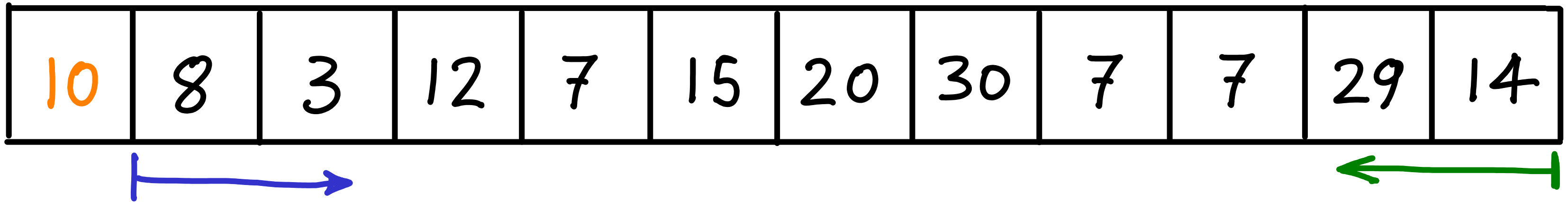
Grow "prefix" of smaller elements

Grow suffix of larger elements

10	8	3	12	7	15	20	30	7	7	29	14

Now either the two sides meet or we can SWAP

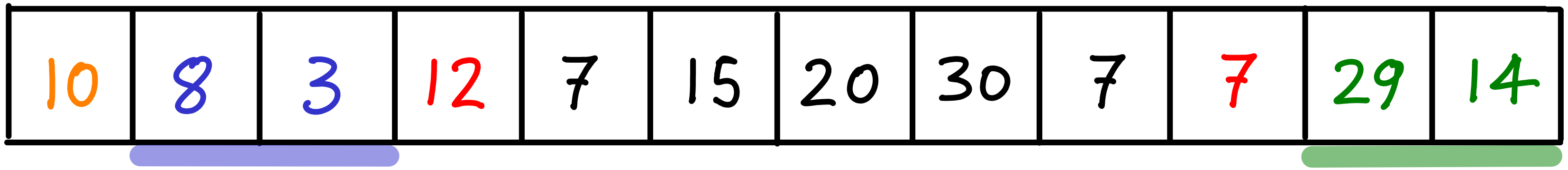
10	8	3	7	7	15	20	30	7	12	29	14



10	8	3	12	7	15	20	30	7	7	29	14

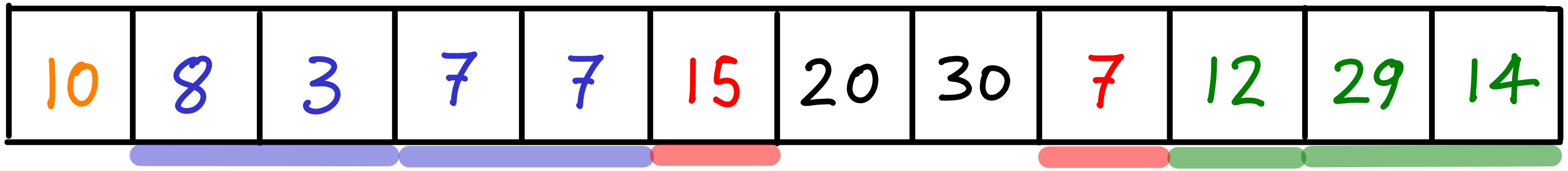
Grow "prefix" of smaller elements

