Partition label

St → a b a b c b a c a | d e f e g d e | h i j h k l i j

0 1 2 3 4 5 6 7 8 | 9 10 11 12 13 14 15 | 16 17 18 19 20 21 22 23

105

Partition: ~~0 8 14 18 19 22~~ 23

O(1) => 26

9	7	8

j - 23	h - 19	f - 11
i - 22	e - 15	a - 8
l - 21	d - 14	c - 7
k - 20	g - 13	b - 5

Space ↓ better

Buddy nim

Charlie
Alice
Bob

Charlie

Alice

Bob

→ (I) Coins 1 ≠ Coins 2 ⇒ Alice

(II) Coins 1 = Coins 2

Structure same ⇓ Bob

Structure different ⇓ Alice

A: 0 0 2 3 4 5 6

B: 0 2 3 4 5 6

Maximum Product Subarray

0	1	2	3	4	5	6	7
2	3	-1	0	4	-1	3	-4

→ { prefixProd or suffixProd } — Single iteration

→ 0 → split

Consecutive no. Sum

K=1 ✓

K=2 ✓

K=3 ✓

K=4 ✗

K=5

N=15

$$n + (n+1) + (n+2) + \cdots + (n+(k-1)) = N$$

$$kn + (1 + 2 + 3 + \cdots + k-1) = N$$

$$kn + \frac{(k-1)*k}{2} = N$$

$$n = \left(N - \frac{k+(k-1)}{2}\right) / k$$

$$N - k + \frac{k-1}{2} > 0$$

$$N > k + \frac{(k-1)}{2}$$

$$N > \frac{k^2}{2} \Rightarrow \sqrt{2N} > k$$

```
for(int k=1; k<sqrt(2N); k++)
{
  if(((N - k*(k-1)/2) % k == 0)
    count++;
}
return count
```