Recursion - Array Questions

Q) Find if array is sorted or not?

arr = [1, 2, 4, 8, 9, 12]

arr = [1, 2, 4, 3, 8, 9]

arr[i] < arr[i+1] & the remaining array is sorted or not.

arr[i] < arr[i+1] & sorted()

[1, 2, 4, 8, 9, 12]

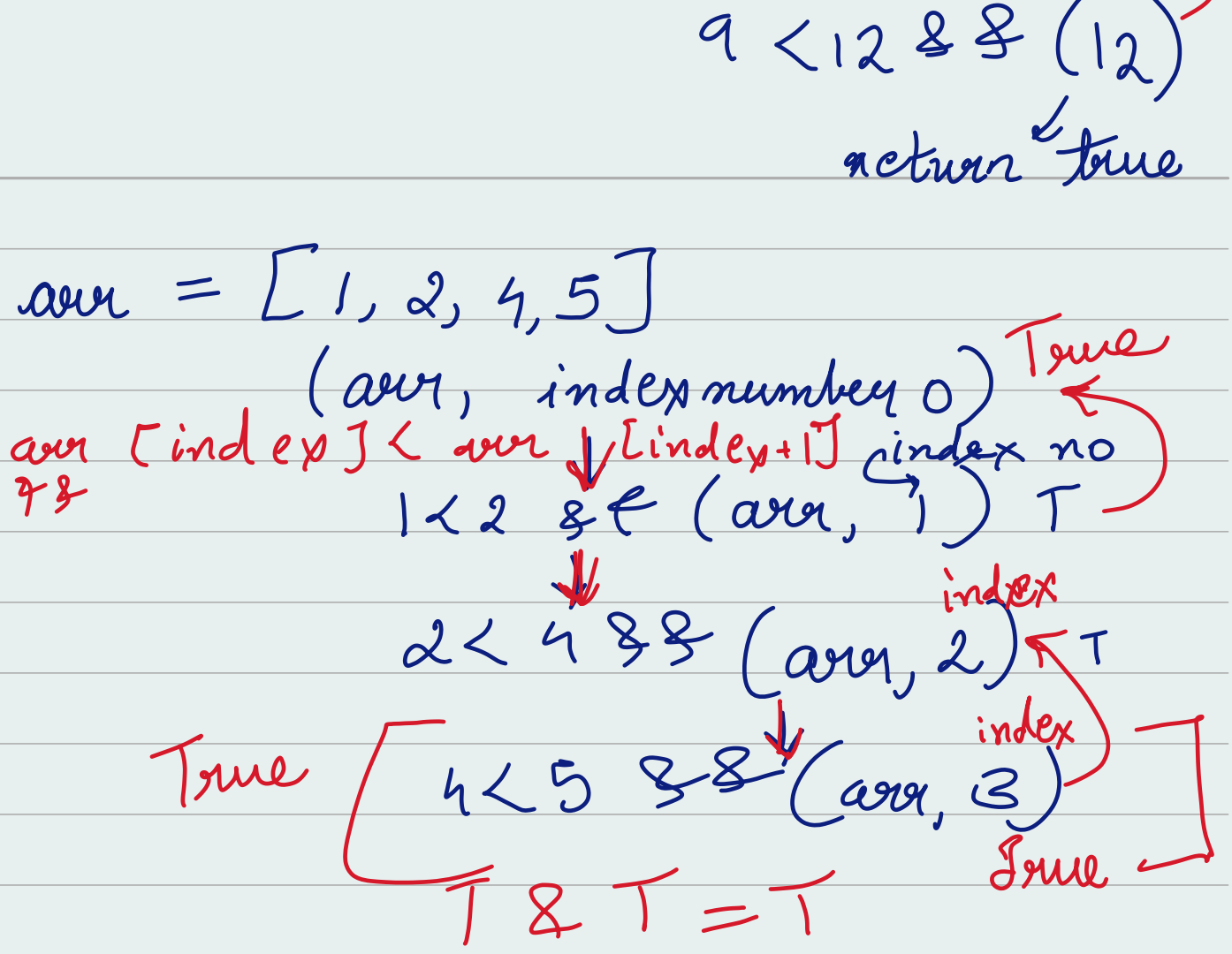
arr[i] < arr[i+1] && (2, 4, 8, 9, 12)

1 < 2 && (2, 4, 8, 9, 12)

2 < 4 && (4, 8, 9, 12)

