

알튜비튜

구현 & 코너케이스

최근 대다수의 코딩테스트에서 구현 능력을 보기 위한 문제들이 출제되고 있습니다. 또한 대다수의 기업에서 코딩테스트 시 히든 케이스를 제공하지 않습니다. 따라서 이런 문제를 만났을 때 어떻게 접근하면 좋을지, 어떤 코너 케이스가 있을지 함께 생각해보면서 풀어보는 시간을 가져보겠습니다.

구현

- 문제에서 시키는 대로 코딩하면 되는 유형
- 언뜻 풀이(아이디어)를 생각하는 쉽지만, 실제 소스코드로 옮기기는 어려운 유형

- 조건문, 반복문, 재귀함수 + (어렵지 않은 알고리즘 개념 - ex. 브루트포스)
- 문자열, float형 다루기
- 문제에 적합한 자료구조 선택
- 언어에 따라 적절한 라이브러리/함수 사용

/<> 10804번 : 카드 역배치 - Bronze 2

문제

- 1부터 20씩 숫자가 하나씩 쓰인 20장의 카드
- 입력으로 주어지는 구간에 있는 카드를 역배치 하는 문제

풀이

- 특별한 알고리즘 없이, 문제에서 시키는 대로 카드를 뒤집어 주기만 하면 된다!
- C++ <algorithm>헤더의 `reverse()`
- Python3 `reversed()`, 인덱스 슬라이싱

구현 문제 - 예시 (팰린드롬 만들기)

/<> 1213번 : 팰린드롬 만들기 - Silver 4

문제

- 입력으로 주어진 문자열의 순서를 바꿔 팰린드롬 문자열을 만드는 문제

풀이

- 역시 사용되는 특별한 알고리즘 X
- 문자열, 아스키코드
- 알파벳을 인덱스로 나타내서 알파벳 개수를 저장해서 푸는 것이 핵심!

C++

```
count[arr[i] - 'A']++;
```

Python3

```
count[ord(arr[i]) - ord('A')] += 1
```

✓ collections.Counter 사용

/<> 2503번 : 숫자야구 - Silver 4

문제

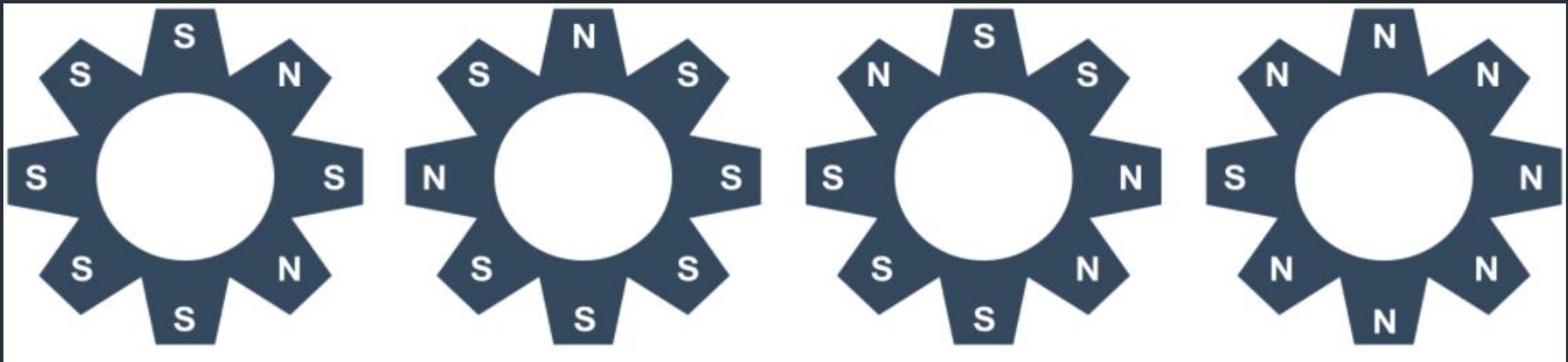
- 영수는 1에서 9까지의 서로 다른 숫자 세 개로 구성된 세 자리 수를 결정
- 민혁이는 1에서 9까지의 서로 다른 숫자 세 개로 구성된 세 자리 수를 추측
- 위치와 숫자 모두 일치 -> 스트라이크
- 숫자는 포함되었으나 위치가 불일치 -> 볼

풀이

- 알고리즘: 브루트포스 (완전탐색)
- 각 숫자가 일치하는지 어떻게 확인? -> 정수를 쪼개...? -> 정수보단 다른 자료형이..!
- 결국 구현이 중요!

