

## 阿里巴巴找黄金宝箱(V)

### 题目描述：

一贫如洗的樵夫阿里巴巴在去砍柴的路上，无意中发现了强盗集团的藏宝地，藏宝地有编号从  $0 \sim N$  的箱子，每个箱子上面贴有一个数字。

阿里巴巴念出一个咒语数字  $k(k < N)$ ，找出连续  $k$  个宝箱数字和的最大值，并输出该最大值。

### 输入描述：

第一行输入一个数字字符串，数字之间使用逗号分隔，例如：2,10,-3,-8,40,5

字符串中数字的个数  $\geq 1$ ， $\leq 100000$ ；每个数字  $\geq -10000$ ， $\leq 10000$ ；

第二行输入咒语数字，例如：4，咒语数字大小小于宝箱的个数

### 输出描述：

连续  $k$  个宝箱数字和的最大值，例如：39

### 示例 1

#### 输入：

2,10,-3,-8,40,5

4

#### 输出：

39

### 说明：

### 示例 2

#### 输入：

8

1

输出:

8

说明:

```
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(' ');
```

```
        //     let a = parseInt(tokens[0]);
```

```
        //     let b = parseInt(tokens[1]);
```

```
        //     console.log(a + b);
```

```
    // }
```

```
const nums = (await readline()).split(',');
```

```
const win = parseInt(await readline())
```

```
let maxSum = 0;
```

```
for(let i = 0; i<= nums.length-win;i++){
```

```
    let subNums = nums.slice(i, i+win)
```

```
    let sum = 0;
```

```
    for(let num of subNums){
```

```
        sum += parseInt(num)
```

```
}
```

```
if(sum > maxSum) maxSum = sum;
```

```
}
```

```
console.log(maxSum)
```

```
}()
```