4 < 8 && (8, 9, 12)

8 < 9 && (9, 12)



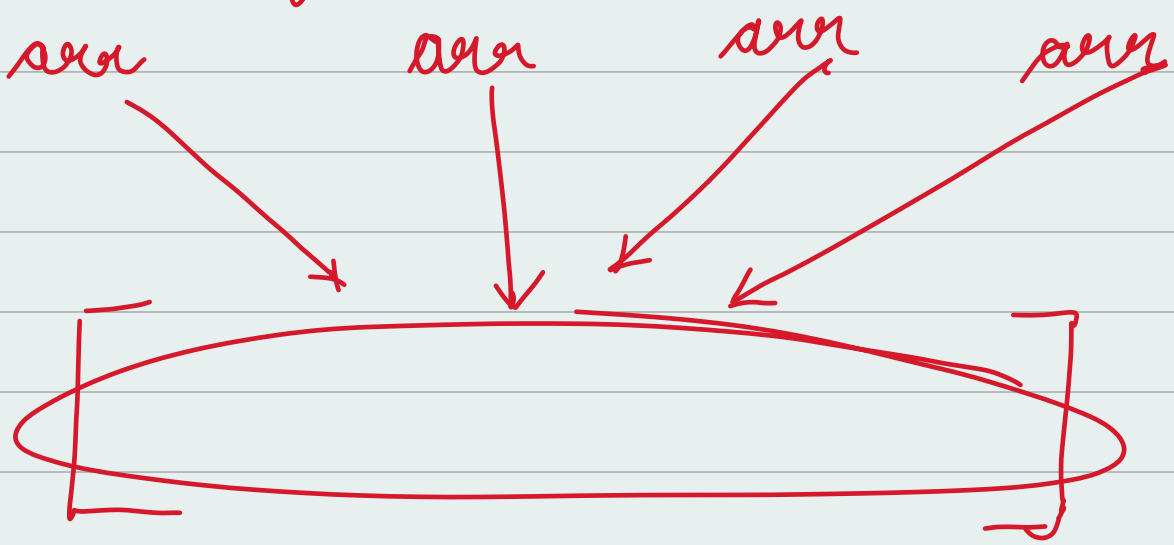

```
psvm( ){
    int[] arr = {1, 2, 4, 5, 10};
    System.out.println(SortedArr(arr, 0));
```

mention the index

```
static boolean SortedArr(int[] arr,
                         int index){
    // base condition
    if (index == arr.length-1){
        return True;
    }
    return arr[index]<arr index+1]
```

&& Sorted Arr (arr, index+1);
}
}

If you modify the object via reference variable such as array all the future recursion calls will also change



8) Linear Search with recursion.

arr = [3, 2, 1, 18, 9]
0 1 2 3 4

target = 18

(arr, target, 0) ← index will change

These two won't change

if arr[0] == target || (arr, target, 1);

check → [or function either one should be true to give result]

arr[1] == target || (arr, target, 2)

arr[2] == target || (arr, target, 3)

arr[3] == target || (arr, target, 4)

code in Mac — till the index become out of bound: false

---

5) Linear Search (on multiple occurances)

arr = [1, 2, 3, 4, 4, 8], target = 4

(0 1 2 3 4 5)

ans = [3, 4]

