



① 구현

- 문제에서 시키는대로 코딩하면 되는 유형
- 단, 아이디어는 생각하기 쉬워도 실제 소스코드를 작성하는 것은 어려운 유형

② 코너케이스

- 문제에서 주어진 예제 외에 스스로 생각해내야 하는 반례가 존재하는 유형
- 어렵진 않은데 정답률이 매우 낮은 문제들이다.
- 알고리즘 유형과 상관없이 항상 고려해야 한다.

<고려해야 할 것들>

- 코드를 짜며 생각하지 못한 반례 ~> 최솟값, 최댓값
- 자료형의 범위 (연산 범위 확인), 입력 범위의 양끝 항상 확인
- 입력이 너무 간단한 경우, 문제의 조건에 따라 나올 수 있는 여러 상황 생각해서 테스트케이스 만들어보기
- 조건문이 모든 경우를 커버하는지 확인

~> 구현 & 코너케이스 문제를 잘 풀기 위해서는 언어에 대한 이해가 높고, 경험이 많아야 함!