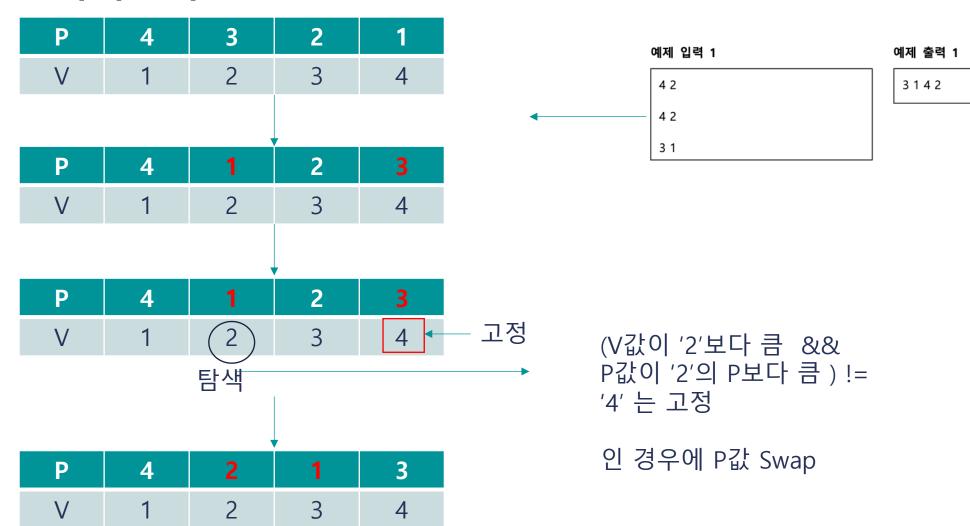
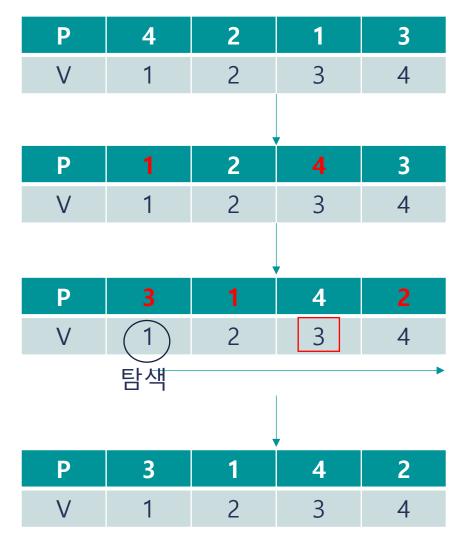
#### ❖ 조건

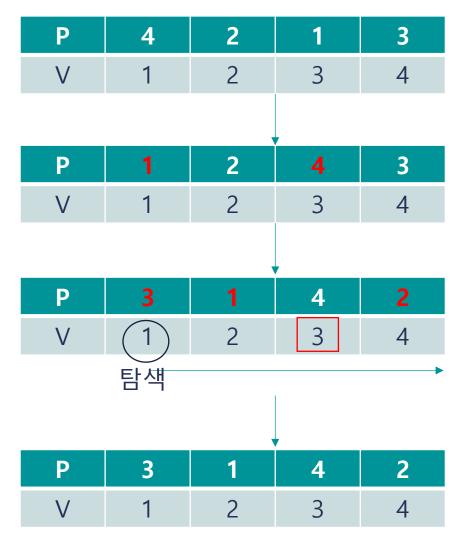
- 다음 슬라이드는 PQ를 Queue로 구현, NodeType을 Priority, info, next 세 변수로 수 정하여 구현한 예시입니다.
- Main.cpp의 짜여진 코드로만 돌아가면 되며, Ordering 함수를 구현하기위해 추가적 인 변수, 함수 추가는 모든 가능합니다. (단 STL library 사용 금지)
- 이 help-slide의 알고리즘은 참고용 일뿐 정답이 아닙니다.
  - 다른 더 좋은 알고리즘이나 본인이 개발한 알고리즘 사용을 적극 지향합니다.
  - 꼭 이 알고리즘을 구현 할 필요는 없습니다.
- PQ를 구현하는데 Queue말고 다른 자료구조(minheap, Stack)을 사용해도 좋습니다.
  - 채점을 위해 main부분은 (사용자 변수 입력 -> Ordering -> 출력)의 포맷은 꼭 지키면서 수정해주시면 감사하겠습니다.

