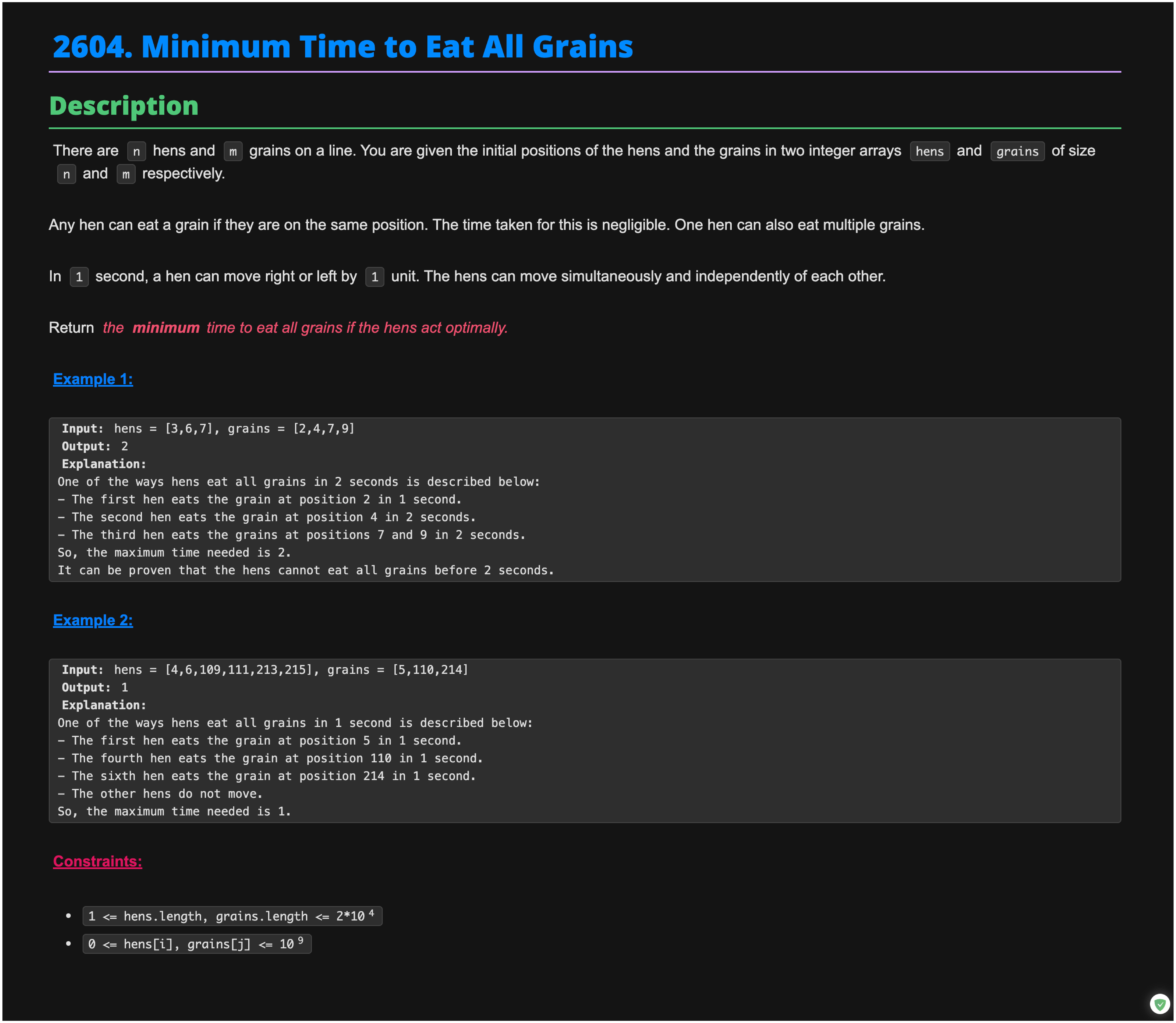

2604. Minimum Time to Eat All Grains

Description

There are `n` hens and `m` grains on a line. You are given the initial positions of the hens and the grains in two integer arrays `hens` and `grains` of size `n` and `m` respectively.

Any hen can eat a grain if they are on the same position. The time taken for this is negligible. One hen can also eat multiple grains.

In `1` second, a hen can move right or left by `1` unit. The hens can move simultaneously and independently of each other.

Return *the **minimum** time to eat all grains if the hens act optimally.*

Example 1:

```
Input: hens = [3,6,7], grains = [2,4,7,9]
Output: 2
Explanation:
One of the ways hens eat all grains in 2 seconds is described below:
- The first hen eats the grain at position 2 in 1 second.
- The second hen eats the grain at position 4 in 2 seconds.
- The third hen eats the grains at positions 7 and 9 in 2 seconds.
So, the maximum time needed is 2.
It can be proven that the hens cannot eat all grains before 2 seconds.
```

Example 2:

```
Input: hens = [4,6,109,111,213,215], grains = [5,110,214]
Output: 1
Explanation:
One of the ways hens eat all grains in 1 second is described below:
- The first hen eats the grain at position 5 in 1 second.
- The fourth hen eats the grain at position 110 in 1 second.
- The sixth hen eats the grain at position 214 in 1 second.
- The other hens do not move.
So, the maximum time needed is 1.
```

Constraints:

- $1 <= \text{hens.length, grains.length} <= 2*10^{4}$
- $0 <= \text{hens[i], grains[j]} <= 10^{9}$