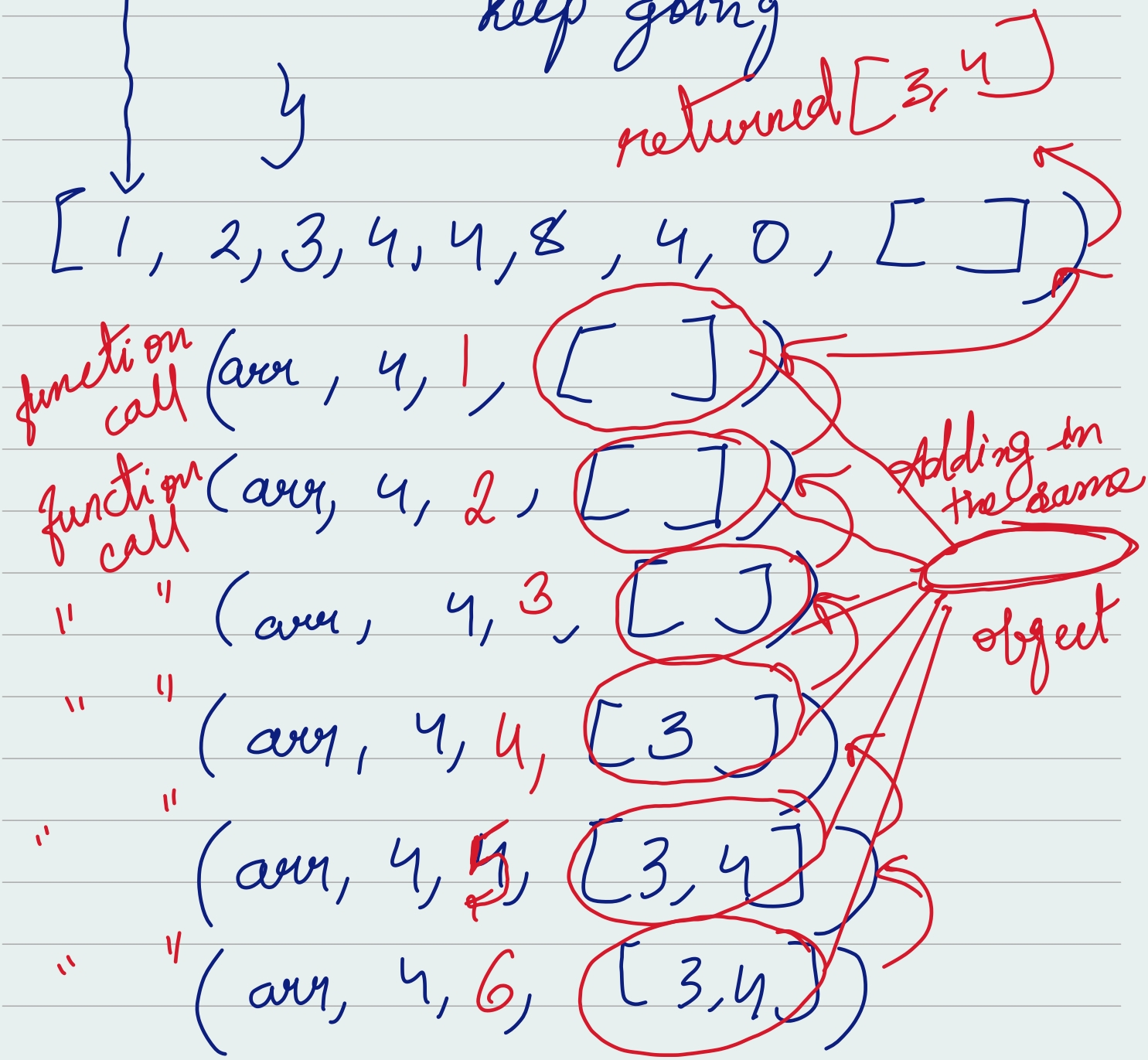
```
function( arr, target, index, list)
    if (index == end){
```

return res;

}

if (target == arr(index)
add in list.
keep going

}



---

8) VVIP: Goal *return the list* dont take it in argument

challenges: We know return type will be arraylist.

problem: Every call will have a new list

arr = [1, 2, 3, 4, 4, 8], target = 4

index = 0

(arr, target, 0)

[list = [ ]] → body of function

↓

(arr, target, 1)

[list = [ ]] → new list are created at every function call.

↓

(arr, target, 2)

[list = [ ]]

↓

(arr, target, 3)

[list = [3]]

↓

(arr, target, 4)

[list = [4]]

Here new list are created for every function call, so the target, 3 and target, 4 won't be

a new list.

even the future and past list don't know about it The ans is only available in the individual function calls.

(arr, target, 5)
[list = [ ]]
(arr, target, 6)
[list = [ ]]
base condition

while the ans is being returned from below function calls, we can check condition like is this function have an answer if it does then return it and move it to upper functions.

---

8) Rotated Binary Search

$\frac{7}{2} = 3$

arr = [5, 6, 7, 8, 9, 1, 2, 3]
start, mid, end

① if arr[start] ≤ arr[mid]

suppose 7 is target → if key ≥ arr[start] && ≤ arr[mid]

end = mid - 1

else

start = mid + 1

suppose ② 9 is the target

② if key > arr[mid] && key <= arr[end]

start = mid + 1;

arr = 5, 6, 1, 2, 3, 4 (s, m, e)

5/2 ≅ 2

suppose target = 6

③ else end = mid - 1

// Base condition.

if (start > end)

return -1;