

## Homework 12, CSE 232

**Due December 3** Note: On most of the problem sets through the semester, I'll put a horizontal line with "Optional" under it. Any problems below this section are encouraged - I think they're interesting and will help you learn the subject - but not necessary to complete in order to get credit for the homework.

### Problem 1

This week's problem is [Google Code Jam Round 1B 2012 - Equal Sums](#). Here's the problem statement:

I have a set of positive integers  $S$ . Can you find two non-empty, distinct subsets with the same sum?

Note: A subset is a set that contains only elements from  $S$ , and two subsets are distinct if they do not have exactly the same elements.

### Input

The first line of the input gives the number of test cases,  $T$ .  $T$  test cases follow, one per line. Each test case begins with  $N$ , the number of positive integers in  $S$ . It is followed by  $N$  distinct positive integers, all on the same line.

### Output

For each test case, first output one line containing "Case #x:", where  $x$  is the case number (starting from 1).

If there are two different subsets of  $S$  that have the same sum, then output these subsets, one per line. Each line should contain the numbers in one subset, separated by spaces. If it is impossible, then you should output the string "Impossible" on a single line. If there are multiple ways of choosing two subsets with the same sum, any choice is acceptable.

### Limits

No two numbers in  $S$  will be equal.

$1 \leq T \leq 10$ .

### Small dataset

$N$  is exactly equal to 20.

Each number in  $S$  will be a positive integer less than  $10^5$ .

### Large dataset

$N$  is exactly equal to 500.

Each number in  $S$  will be a positive integer less than  $10^{12}$ .

*a) For the Small input, what is the smallest and largest possible sum of a subset of  $S$ ? How about for the Large input?*

*b) Describe how you could write a dynamic programming solution to the Small input by relating larger subset sums to smaller ones. Why won't this approach work for the Large subset?*

*c) What are the total possible number of subsets for the Large inputs? How about only subsets of size 6? How does this suggest a simple randomized approach to solving the problem? Implement a randomized algorithm to the Large input, test it, and attach it to your homework.*