

House Robber II

```
class Solution {
public:
    int rob(vector<int>& nums){
        if (nums.size() == 1) return nums[0];
        if (nums.size() == 2) return max(nums[0],nums[1]);
        vector<int> n1 = vector<int>(nums.begin()+1,nums.end());
        vector<int> n2 = vector<int>(nums.begin(),nums.end()-1);

        return max(rob1(n1),rob1(n2));
    }

    int rob1(vector<int>& nums) {
        int size = nums.size();
        if (size == 1) return nums[0];
        if (size == 2) return max(nums[0],nums[1]);

        int *awns = new int[size+1];
        awns[size] = 0;
        awns[size-1] = nums[size-1];
        awns[size-2] = nums[size-2];
        int db = 0;
        int tb = 0;
        for (int i = size-3;i>=0;i--){
            db = nums[i] + awns[i+2];
            tb = nums[i] +awns[i+3];
            awns[i] = max(db,tb);
        }

        return max(awns[0],awns[1]);
    }
}
```

```
};
```