

题目描述：

小明在玩一个游戏，游戏规则如下：

在游戏开始前，小明站在坐标轴原点处（坐标值为 0 ）。

给定一组指令和一个幸运数，每个指令都是一个整数，小明按照指定的要求前进或者后退指定的步数。前进代表朝坐标轴的正方向走，后退代表朝坐标轴的负方向走。

幸运数为一个整数，如果某个指令正好和幸运数相等，则小明行进步数加 1 。

例如：

幸运数为 3 ，指令为 $[2, 3, 0, -5]$

指令为 2 ，表示前进 2 步；

指令为 3 ，正好和幸运数相等，前进 $3+1=4$ 步；

指令为 0 ，表示原地不动，既不前进，也不后退。

指令为 -5 ，表示后退 5 步；

请你计算小明在整个游戏过程中，小明所处的最大坐标值。

输入描述：

第一行输入 1 个数字，代表指令的总个数 n ($1 \leq n \leq 100$)。

第二行输入 1 个数字，代表幸运数 m ($-100 \leq m \leq 100$)

第三行输入 n 个指令，每个指令值的取值范围为： $-100 \leq \text{指令值} \leq 100$ 。

输出描述：

输出在整个游戏过程中，小明所处的最大坐标值。异常情况下输出： 12345

示例 1

输入：

1

-5 1

输出：

0

说明：

总共 2 个指令，幸运数为 1，依照指令行进，依次如下：

游戏开始前，站在坐标轴原点，此时坐标值为 0；

指令为 -5，后退 5 步，此时坐标值为 -5；

指令为 1，正好等于幸运数，前进 $1+1=2$ 步，此时坐标值为 -3；

整个游戏过程中，小明所处的坐标值依次为 $[0, -5, -3]$ ，最大坐标值为 0。

示例 2

输入：

5

-5

-5 1 6 0 -7

输出：

1

说明：

总共 5 个指令，幸运数为 -5，依照指令行进，依次如下：

游戏开始前，站在坐标轴原点，此时坐标值为 0；

指令为 -5，正好等于幸运数，后退 $5+1=6$ 步，此时坐标值为 -6；

指令为 1，前进 1 步，此时坐标值为 -5；

指令为 6，前进 6 步，此时坐标值为 1；

指令为 0，既不前进，也不后退，此时坐标值为 1；

指令为-7，后退 7 步，此时坐标值为-6；

整个游戏过程中，小明所处的坐标值依次为[0,-6,-5,1,1,-6]，最大坐标值为 1。

```
try:
    count = int(input())

    lucky = int(input())

    num_list = list(map(int, input().split()))
    if len(num_list) != count:
        raise Exception

    max_num = 0
    current = 0
    for num in num_list:
        if num == lucky:
            num += 1
        current += num
        if current > max_num:
            max_num = current

    print(max_num)
except Exception:
    print('12345')
```