Grow suffix of larger elements



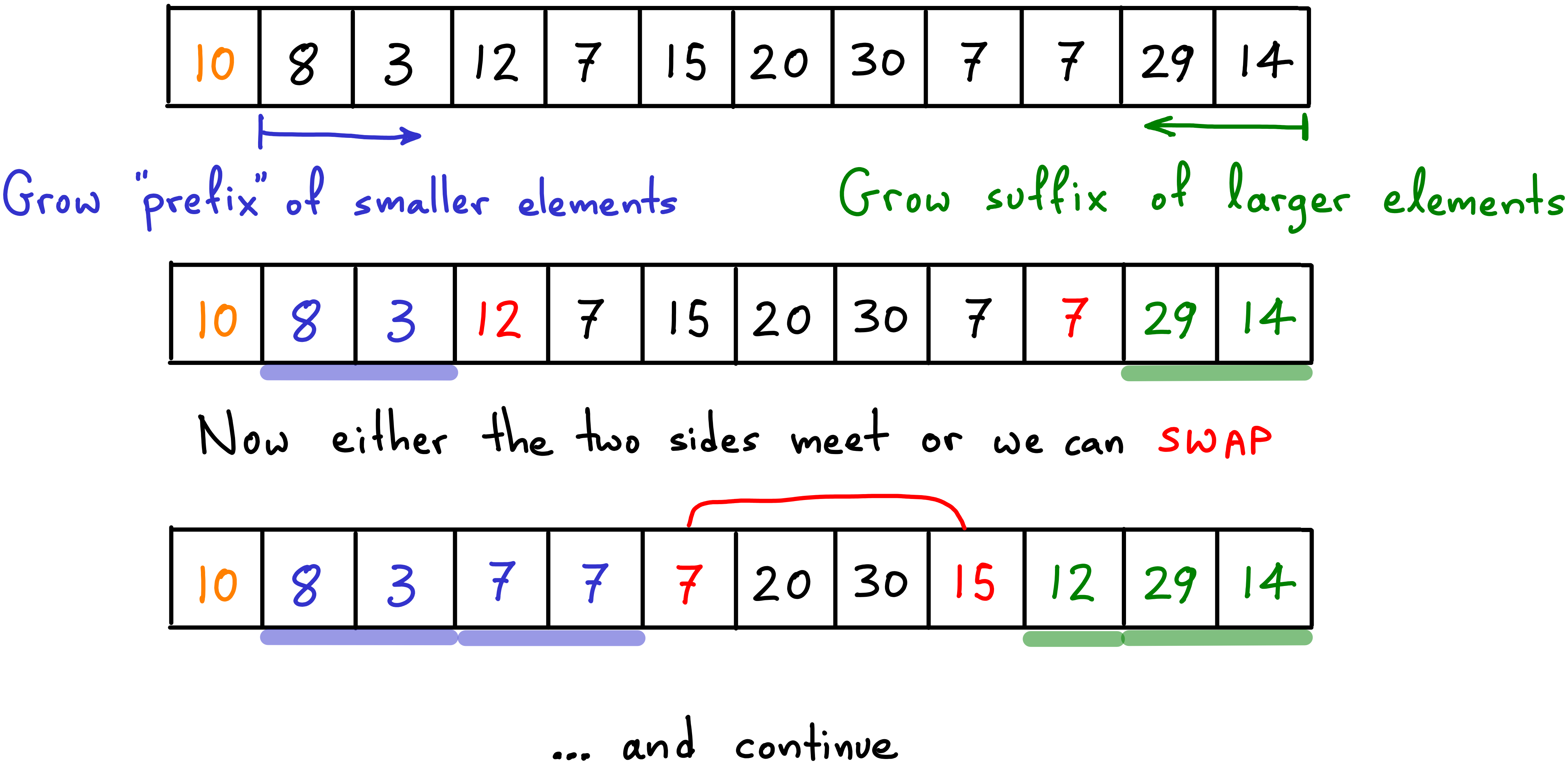
10	8	3	12	7	15	20	30	7	7	29	14

Now either the two sides meet or we can SWAP



10	8	3	7	7	15	20	30	7	12	29	14

... and continue



10	8	3	12	7	15	20	30	7	7	29	14

Grow "prefix" of smaller elements

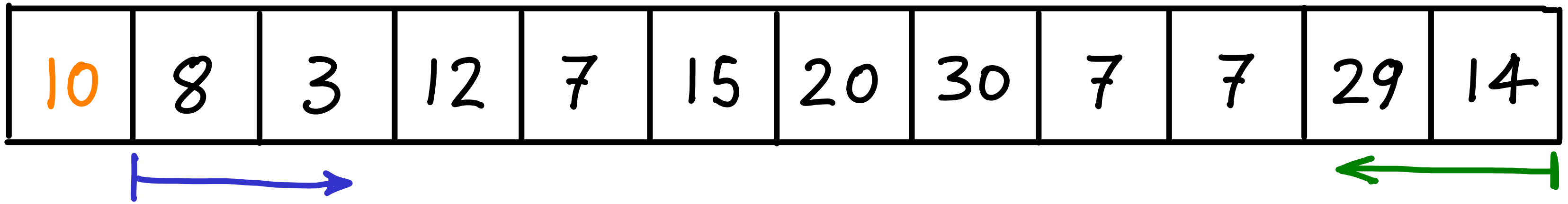
Grow suffix of larger elements

10	8	3	12	7	15	20	30	7	7	29	14

