1. Given an integer array(a_1a_2…..a_n) of size n, you've to report the length of the longest contiguous increasing sequence.

   You must solve it in O(n) complexity.

   Sample Input   Sample Output

   7
   3 1 5 2 7 9 3   3

   Explanation: Here 2,7,9 is the desired contiguous increasing sequence.

   6
   4 -3 -1 2 5 -2   4

2. Given an integer array(a_1a_2…..a_n) of size n containing non-negative integers. If you are currently at position i, at the next step you can go to one of the positions i+1 , i+2…………i+a_i (You are not allowed to move beyond n). Initially you're located at position 1, you've to print the minimum number of moves required to reach position n if it is possible to reach, otherwise print -1

   You must solve it in O(n²) or better complexity.

   Sample Input   Sample Output
   7
   3 0 5 2 0 0 0   2

   Explanation: Initially you're at position 1. In one move, you can reach either 2,3,4. If you move to position 3, then using another move you can reach to position 4,5,6,7. So in 2 moves you can reach position 7.

   7
   3 0 3 2 0 0 0   -1

   9
   4 3 0 2 1 3 1 1 1   3

3. You are given n coins(c_1c_2…..c_n) and a number w. You have to answer how many numbers from 1 to w can be made using these coins. You can use any coin multiple times.
   You must solve it in O(n*w) complexity.

Sample Input

3 20
4 6 9

Sample Output

4 6 8 9 10 12 13 14 15 16 17 18 19 20

Explanation:
4 -> 4
6 -> 6
8 -> 4+4
9 -> 9
10 -> 4+6
12 -> 6+6
13 -> 4+9
14 -> 4+6+6
15 -> 6+9
16 -> 4+6+6
17 -> 4+4+9
18-> 6+6+6
19 -> 4+6+9
20 -> 4+4+6+6

4. There are n containers of food containing c_1, c_2 .....c_n kg food in front of you. You want to eat some food but you can't eat more than W kg. From each container you'll either eat all the food or none. You have to report the maximum amount of food you can eat and which containers you'll use.

You need to solve this problem in O(n*W) complexity.

Sample Input

5 15
6 3 5 8 1

Sample Output

15
1 4 5

4 12
2 8 9 6

11
1 3

4 15
4 5 4 12

13
1 2 3