

# MCIPC Contest 2 P6 - Block Tower

---

**Time limit:** 1.0s    **Memory limit:** 256M

---

A freak snowstorm has left you and your friends stranded at school without power, service, or even a way out! To overcome the inevitable boredom that ensues, your friends have devised a game, the rules are as follows:

- You and your opponent have a collection of  $N$  blocks, where the  $i^{\text{th}}$  block has a height of  $H_i$ .
- The two of you will then take turns stacking blocks. On each turn, a player will take any block that has not been chosen and place it on the top of the tower. Placing the  $i^{\text{th}}$  block will add its  $H_i$  height to the total height of the tower. Consider the initial height of the tower to be 0.
- The first person to place a block that will cause the tower to fall is the loser of the game, while the other player is the winner.

As your friends are very nice, they insist that you must go first.

Additionally, due to you and your friend's intricate understanding of physics, you both know that if the tower exceeds a height of  $K$ , it will collapse.

Because of this, you and your friend will play optimally, and neither of you will ever make a mistake. Therefore, you have reasoned the only way to win is to never play a game which is impossible to win in the first place. Since you are a **nerd** competitive programming enthusiast, you get out your laptop to use your skills to figure out if you can win.

## Constraints

---

$$1 \leq N \leq 20$$

$$1 \leq K, H_i \leq 10^9$$

The sum of the heights of all blocks will exceed  $K$ .

### Subtask 1 [10%]

The height of all blocks will be 1.

### Subtask 2 [90%]

No additional constraints.

## Input Specification

---

The first line contains the integers  $N$  and  $K$  separated by a space.

The second line contains  $N$  space-separated integers, where the  $i^{\text{th}}$  integer represents the height of the  $i^{\text{th}}$  block.

## Output Specification

---

Output either `win` if you can win the game by playing optimally, otherwise output `lose`.

## Sample Input 1

---

```
3 10
6 5 4
```

## Sample Output 1

---

```
lose
```

## Explanation for Sample 1

---

No matter which block you choose, your opponent can always choose a block which does not collapse the tower. You will then be forced to play the final block which will collapse the tower.

## Sample Input 2

---

```
5 6
1 2 3 4 5
```

## Sample Output 2

---

```
win
```

## Explanation for Sample 2

---

By playing the block with height 3 first, the other player is forced to play either the block of height 1 or the block of height 2.

If they choose the block with height 1, you can choose the block with height 2 to force a win.

If they choose the block with height 2, you can choose the block with height 1 to force a win.