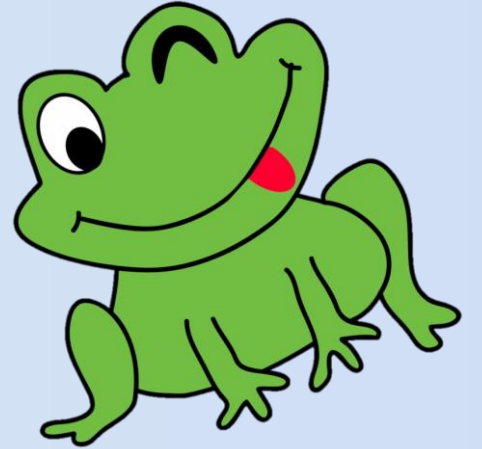


*TMITalk: Too Much Information Talk*

**점프하는 개구리 문제:**



**점프하는 개구리는  
순환할 수 있을까?**



# 점프하는 개구리는 순환할 수 있을까?

2

- 점프하는 개구리: 흥미로운 수학 문제
  - 1부터  $N(N \geq 2)$  까지의 숫자가 적힌 동전이 원 위에 놓여 있다.
  - 점프하는 개구리는 해당 동전 위에서
    - 동전에 적힌 숫자 크기에 따라 시계 방향으로 점프를 하며 여행을 한다.
  - 만약 개구리가 1이 적힌 동전 위에 있다면,
    - 개구리는 원 위에 놓인 모든 동전에 방문할 수 있을까?
  - 이런 방법으로 모든 동전을 방문할 수 있는 순환 배열을 가진
    - 숫자  $N$ 을 모두 찾아라.

문제 출처: 2021 Pan-American Girl's Mathematical Olympiad

[https://artofproblemsolving.com/community/c2499895\\_2021\\_panamerican\\_girls\\_mathematical\\_olympiad](https://artofproblemsolving.com/community/c2499895_2021_panamerican_girls_mathematical_olympiad)