/<> 14891번 : 톱니바퀴 - Gold 5

문제



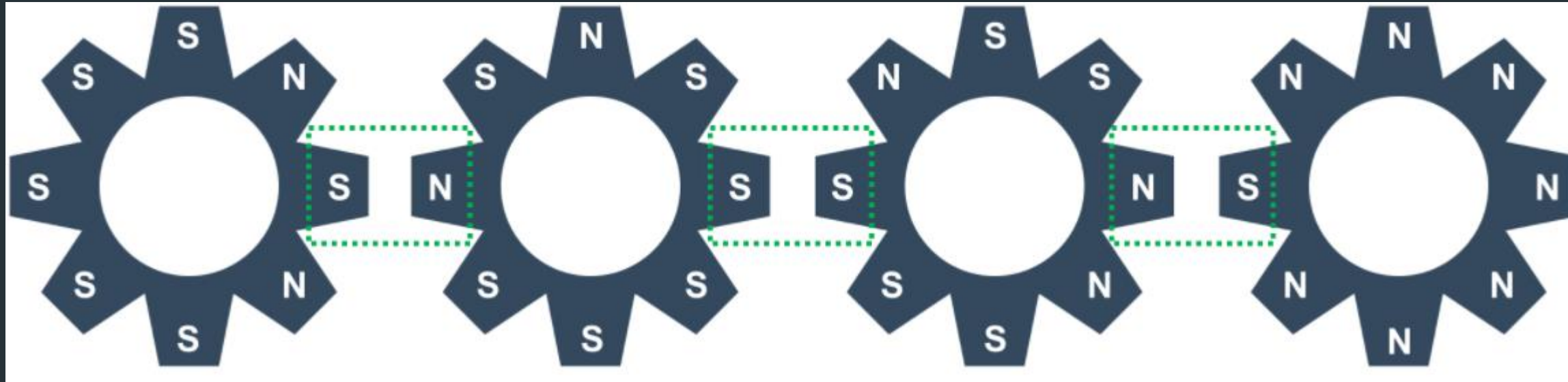
- 가장 왼쪽 톱니바퀴부터 각각 입력의 1, 2, 3, 4번째 줄에 해당
- 각 줄에는 12시 방향부터 시계방향으로 N극은 0, S극은 1로 표기
- 5번째 줄에는 회전 횟수, $1 \leq K \leq 100$
- 이후로는 톱니바퀴의 번호($1 \leq N \leq 4$)와 회전방향(시계: 1, 반시계:-1)

예제 입력

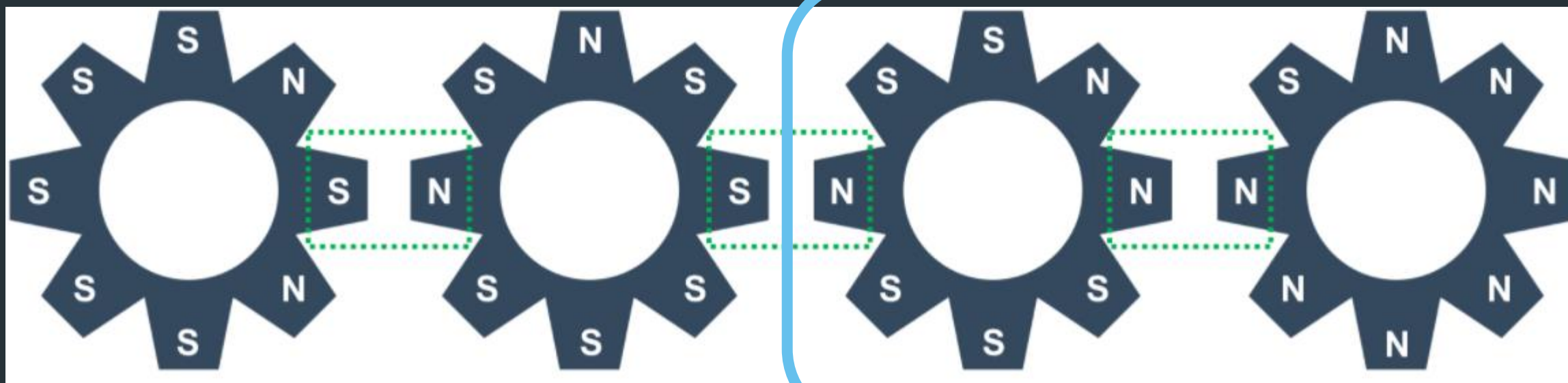
```
10101111
01111101
11001110
00000010
2
3 -1
1 1
```