Now either the two sides meet or we can SWAP

10	8	3	7	7	7	20	30	15	12	29	14

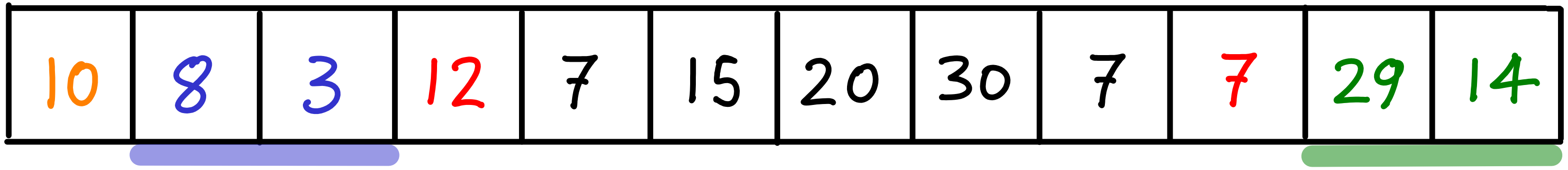
... and continue



10	8	3	12	7	15	20	30	7	7	29	14

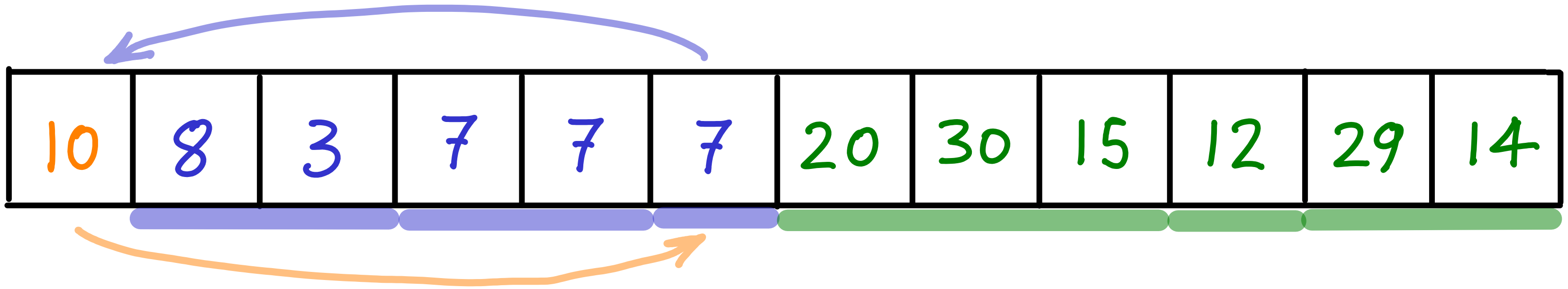
Grow "prefix" of smaller elements

Grow suffix of larger elements



10	8	3	12	7	15	20	30	7	7	29	14

Now either the two sides meet or we can SWAP



10	8	3	7	7	7	20	30	15	12	29	14