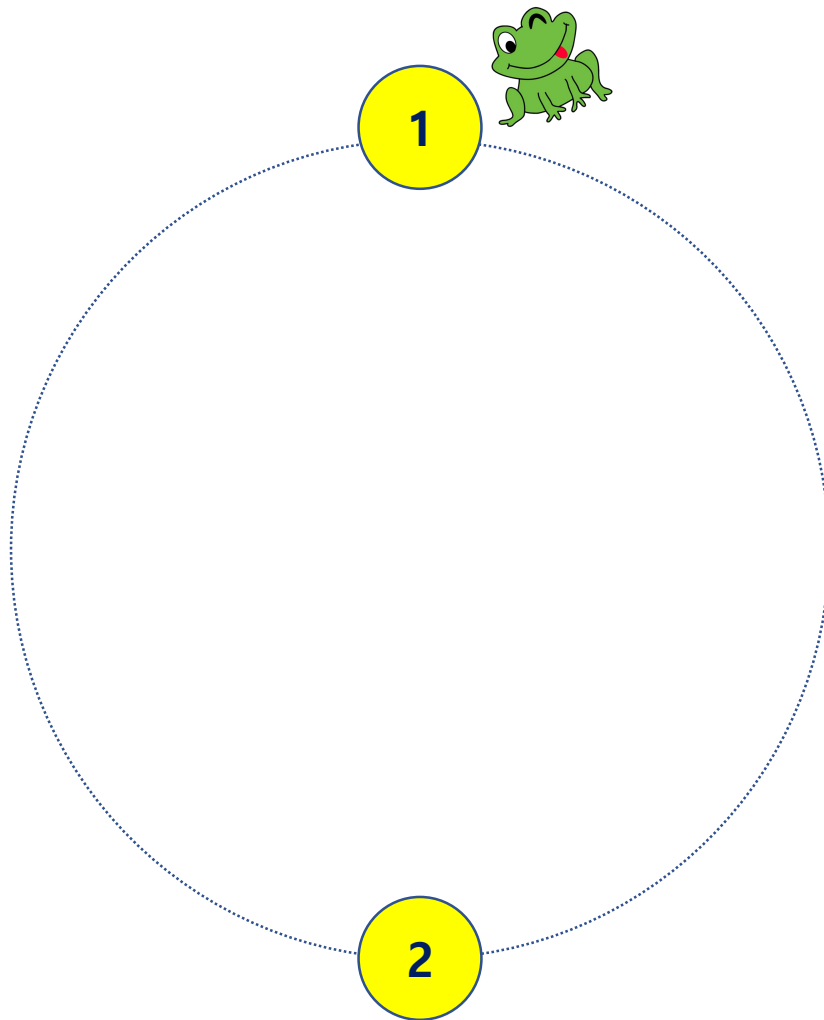




# 점프하는 개구리는 순환할 수 있을까?

3

$$N = 2$$

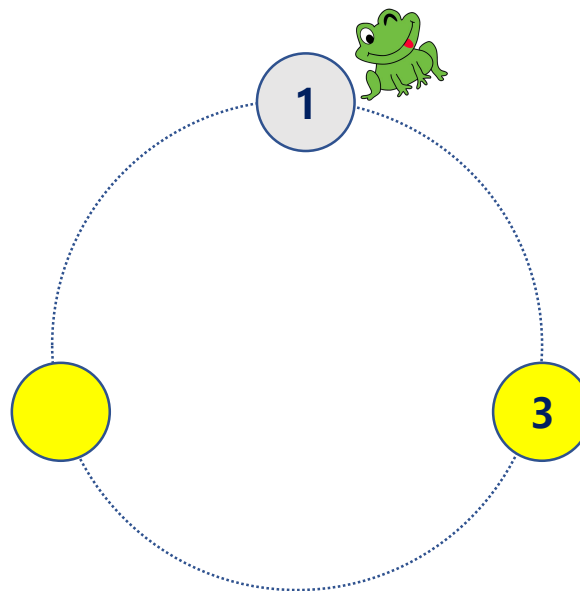
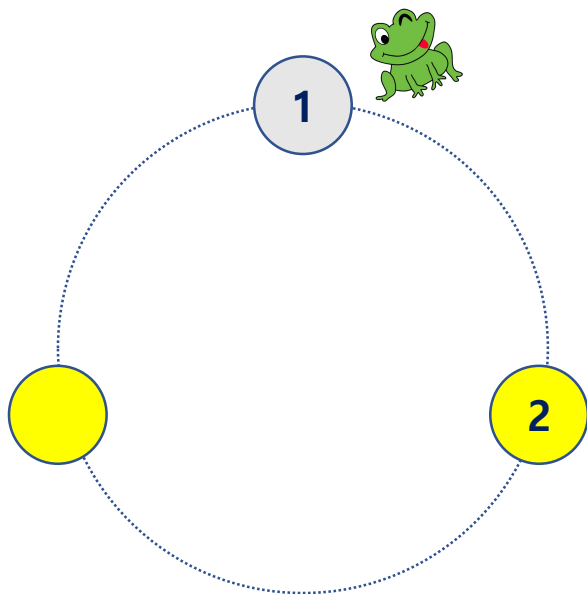




# 점프하는 개구리는 순환할 수 있을까?

4

$$N = 3$$





# 점프하는 개구리는 순환할 수 있을까?

5

$$N = 4$$

[1, 2, 3, 4]

