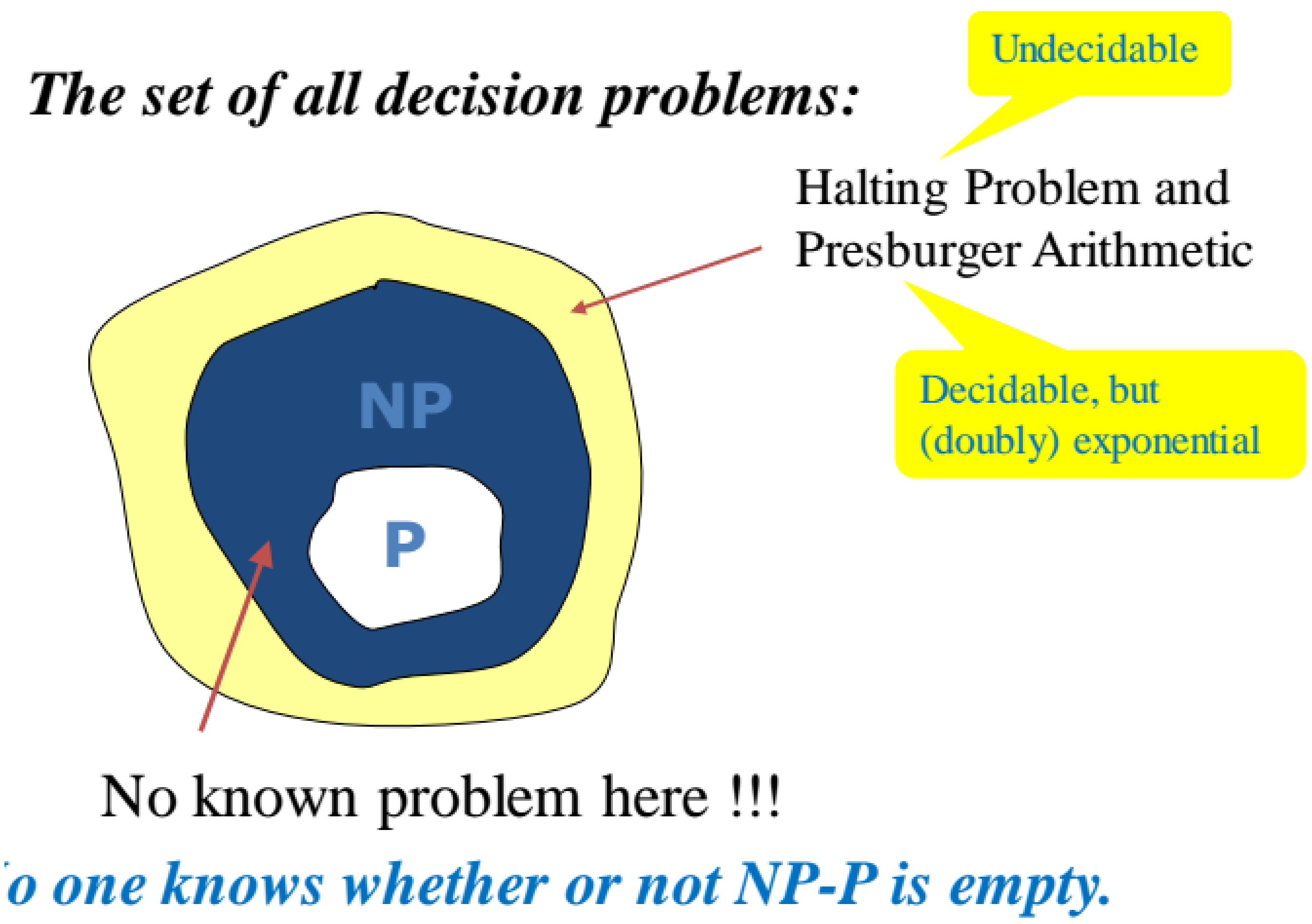
P-NP이론

어려운 문제를 쉬운 문제로 바꿔서 풀면 되지 않을까?

