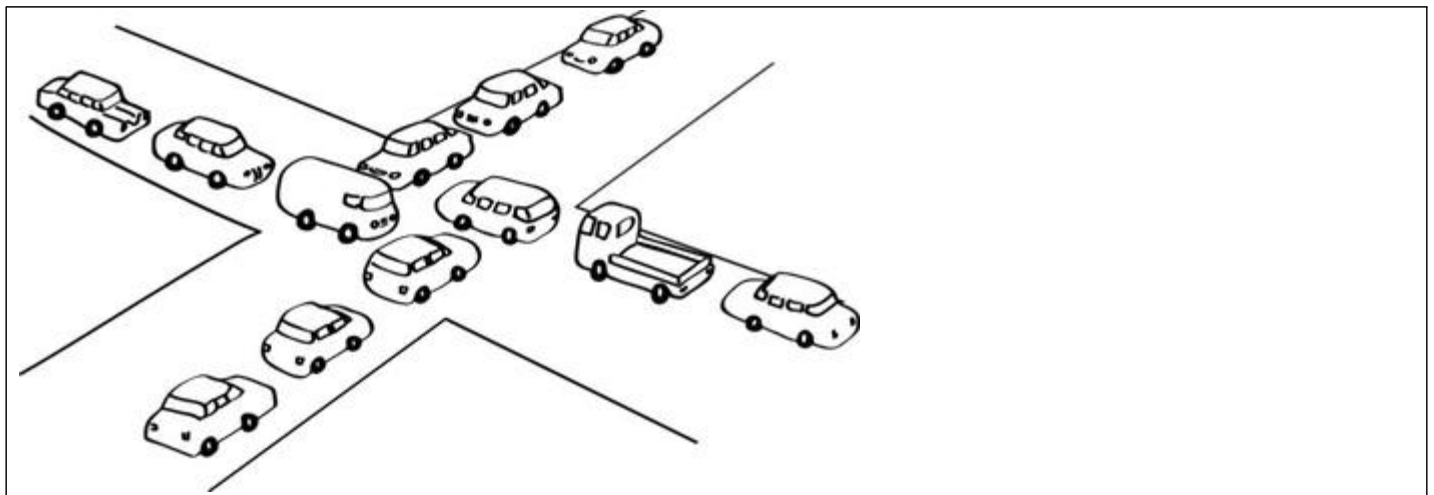
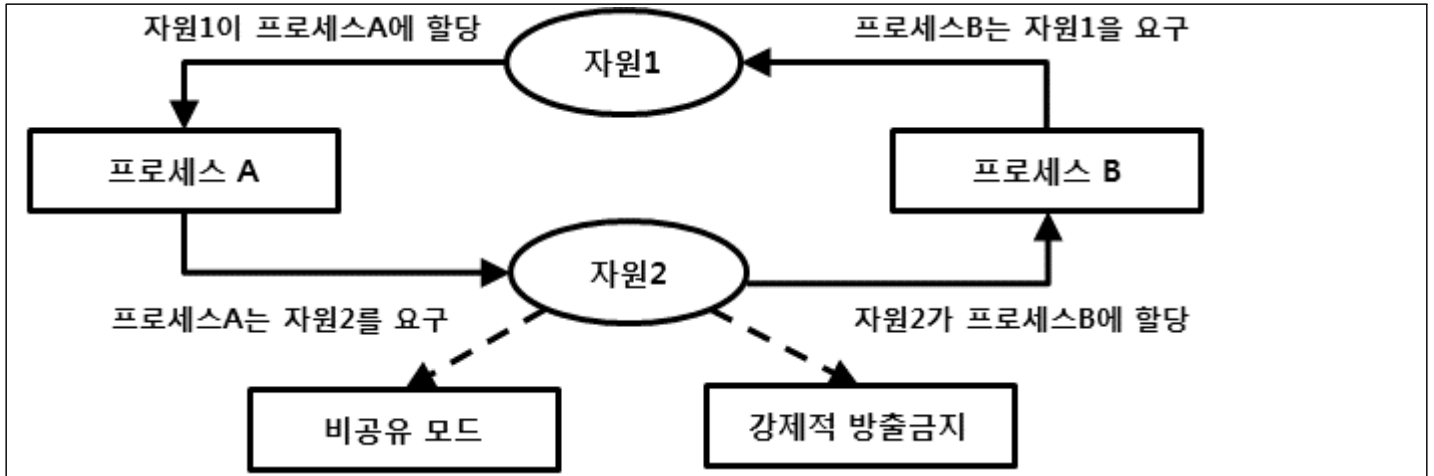


