



Mock Interviews @ ACM

Problem Card

Room Draw Woes: The Sequel

Problem Statement

Housing is struggling with another problem — can you help them out? **Note:** We recommend that you first do *Room Draw Woes* and then solve this problem as a follow-up.

You have a draw group with N people and Housing has a specific set of room sizes `rooms` they can offer. Write a function `roomDrawTwo(N, rooms)` that outputs the *total number of ways* Housing can assign some set of rooms so that everyone is accommodated *and* no room is under capacity. (Assume there is an infinite supply of each room.)

Assumptions. You can assume N and the elements of `rooms` (i.e. possible room sizes) will fit in a 32-bit positive integer. Assume the total number of ways can also fit in a 32-bit integer.

Example. Using the same example from Room Draw Woes 1 of $N = 7$ and `rooms = {1, 2, 4}`, the correct answer this time would be 5, because the you can make 7 in the following ways:

```
4 + 2 + 1 = 7
2 + 2 + 2 + 1 = 7
2 + 2 + 1 + 1 + 1 = 7
2 + 1 + 1 + 1 + 1 + 1 = 7
1 + 1 + 1 + 1 + 1 + 1 + 1 = 7
```

As usual, you should analyze the runtime and space usage of any solution you come up with.

Test Cases

```
// Input 1
int N = 7; int[] rooms = {1, 2, 4};
System.out.println(roomDrawTwo(N, rooms));
```

```
// Output 1
5

// Explanation
// See the example above.

// Input 2
int N = 3; int[] rooms = {2, 4};
System.out.println(roomDrawTwo(N, rooms));

// Output 2
0

// Explanation
// There's no way to house exactly 3 people in only doubles and quads.

// Input 3
int N = 9; int[] rooms = {3, 5};
System.out.println(roomDrawTwo(N, rooms));

// Output 3
1

// Explanation
// The only way to accommodate the draw group exactly is with three triples.
```