그렇다면 어려운 문제를 쉽게 바꿔주는 열쇠가 존재할까?
이 열쇠 존재의 여부를 증명하면 어려운 문제를 쉽게 바꿀 수 있으므로 그것을 찾으면 된다. 증면되지 않으면 쉽게 바꿀 수 없다는 걸 알기 때문에 시간을 낭비하지 않아도 된다.

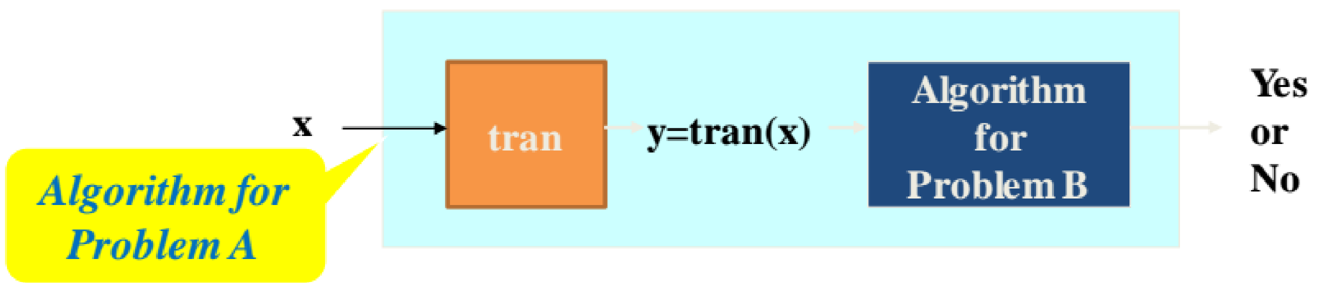
https://inverse90.tistory.com/entry/PNP-NP-Hard-NP-Complete



여기서 의문점은 NP에는 속하지만 P에는 속하지 않는 문제가 명확히 존재하는가??
→ 해당 문제를 P로 풀 수 없다는 걸 증명하면 된다.

++ Halting Problem과 Presburger Arithmetic은 NP에 포함되지 않는 더 큰 범위의 문제(미제?)이다.

page 15



x가 원래는 Problem B를 푸는데 사용하려면 매우 힘듦
하지만 x에 어떤 tran()를 적용하고 이 결과값 y가 Problem의 인스턴스 인 경우를 트랜스포메이션 알고리즘이라고 한다.
즉 기존의 변수를 문제에 맞게 변환하는 과정이고 변환하는 과정 + Problem B를 푸는 과정을 합친 알고리즘을 만들면 x가 Problem B를 풀게할 수 있다

page 16

Problem A: 배열의 가장 작은 값이 배열 절반 기준 뒤쪽에 있냐?
ex {2,3,4,5,6,7,8,9} → 가장 작은 값이 6,7,8,9 중에있냐? → No

Probelm B: 배열의 가장 큰 값이 배열 절반 기준 뒤쪽에 있냐?

problem a가 no이면 problem b는 yes

page 17

B를 풀기만 하면 A를 풀 수 있다. (따라서 B가 풀기 더 어렵고 중요함)

9.1 B가 P로 풀리면 A또한 P로 풀린다는 것이 정의

page 18 모든 NP에 있는 다른 문제가 B에 의해 풀 수 있다?

page 19 A reduce B reduce C 인 경우에도 NP-Complete이다.
한 문제에 대해서만 풀 수 있다면

page 20이 제일 중요

NP - P = 공집합인지는 모름 → 유력하게 공집합이 아닐거라고 생각하기는함.

NPc는 NP를 모아놓은 것들의 집합

NPc는 NP의 진부분집합이다.

$$\text{If } P = NP,\ NP_C \subset P.$$
$$\text{If } P \subset NP,\ NP_C \cap P = \varnothing.$$

P가 NP의 진부분 집합이라면 NPc는 P로 풀수없는 아주 어려운 문제 즉, NPc와 P는 공집합 관계이다.

모든 NP의 문제는 NP-hard에 reduce

NP-hard를 P로 푼다면 역시 P = NP를 증명하는것과 같다.

NP-complete를 P로 풀어도 역ㄱ시 P=NP를 증명하는것과같다.