import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

public class CombinationFinder {

public static List<List<Integer>> findCombinations(int[] arr, int target) {

List<List<Integer>> combinations = new ArrayList<>();

// Step 1: Find pairs with sum equal to target

for (int i = 0; i < arr.length; i++) {

for (int j = i + 1; j < arr.length; j++) {

if (arr[i] + arr[j] == target) {

combinations.add(Arrays.asList(arr[i], arr[j]));

}

}

}

// Step 2: Merge and sort the array

Arrays.sort(arr);

// Step 3: Double the target value

int doubledTarget = 2 \* target;

// Step 4: Find combinations equal to doubled target

List<List<Integer>> secondCombinations = new ArrayList<>();

for (int i = 0; i < arr.length; i++) {

for (int j = i + 1; j < arr.length; j++) {

int sum = arr[i] + arr[j];

if (sum == doubledTarget) {

List<Integer> combination = new ArrayList<>();

for (int k = i; k <= j; k++) {

combination.add(arr[k]);

}

secondCombinations.add(combination);

}

}

}

return secondCombinations;

}

public static void main(String[] args) {

int[] arr = {1, 3, 2, 2, -4, -6, -2, 8};

int target = 4;

List<List<Integer>> result = findCombinations(arr, target);

System.out.println("First Combination For " + target + " : " + result);

Arrays.sort(arr);

System.out.println("Merge Into a single Array : " + Arrays.toString(arr));

int doubledTarget = 8;

List<List<Integer>> secondResult = findCombinations(arr, doubledTarget);

System.out.println("Second Combination For " + doubledTarget + " : " + secondResult);

}

}

output

First Combination For 4 : [[-4, 8], [-2, 6], [1, 3], [2, 2]]

Merge Into a single Array : [-6, -4, -2, 1, 2, 2, 3, 8]

Second Combination For 8 : [[1, 3, 2, 2], [2, 2, -4, 8], [2, 2, 8, -4], [3, 1, 2, 2], [8, -4, 2, 2]]