14891번 : 톱니바퀴 - Gold 5

- 톱니바퀴 A를 회전할 때, 인접한 톱니바퀴 B와 서로 맞닿은 톱니의 극이 다를 경우, B는 A가 회전한 방향과 반대방향으로 회전



- 3번 톱니바퀴를 -1(반시계)방향으로 회전

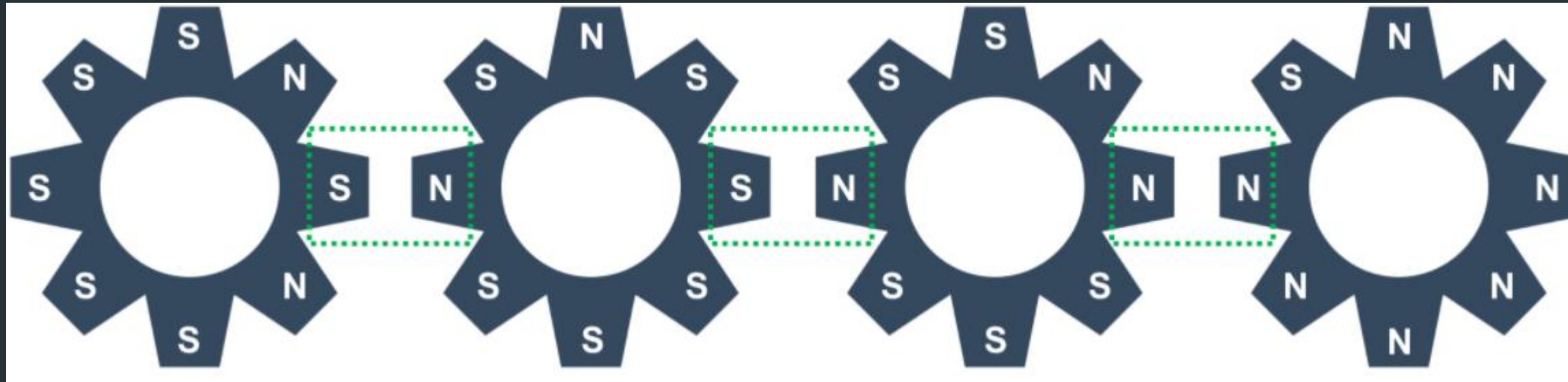


예제 입력

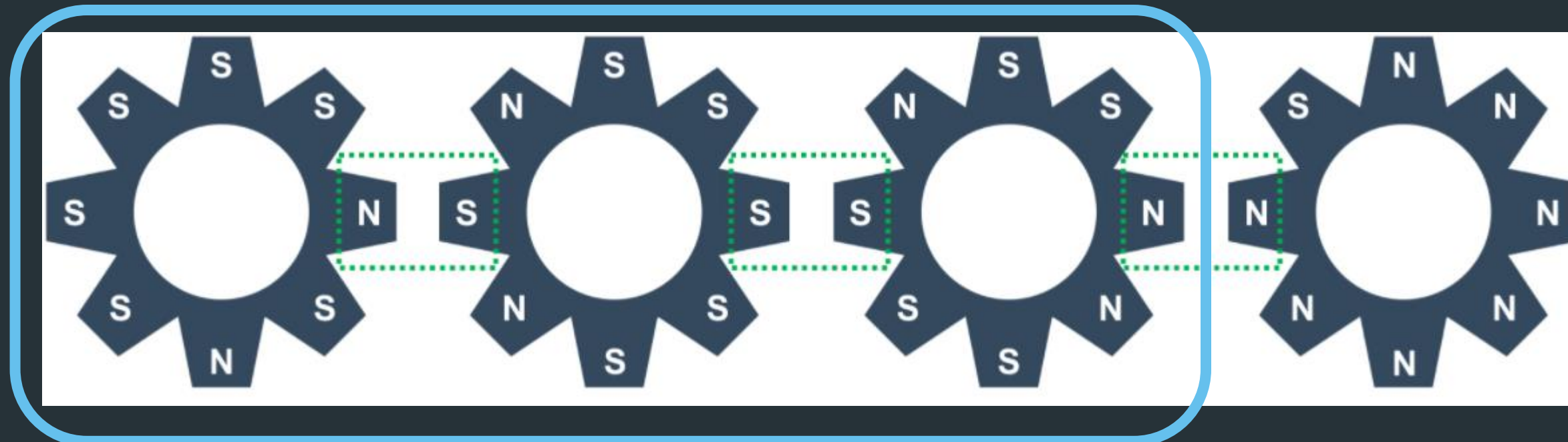
```
10101111
01111101
11001110
00000010
2
3 -1
1 1
```