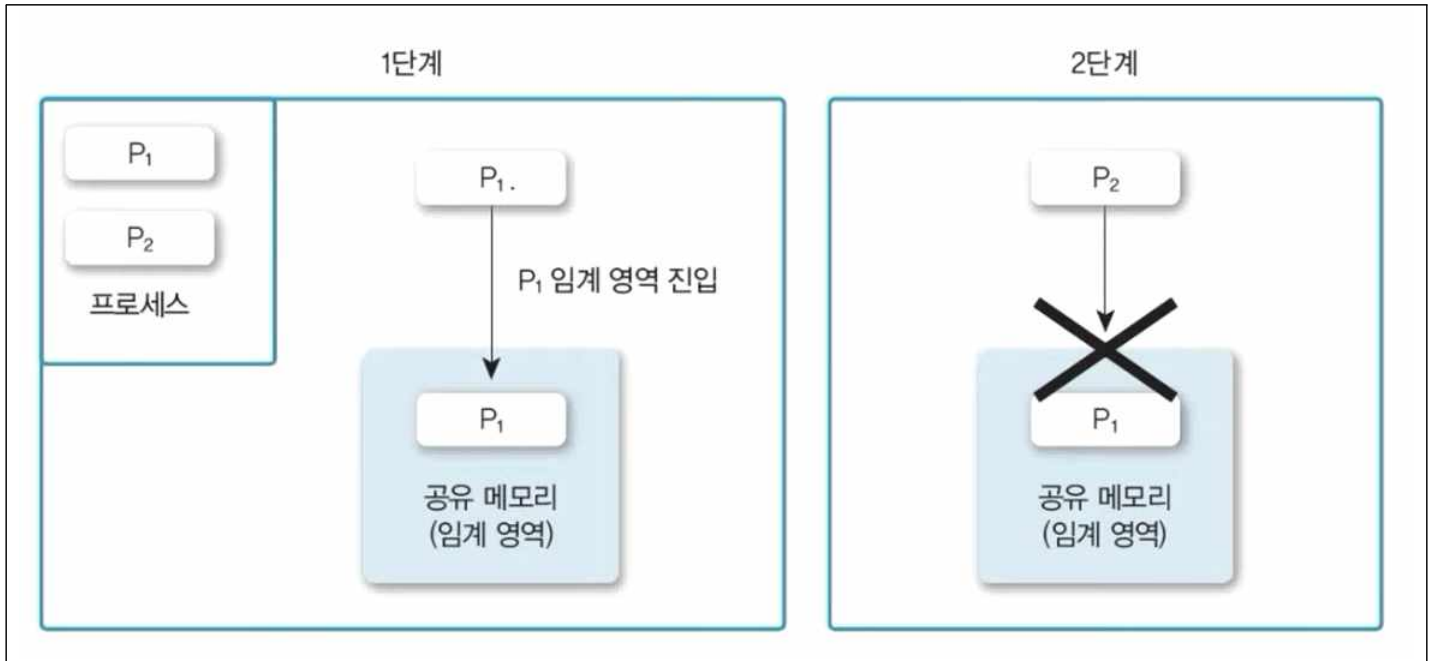
## 교착상태

### [1] 교착상태



### [2] 교착상태 발생의 필요 충분 조건

- **상호 배제(mutual exclusion):** 한 프로세스가 사용하는 자원은 다른 프로세스와 공유할 수 없는 배타적인 자원이어야 함 → 배타적 자원 사용하면 교착 상태 발생
- **비선점(non-preemptive):** 한 프로세스가 사용 중인 자원은 중간에 다른 프로세스가 빼앗을 수 없는 비선점 자원이어야 함 → 빼앗을 수 없으면 공유 안되니 교착 상태 발생
- **점유와 대기(hold and wait):** 프로세스가 어떤 자원을 할당받은 상태에서 다른 자원을 기다리는 상태여야 함 → 자원을 점유하면서 다른 자원을 기다리면 교착 상태 발생
- **원형 대기(circular wait):** 점유와 대기를 하는 프로세스 간의 관계가 원을 이루어야 함 → 프로세스들이 서로 양보하지 않아 교착 상태 발생



