Java

```java
class Solution {
    public long maxAlternatingSum(int[] nums, int[][] swaps) {

    }
}
```

JavaScript

```javascript
/**
 * @param {number[]} nums
 * @param {number[][]} swaps
 * @return {number}
 */
var maxAlternatingSum = function(nums, swaps) {

};
```

TypeScript

```typescript
function maxAlternatingSum(nums: number[], swaps: number[][]): number {

};
```

C++

```cpp
class Solution {
```

```
public:
    long long maxAlternatingSum(vector<int>& nums, vector<vector<int>>& swaps) {

    }
};
```

C#

```
public class Solution {
    public long MaxAlternatingSum(int[] nums, int[][] swaps) {

    }
}
```

Kotlin

```
class Solution {
    fun maxAlternatingSum(nums: IntArray, swaps: Array<IntArray>): Long {

    }
}
```

Go

```
func maxAlternatingSum(nums []int, swaps [][]int) int64 {

}
```