Finally place pivot at correct spot

10	8	3	12	7	15	20	30	7	7	29	14

Grow "prefix" of smaller elements

Grow suffix of larger elements

10	8	3	12	7	15	20	30	7	7	29	14

Now either the two sides meet or we can SWAP

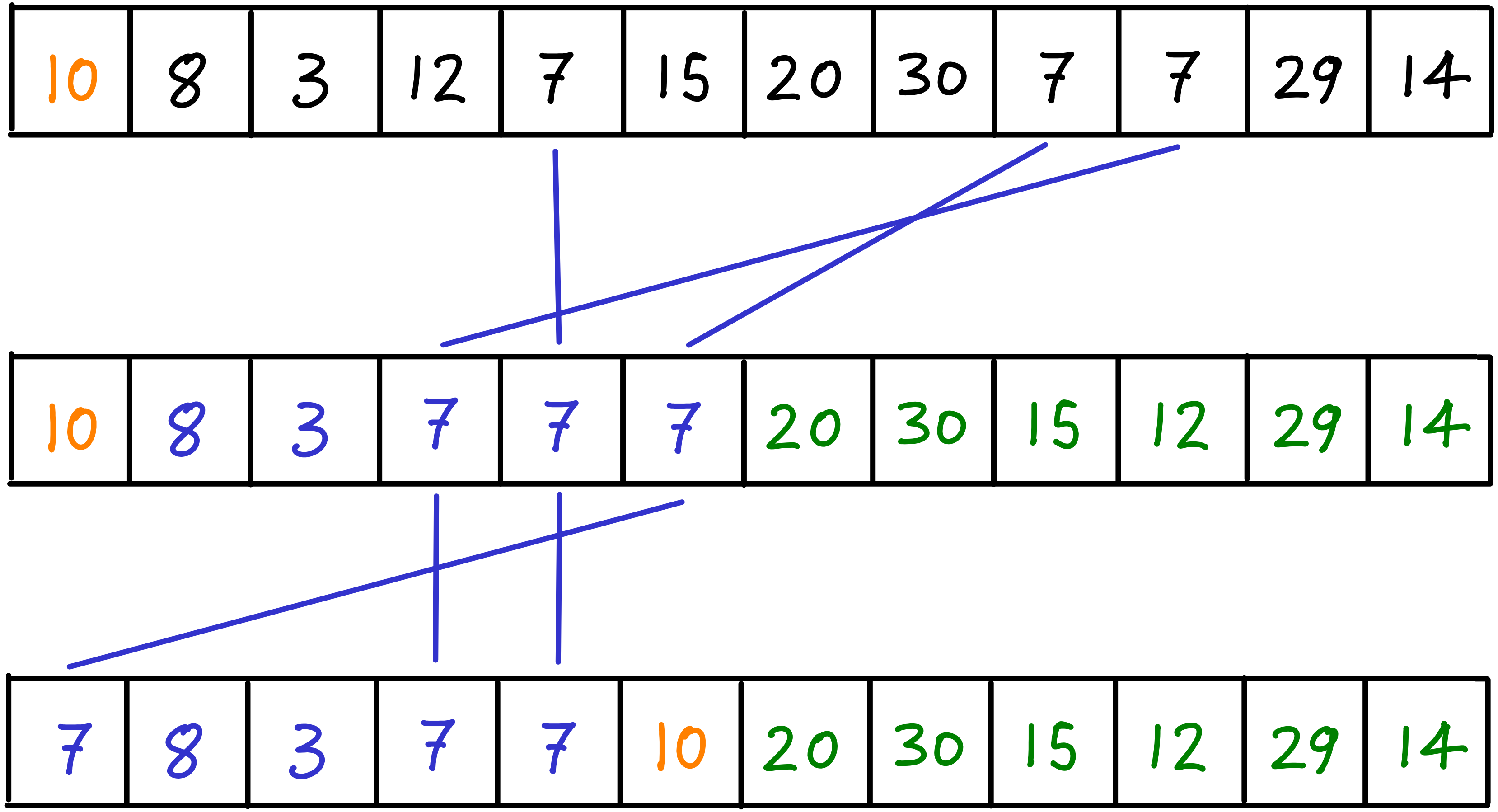
7	8	3	7	7	10	20	30	15	12	29	14

Finally place pivot at correct spot

$\Theta(n)$ time

Partition was done IN-PLACE : only O(1) extra space.

