Quick Sort Algorithm

```
int arr[] = { 3,1,2,5,8,7,6 };
int s = 0;
int pivot = arr[s];
count = 0;  (2)
( int i = s+1; i <= e; i++ ){
  if (arr[i] <= pivot)
  {
    count ++;
  }
}
swap(arr[pivotindex], arr[s])
```

Recursion

```
int pivotIndex = [s + count]
               = 0 + 2
               = (2)
```

partition index

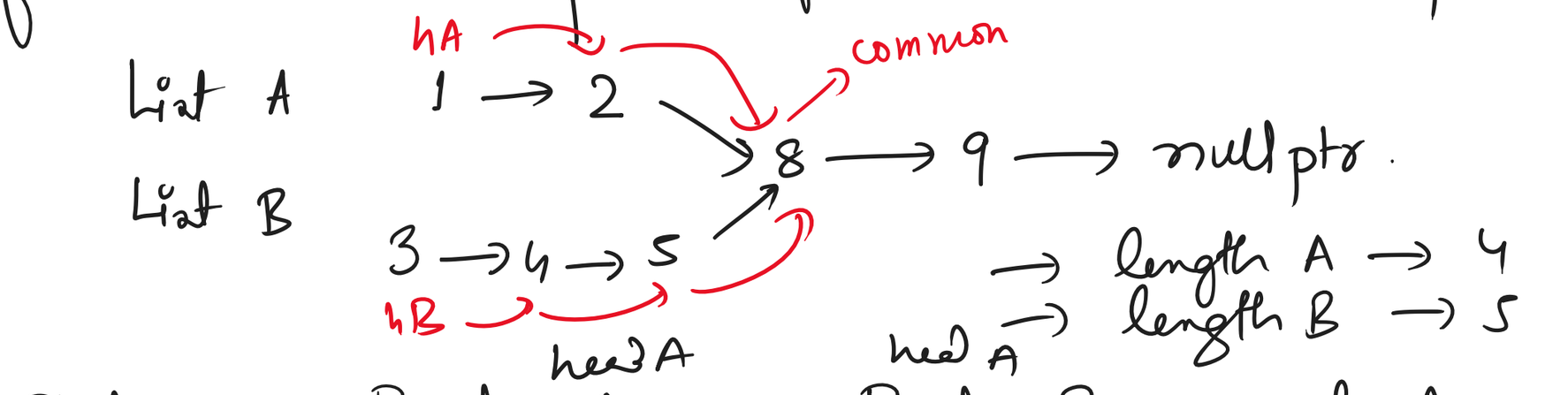
0 2, 1, 3, 5, 8, 7, 6 9

```
while (i < pind && j > pind)
{
  while(arr[i] < pivot) i++;
  } while (arr[i] > pivot) j--;
```

Give me one soln, I will give you the rest.

Interview Questions on Linked Lists :→

Cabgemini :→ Given two linked lists list A & list B find the intersection point. If not, return nullptr.



Steps	Pointer A	Pointer B
1	1	3
2	2	4
3	8	5
4	9	8
5	null	9
6	3	null
7	4	1
8	5	2
9	→ 8	8 ←

** Imp (TCS/Accenture/Math Co)

Floyd's Algorithm for Cycle Detection in SLL

List :→ 1 → 2 → 3 → 4 → 5 (+1 +2)

5 is connected back to 3 forming a cycle.

true or false ;

Steps	Slow	Fast
1	1	1
2	2	3
3	3	5
4	4	3
	(5)	(5)