14891번 : 톱니바퀴 - Gold 5

- 톱니바퀴 A를 회전할 때, 인접한 톱니바퀴 B와 서로 맞닿은 톱니의 극이 다를 경우, B는 A가 회전한 방향과 반대방향으로 회전

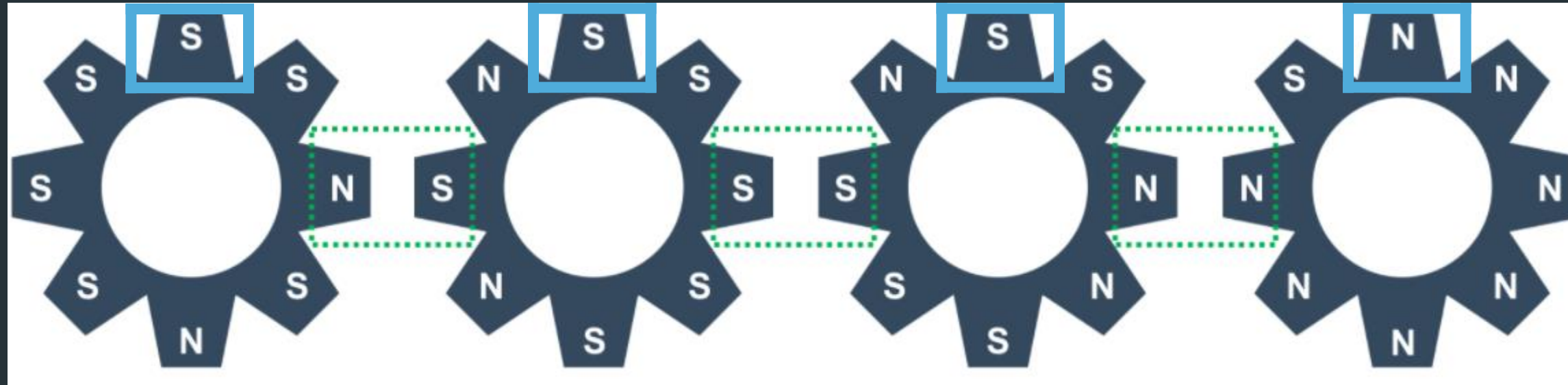


- 1번 톱니바퀴를 1(시계)방향으로 회전



예제 입력

```
10101111
01111101
11001110
00000010
2
3 -1
1 1
```

예제 출력

7

- 총 K번 회전시킨 이후에 네 톱니바퀴의 점수의 합을 출력
- 1번 톱니바퀴의 12시방향이 N극이면 0점, S극이면 1점
- 2번 톱니바퀴의 12시방향이 N극이면 0점, S극이면 2점
- 3번 톱니바퀴의 12시방향이 N극이면 0점, S극이면 4점
- 4번 톱니바퀴의 12시방향이 N극이면 0점, S극이면 8점

- 어떤 자료구조를 사용해야 할까?
→ "회전"을 해야하니 회전을 구현하기 편한 자료구조에는 뭐가 있을까?
- 함수는 어떻게 구성하면 좋을까?
→ 필요한 과정이.. 우선 맞닿은 톱니 확인해서 회전할지 봐야하고 ..
→ 회전 방향도 한 방향이 아니라 계속 바뀌는데, 어떻게 관리하면 좋을까?
→ 회전은 어떻게 구현하지?
- 혹시 사용할 수 있는 라이브러리/함수가 있을까?