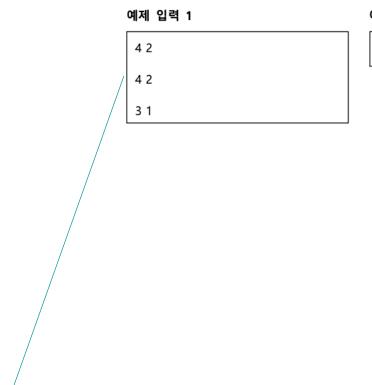






(V값이 '1'보다 큼 && P값이 '1'의 P보다 큼 )!= '3' 는 고정

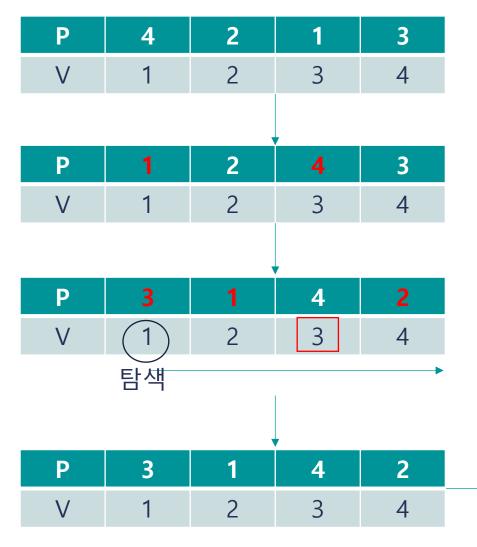


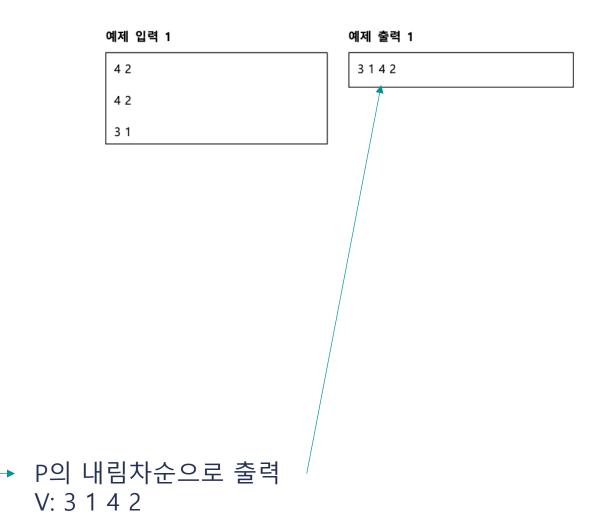


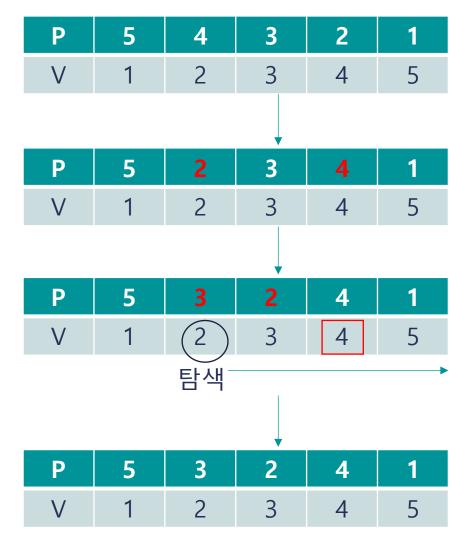
예제 출력 1

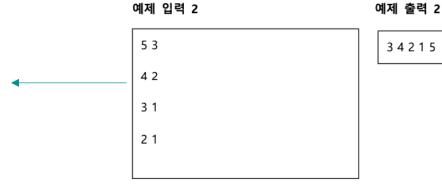
3 1 4 2

이전들의 조건을 만족하는지 확인 (4의 P > 2의 P) ? 그대로 : SWAP