임계 영역

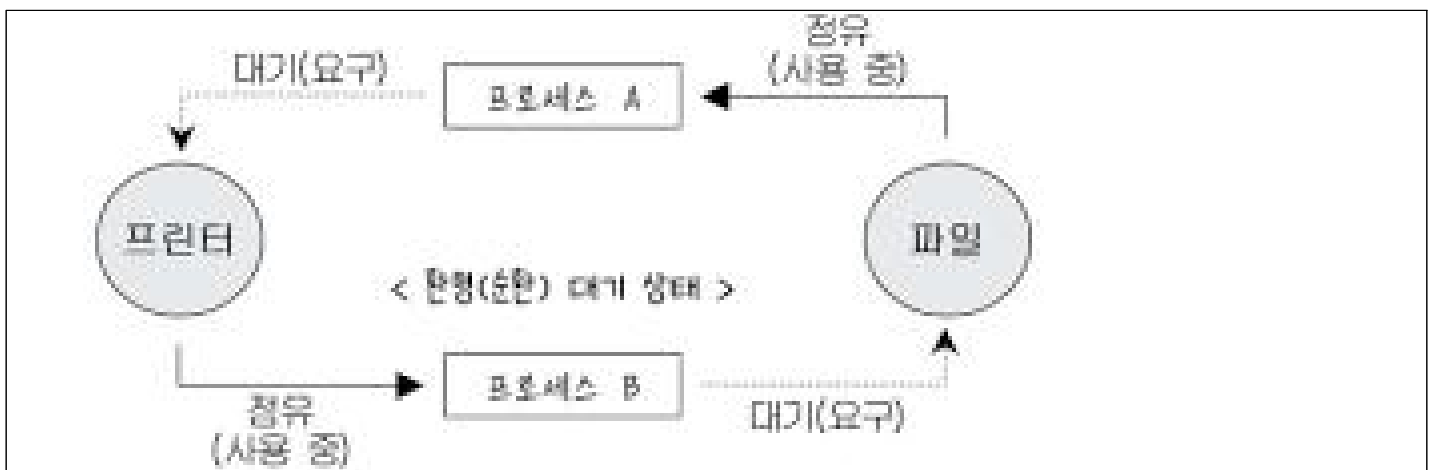
임계 자원에 접근하고 실행하는 코드 부분

프로세스가 공통 변수를 읽고 테이블을 갱신하고 파일을 수정하는 등 공유 데이터에 접근할 때 임계 영역에 있다고 함

임계 자원

두 프로세스가 동시에 사용할 수 없는 공유 자원

- 점유와 대기 -

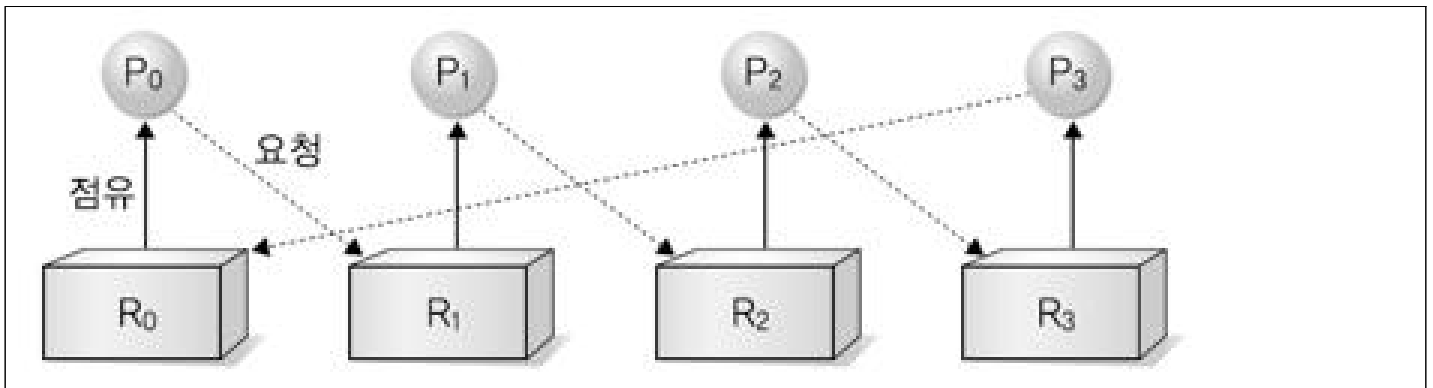


- 비선점 -

## 스케줄링 알고리즘



- 환경(순환) 대기 -



[3] 교착상태의 해결 방법

- 은행원 알고리즘 -

# 은행원 알고리즘 쉽게 이해하기!

🗨️ 캡션 원본 ...

은행원 알고리즘(Banker algorithm)은 다익스트라 알고리즘을 개발한 Edsger Dijkstra가 개발한 알고리즘이에요 (다익스트라 알고리즘은 네비게이션에서 쓰일 정도로 엄청 유명한 알고리즘입니다)

교착상태에 빠질 가능성이 있는지 판단하기 위해 상태를 '안전상태(safe state)'와 '불안전상태(unsafe state)'로 나눴습니다. (= 안전상태 개념). 즉 은행원 알고리즘에서 운영체제는 안전상태를 유지할 수 있는 요구만을 수락하고 불안전 상태를 초래할 사용자의 요구는 나중에 만족될 수 있을 때까지 계속 거절합니다.



<http://hnyang.tistory.com>