코너 케이스

- 예제는 다 맞았는데...
 - 어렵지 않은데 정답률이 낮은 문제들
 - 알고리즘 유형과 상관 없이 고려해야 함
 - 코딩테스트에서는 주어지지 않는 경우가 대부분 -> 스스로 생각해내야 함
-
- 코드를 짜면서 생각하지 못한 반례 - 최솟값, 최댓값
 - 범위가 너무 클 때 - 자료형의 범위
 - 입력이 간단할 경우 문제의 조건에 따라 나올 수 있는 여러 상황 생각해서 테스트 케이스 만들어보기!

예제는 다 돌아갑니다.

c++ 반례좀 찾아주세요 해결

파이썬 주어진 예제는 전부 맞는데 어느 부분이 틀린건지 모르겠습니다 ㅠㅠ 해결

예제 답은 전부 맞는데 해결

(파이썬) 반례 부탁드립니다. (시험 감독 문제) 해결

어떤 부분에서 틀린걸까요..?

c언어) 모든 예제가 맞는데 틀렸습니다 해결

어디가 틀렸는지 모르겠어요 ㅜㅜ

왜 틀렸는지 모르겠습니다 ㅜ

/<> 12840번 : 창용이의 시계 - Bronze 3

문제

- 주어지는 입력에 따라 시계를 **앞으로**, 혹은 **뒤로** 돌리다가 요청이 들어오면 현재 시계의 상황을 출력하는 문제

코너케이스

- $0 \leq c \leq 10,000,000$
- 시계를 조작하다가 하루의 범위를 넘어가는 경우 처리 -> 모듈러 연산
- 조작하는 시간이 **매우 크거나 작은 경우** (하루 이상을 조작하는 경우)
- 전 날로** 넘어간 경우 -> 따로 조건 처리

예제 입력

```
0 1 0
4
1 4263
3
2 1175
3
```

/<> 2108번 : 통계학 - Silver3

문제

- 입력된 숫자들의 산술평균, 중앙값, 최빈값, 범위를 출력

코너케이스

- round() 함수 사용 시 주의 (C++ : -0, Python3: 경계(0.5)에서의 반올림)
- 최빈값이 유일한 경우 고려

코너케이스 문제 - 예시 (큰 수 A+B)



/<> 10757번 : 큰 수 A+B - Bronze 5

문제

- 두 정수 A, B의 합을 출력하는 문제

코너케이스

- $0 < A, B < 10^{10000}$
- A와 B의 자리수가 다른 경우

예제 입력

```
9223372036854775807 9223372036854775808
```

예제 출력

```
18446744073709551615
```


/<> 13458번 : 시험 감독 - Bronze 2

문제

- N개의 시험장
- B명을 감시할 수 있는 총감독관
- C명을 감시할 수 있는 부감독관
- 총 감독관은 각 시험장마다 1명만 배치
- 부감독관은 여러 명 배치 가능
- 필요한 감독관 수의 최솟값

예제 입력

```
3 (N)
3 4 5 (시험장 별 응시자 수)
2 2 (B, C)
```

예제 출력

```
7
```

- 코너 케이스에 유의하여 생각한 대로 구현
- 주어진 예제가 맞는지 확인
- 예제가 다 맞았고, 놓친 조건이 없는 것 같다! -> 제출해서 결과를 확인
- 틀렸을 경우, 반례를 고민

정리

- 구현문제를 잘 풀기 위해서는, 언어에 대한 이해가 높고, 경험이 많아야 합니다. (매주 구현 문제가 필수로 나가는 이유...)
- 가장 기본적인 반례, 입력 범위의 양 끝을 주의
- 연산 범위를 확인하여, 알맞은 자료형 설정
- 조건문이 모든 경우를 커버하는지 확인

3문제 이상 선택

- /<> 11723번 : 집합 - Silver 5
- /<> 1316번 : 그룹 단어 체커 - Silver 5
- /<> 19636번 : 요요 시뮬레이션 - Silver 4
- /<> 20923번 : 숫자 할리갈리 게임 - Silver 1
- /<> 1009번 : 분산처리 - Bronze 1

코드리뷰 O 마감 ~ 3월 28일 월요일 낮 12시

코드리뷰 X 마감 ~ 3월 28일 월요일 밤 12시 (28일에서 29일로 넘어가는 자정)

추가제출 마감 ~ 3월 29일 화요일 밤 12시 (29일에서 30일로 넘어가는 자정)