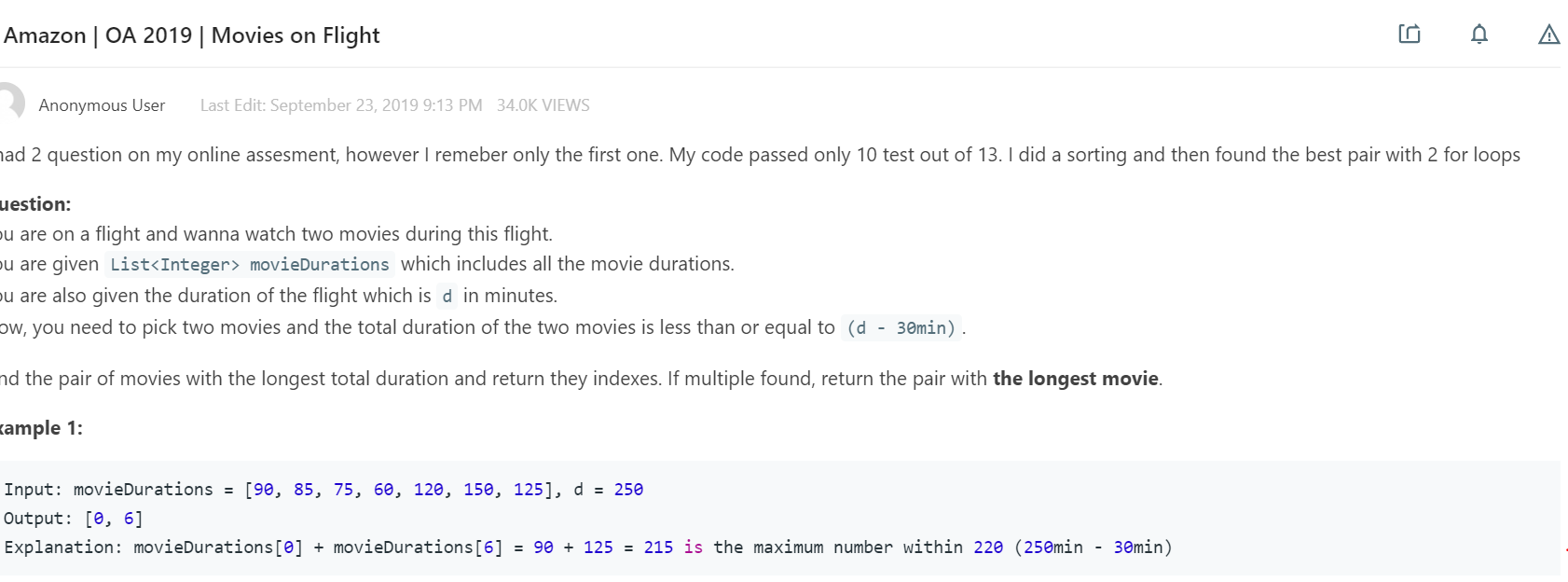
LEETCODE : <https://leetcode.com/discuss/interview-question/313719/Amazon-or-Online-Assessment-2019-or-Movies-on-Flight>

https://leetcode.com/discuss/interview-question/313719/Amazon-or-Online-Assessment-2019-or-Movies-on-Flight



public static void main(String[] args) {

int[] movie\_duration1 = {90, 85, 75, 60, 120, 150, 125};

int d1 = 250;

int[] movie\_duration2 = {90, 85, 75, 60, 155, 150, 125};

int d2 = 250;

int[] movie\_duration3 = {90, 85, 75, 60, 120,110,110, 150, 125};

int d3 = 250;

System.out.println(Arrays.toString(get2SumClosest(movie\_duration1, d1-30)));

System.out.println(Arrays.toString(get2SumClosest(movie\_duration2, d2-30)));

System.out.println(Arrays.toString(get2SumClosest(movie\_duration3, d3-30)));

}

private static int[] get2SumClosest(int[] movie\_duration, int d) {

Map<Integer, List<Integer>> map = new HashMap<>();

for(int i=0;i<movie\_duration.length;i++) {

map.putIfAbsent(movie\_duration[i], new ArrayList<>());

map.get(movie\_duration[i]).add(i);

}

Arrays.sort(movie\_duration);

int l = 0, r = movie\_duration.length - 1;

int max = 0;

int[] res = new int[]{-1, -1};

while(l < r) {

int sum = movie\_duration[l] + movie\_duration[r];

if((sum > max || (sum == max && Math.max(movie\_duration[l] , movie\_duration[r]) > Math.max(res[0], res[1]))) && sum <= d) {

max = sum;

res[0] = movie\_duration[l];

res[1] = movie\_duration[r];

}

if(sum > d)

r--;

else

l++;

}

if(map.get(res[0]) == map.get(res[1])) {

res[0] = map.get(res[0]).get(0);

res[1] = map.get(res[1]).get(1);

}else {

res[0] = map.get(res[0]).get(0);

res[1] = map.get(res[1]).get(0);

}

return res;

}