

题目描述：

小明和朋友玩跳格子游戏，有 n 个连续格子，每个格子有不同的分数，小朋友可以选择从任意格子起跳，但是不能跳连续的格子，也不能回头跳；

给定一个代表每个格子得分的非负整数数组，计算能够得到的最高分数。

输入描述：

给定一个数例，如：

1 2 3 1

输出描述：

输出能够得到的最高分，如：

4

补充说明：

$1 \leq \text{nums.length} \leq 100$

$0 \leq \text{nums}[i] \leq 1000$

示例 1

输入：

1 2 3 1

输出：

4

说明：

选择跳第一个格子和第三个格子

示例 2

输入：

2 7 9 3 1

输出:

12

说明:

$2+9+1=12$

```
const rl = require("readline").createInterface({ input: process.stdin });
var iter = rl[Symbol.asyncIterator]();
const readline = async () => (await iter.next()).value;
```

```
void async function () {
    // Write your code here
    while(line = await readline()){
        let tokens = line.split(' ').map(Number);
        const len = tokens.length
        if (!len) {
            console.log(0)
        } else {
            const arr = new Array(len)
            let result = -Infinity
            for (i = len - 1; i >=0; i--) {
                let item
                if (i === len - 1) {
                    item = tokens[i]
                } else if (i === len - 2) {
                    item = Math.max(tokens[i], tokens[i + 1])
                } else {
                    item = Math.max(tokens[i] + arr[i + 2], arr[i + 1])
                }
                arr[i] = item
                if (item > result) {
                    result = item
                }
            }
            console.log(result)
        }
    }
}()
```