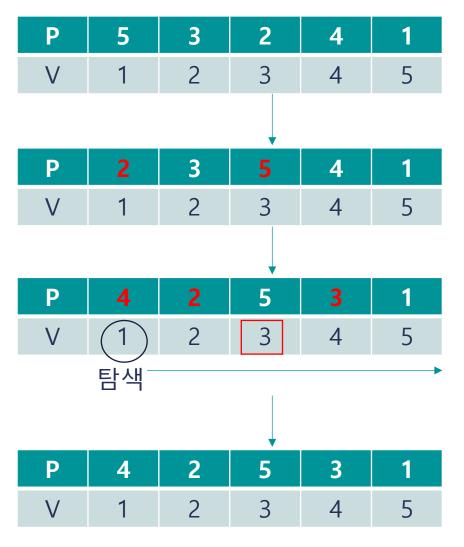


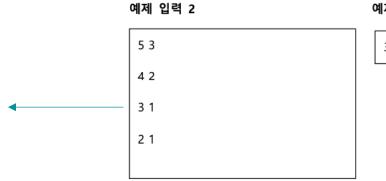




예제 출력 2

(V값이 '2'보다 큼 && P값이 '2'의 P보다 큼 )!= '4' 는 고정

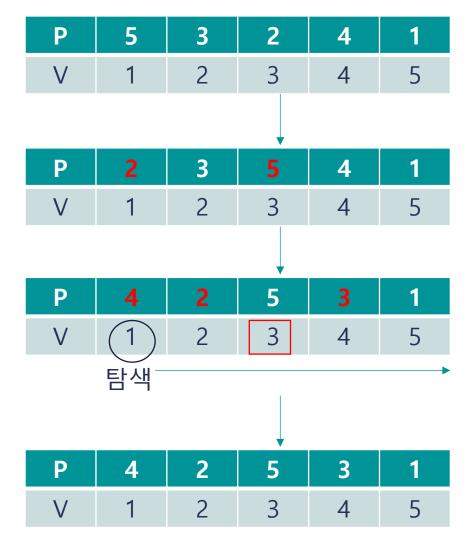


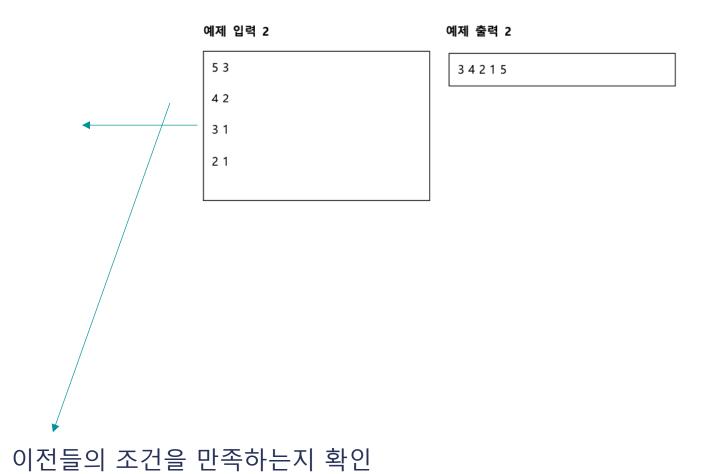


예제 출력 2

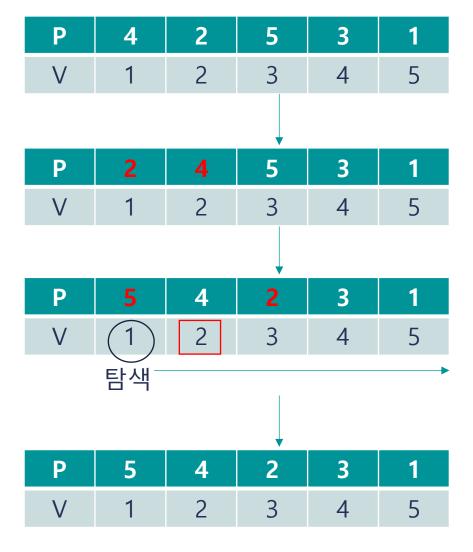
34215

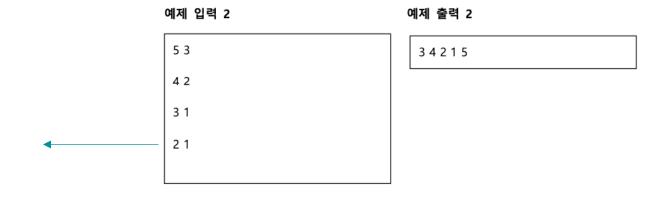
