

## 2560 - House Robber IV

---

There are **several consecutive houses** along a street, each of which **has some money inside**.

Có một **dãy các ngôi nhà liên tiếp** nằm dọc theo một con phố, mỗi nhà đều **chứa một số tiền nhất định**.

There is also a robber, who wants to **steal money** from the homes, but he **refuses to steal from adjacent homes**.

Một tên trộm muốn lấy **trộm tiền** từ các ngôi nhà này, nhưng hắn **từ chối trộm hai ngôi nhà liền kề nhau**.

The **capability** of the robber is **the maximum amount of money he steals from one house of all the houses he robbed**.

"**Năng lực**" (*capability*) của tên trộm được định nghĩa là **số tiền lớn nhất mà hắn trộm được trong một ngôi nhà bất kỳ trong tất cả các ngôi nhà đã trộm**.

You are given an **integer array nums** representing how much **money is stashed in each house**. More formally, the *i*th house from the left has `nums[i]` dollars.

Bạn được cung cấp một **mảng số nguyên nums**, trong đó `nums[i]` là **số tiền chứa trong ngôi nhà thứ *i* tính từ bên trái**.

You are also given an **integer k**, representing the **minimum number of houses the robber will steal** from. It is always possible to steal at least *k* houses.

Bạn cũng được cung cấp một **số nguyên k**, biểu thị **số lượng nhà tối thiểu mà tên trộm phải trộm**. Luôn đảm bảo rằng luôn có thể chọn ít nhất *k* nhà không kề nhau để trộm.

Return the **minimum capability** of the robber out of all the possible ways to steal at least *k* houses.

Trả về **năng lực nhỏ nhất (minimum capability)** của tên trộm **trong tất cả các phương án trộm được ít nhất *k* nhà** (không trộm hai nhà kề nhau).

### Example 1

- Input: `nums = [2,3,5,9]`, `k = 2`
- Output: 5
- Explanation:

There are three ways to rob at least 2 houses:

- Rob the houses at indices 0 and 2. Capability is  $\max(\text{nums}[0], \text{nums}[2]) = 5$ .
- Rob the houses at indices 0 and 3. Capability is  $\max(\text{nums}[0], \text{nums}[3]) = 9$ .
- Rob the houses at indices 1 and 3. Capability is  $\max(\text{nums}[1], \text{nums}[3]) = 9$ .
- Therefore, we return  $\min(5, 9, 9) = 5$ .

### Example 2

- Input: `nums = [2,7,9,3,1]`, `k = 2`
- Output: 2
- Explanation: There are 7 ways to rob the houses. The way which leads to minimum capability is to rob the house at index 0 and 4. Return  $\max(\text{nums}[0], \text{nums}[4]) = 2$ .

## Constraints

- $1 \leq \text{nums.length} \leq 10^5$
- $1 \leq \text{nums}[i] \leq 10^9$
- $1 \leq k \leq (\text{nums.length} + 1)/2$

## Solutions

Solution 1: Binary Search + Greedy

### Solution 1: Binary Search + Greedy

The problem is asking for the minimum stealing ability of the thief.

*Bài toán yêu cầu tìm năng lực trộm tối thiểu của tên trộm.*

We can use binary search to enumerate the stealing ability of the thief.

*Chúng ta có thể sử dụng thuật toán tìm kiếm nhị phân để liệt kê các mức năng lực của tên trộm.*

For the enumerated ability  $x$ , we can use a greedy approach to determine whether the thief can steal at least  $k$  houses.

*Với mỗi giá trị  $x$  được duyệt qua, ta có thể dùng thuật toán tham lam để kiểm tra xem tên trộm có thể trộm được ít nhất  $k$  ngôi nhà hay không.*

Specifically, we traverse the array from left to right.

*Cụ thể, ta duyệt mảng từ trái sang phải.*

For the current house  $i$  we are traversing, if  $\text{nums}[i] \leq x$  and the difference between the index of  $i$  and the last stolen house is greater than 1, then the thief can steal house  $i$ .

*Đối với nhà thứ  $i$  đang xét, nếu  $\text{nums}[i] \leq x$  và vị trí của nhà  $i$  cách nhà trộm trước đó ít nhất 2 vị trí, thì tên trộm có thể trộm nhà này.*

Otherwise, the thief cannot steal house  $i$ .

*Ngược lại, tên trộm không thể trộm nhà  $i$ .*

We accumulate the number of stolen houses.

*Ta đếm tổng số ngôi nhà đã trộm được.*

If the number of stolen houses is greater than or equal to  $k$ , it means that the thief can steal at least  $k$  houses, and at this time, the stealing ability  $x$  of the thief might be the minimum.

*Nếu tổng số nhà trộm được  $\geq k$ , thì điều đó nghĩa là tên trộm có thể trộm đủ yêu cầu, và  $x$  có thể là năng lực tối thiểu phù hợp.*

Otherwise, the stealing ability  $x$  of the thief is not the minimum.

*Ngược lại, năng lực  $x$  hiện tại là chưa đủ để trộm được  $k$  nhà.*

---

The time complexity is  $O(n \times \log m)$ , and the space complexity is  $O(1)$ .

*Độ phức tạp thời gian là  $O(n \times \log m)$ , và độ phức tạp không gian là  $O(1)$ .*

Where  $n$  and  $m$  are the length of the array  $\text{nums}$  and the maximum value in the array  $\text{nums}$ , respectively.

*Trong đó  $n$  là độ dài mảng  $\text{nums}$ , và  $m$  là giá trị lớn nhất trong  $\text{nums}$ .*

```
class Solution {
public:
    int minCapability(vector<int>& nums, int k) {
        auto f = [&](int x) {
            int cnt = 0, j = -2;
            for (int i = 0; i < nums.size(); ++i) {
                if (nums[i] > x || i == j + 1) {
                    continue;
                }
                ++cnt;
                j = i;
            }
            return cnt >= k;
        };
        int left = 0, right = *max_element(nums.begin(), nums.end());
        while (left < right) {
            int mid = (left + right) >> 1;
            if (f(mid)) {
                right = mid;
            } else {
                left = mid + 1;
            }
        }
        return left;
    }
};
```