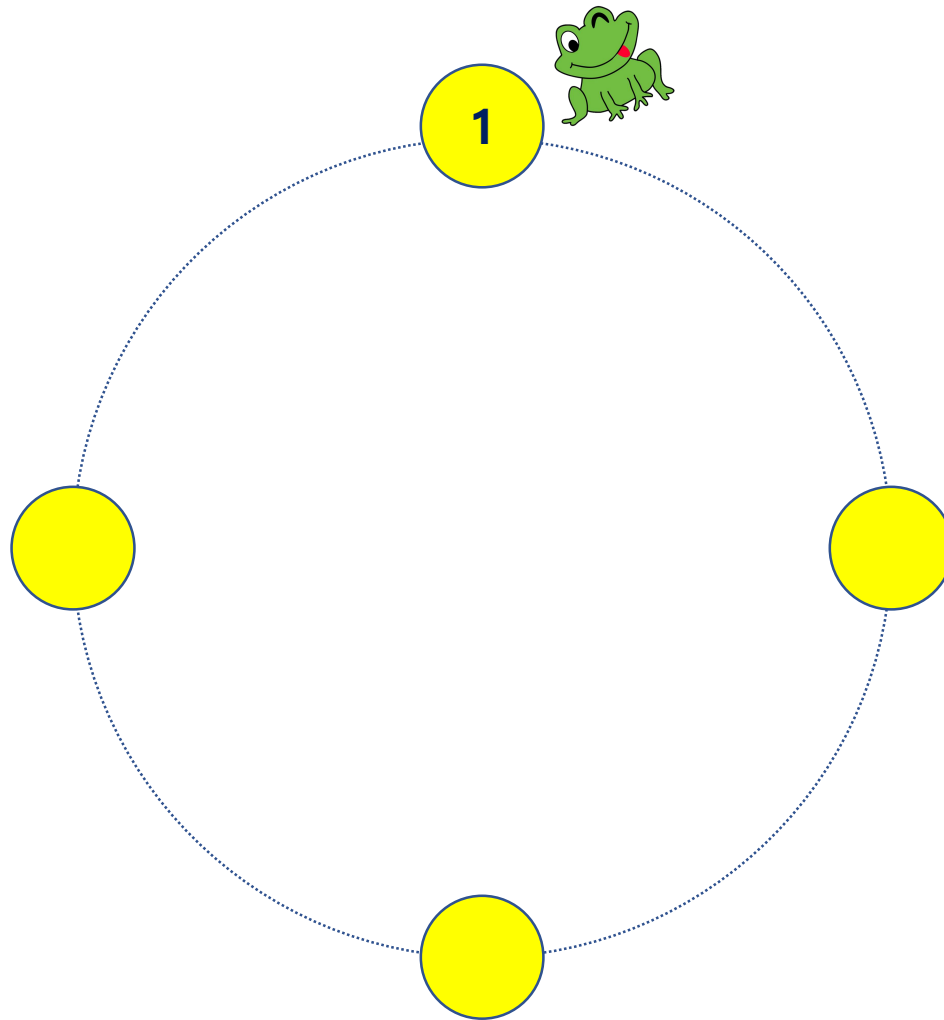
[1, 2, 4, 3]

[1, 3, 2, 4]

[1, 3, 4, 2]

[1, 4, 2, 3]

[1, 4, 3, 2]





# 점프하는 개구리는 순환할 수 있을까?

6

- 수학적으로는...
  - 정답:  $N$ 이 짝수일 때 싹가능!
    - $N$ 이 홀수일 때는 불가능함을 증명
    - $N$ 이 짝수일 때는 가능함을 증명





# 점프하는 개구리는 순환할 수 있을까?

7

- 수학적 증명보다 아름다운 코딩에 도전해 봅시다!
  - N이 주어졌을 때 모든 동전을 방문할 수 있는 순환 배열의 경우의 수는?
    - 순환 배열의 모든 경우의 수에 대해 방문 가능한 지 확인





# 점프하는 개구리는 순환할 수 있을까?

8

```
circle = [1, 2, 3, 4]  
print(visitable(circle))
```

```
circle = [1, 2, 4, 3]  
print(visitable(circle))
```







# 점프하는 개구리는 순환할 수 있을까?

9

```
def visitable(circle):  
    pos = 0  
    while circle[pos] > 0:  
        next_pos = (pos + circle[pos]) % len(circle)  
        circle[pos] = 0  
        pos = next_pos  
    return sum(circle) == 0
```





# 점프하는 개구리는 순환할 수 있을까?

10

```
from itertools import permutations

n = 4
nums = [i for i in range(2, n + 1)]
perms = list(permutations(nums))
for i in range(len(perms)):
    circle = [1] + list(perms[i])
    print(circle)
    if visitable(circle[:]):
        print("solution:", circle)
```





# 점프하는 개구리는 순환할 수 있을까?

11

```
from itertools import permutations

for n in range(1, 13):
    nums = [i for i in range(2, n + 1)]
    perms = list(permutations(nums))
    count = 0
    for i in range(len(perms)):
        circle = [1] + list(perms[i])
        if visitable(circle[:]):
            count += 1
    print(n, ":", count)
```





# 점프하는 개구리는 순환할 수 있을까?

12

- 더 효율적인 알고리즘은 없을까?
  - N이 주어졌을 때 순환 배열의 경우의 수:  $N - 1$ 
    - $O(N!)$
  - 백트래킹으로 가지치기를 하면?
    - DFS: 1번부터 점프를 시작해서 N번의 점프를 할 때까지 탐색
    - 가지치기: 만약 점프한 위치가 이미 방문한 동전이면 되돌아 감
  - 백트래킹의 구현과 해설: 커밍쑈!
    - 직접 한 번 도전해 보세요~





*Any Questions?*