(V값이 '1'보다 큼 && P값이 '1'의 P보다 큼 )!= '3' 는 고정





(4의 P > 2의 P) ? 그대로 : SWAP





(V값이 '1'보다 큼 && P값이 '1'의 P보다 큼 )!= '2' 는 고정



예제 출력 2

3 4 2 1 5

이전들의 조건을 만족하는지 확인

예제 입력 2

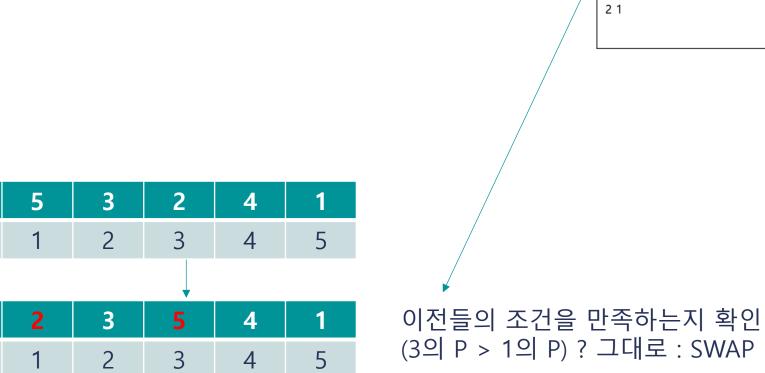
5 3

4 2

3 1

2 1

V



예제 출력 2

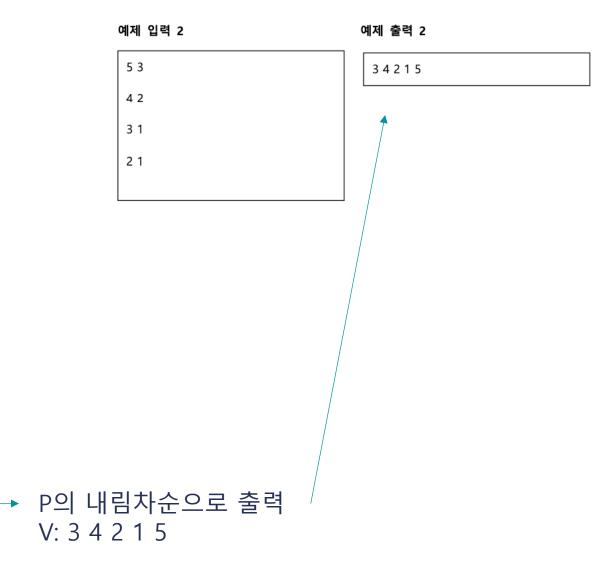
3 4 2 1 5

예제 입력 2

5 3

4 2

3 1



Р	2	3	5	4	1
V	1	2	3	4	5