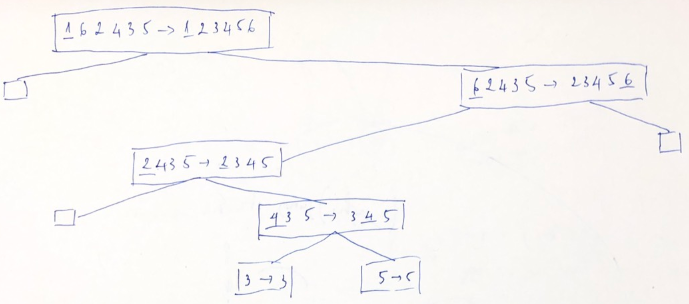
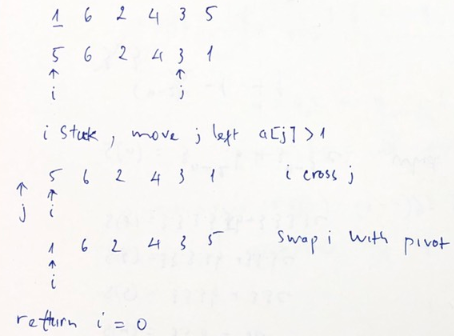
# Prob1:



# Prob2:



# Prob3:

A = [5, 1, 4, 3, 6, 2, 7, 1, 3] (here, n = 9)

3n/4 = 6.75

[1, 1, 2, 3, 3, 4, 5, 6, 7]

a. good pivots: 2, 3, 3, 4, 5 (total: 5>9/2)

b. It’s **TRUE** that at least half the elements of A are good pivots

# Prob4:

static int find(int[] a, int from, int to) {

if (from > to)

return -1;// not found

int mid = (from + to) / 2;

int v = a[mid];

if (v == mid)

return mid;

if (mid < v)

return find(a, from, mid - 1);

return find(a, mid + 1, to);

}

# Prob5:

static HashSet<Integer> subset(int[] a, int to, int k) {

if (k == 0)

return new HashSet<Integer>();

if (to == 0) {

if (a[to] == k)

return new HashSet<Integer>() {

{

add(k);

}

};

return null;

}

HashSet<Integer> t1 = subset(a, to - 1, k);

if (t1 != null)

return t1;

HashSet<Integer> t2 = subset(a, to - 1, k - a[to]);

if (t2 != null) {

t2.add(a[to]);

return t2;

}

return null;

}