But it was not STABLE : order of elements might change.

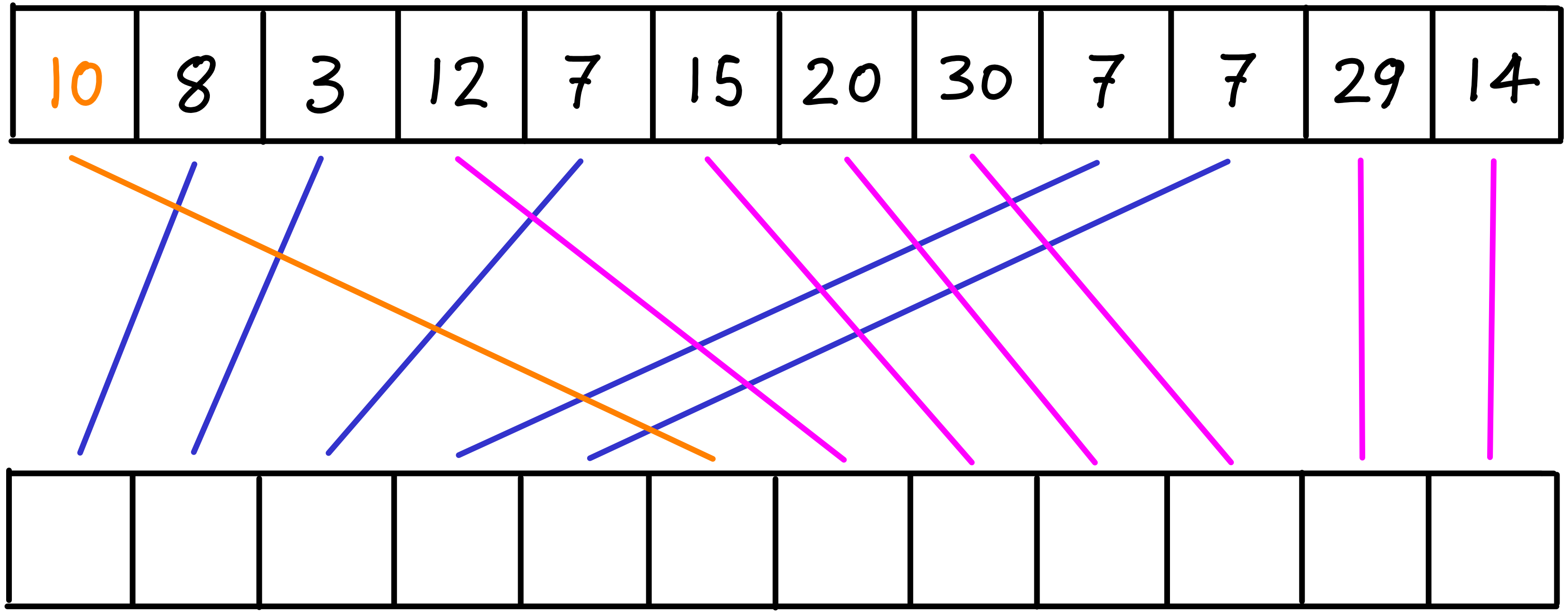


10	8	3	12	7	15	20	30	7	7	29	14

10	8	3	7	7	7	20	30	15	12	29	14

7	8	3	7	7	10	20	30	15	12	29	14

Trivial stable version, using extra space



10	8	3	12	7	15	20	30	7	7	29	14

$\Theta(n)$ output space

Stable AND in-place? → Not $O(n)$ time