

5주차 3차시 교착상태의 해결

【학습목표】

1. 교착상태의 예방과 회피 기법에 대해 설명할 수 있다.
2. 교착상태를 탐지하고 회복하는 기법에 대해 설명할 수 있다.

학습내용1 : 교착상태 예방과 회피

1. 교착상태 예방기법

교착상태가 발생하지 않도록 시스템을 제어 하는 방법이다.

다음의 네 가지 조건 중 하나를 제거 또는 부정함으로써 교착상태 예방이 수행 된다.

- (1) 상호배제(Mutual Exclusion) 부정
- (2) 점유와 대기(Hold and Wait) 부정
- (3) 비선점(Non-Preemption) 부정
- (4) 환형대기(Circular Wait) 부정

(1) 상호배제(Mutual Exclusion) 부정

상호배제는 자원을 공유하지 않는 조건을 전제로 한다.

상호배제 부정은 한 번에 여러 개의 프로세스가 공유자원을 사용할 수 있도록 한다.

(2) 점유와 대기(Hold and Wait) 부정

프로세스 대기를 없애려면 프로세스가 실행되기 전에 필요한 모든 자원을 할당한다.

프로세스가 자원을 점유하지 않은 상태에서만 자원을 요청할 수 있도록 한다.

(3) 비선점(Non-Preemption) 부정

비선점을 부정하는 전제 조건은 이미 할당된 자원에 대해서 선점권을 갖지 않는다.

프로세스가 어떤 다른 자원을 요구할 때, 요청한 자원을 사용 가능한지 검사하여, 사용 가능하다면 점유하고 있는 자원을 반납하고 요구한 자원을 사용하기 위하여 대기한다.

(4) 환형대기(Circular Wait) 부정

모든 자원에 대하여 일련의 순서대로 고유번호를 부여한다.

각 프로세스는 현재 점유 중인 자원의 고유번호를 기준으로 앞이나 뒤 어느 한 방향으로만 자원을 요구 한다.

2. 교착상태 회피기법

교착상태가 발생하지 않는다는 가능성을 두지 않고, 교착상태가 발생하면 피해나가는 방법이다.

교착상태 회피방법 2가지

① 프로세스의 시작거부

- 프로세스의 요구로 인해 교착상태가 발생할 수 있다면 프로세스를 시작하지 않는다.

② 자원할당 거부

- 프로세스의 요구로 자원을 할당하여 교착상태가 발생할 수 있다면 요청한 자원을 할당하지 않는다.

은행가 알고리즘(Banker's Algorithm)

교착상태 회피기법에 주로 사용되는 알고리즘

데이크스트라 (E. J. Dijkstra)가 제안

은행의 모든 고객의 요구가 충족되도록 현금을 할당하는 것과 동일한 기법이다.

안전상태란 각 프로세스에게 자원을 할당하여 교착상태가 발생하지 않고 완료되는 상태이다.

불안전상태란 교착상태가 발생하는 상태이다.

적용조건으로는 자원의 양과 프로세스의 수가 일정해야 한다.

프로세스의 모든 요구를 유한시간 내에 할당해야 한다.

사용자가 자원의 최대 필요량을 시스템에 미리 알려야 한다.

학습내용2 : 교착상태의 탐지와 회복

교착상태 발생을 예방하지 못했다면 교착상태가 발생했는지 검사하는 알고리즘 필요하다.

교착상태가 발생하였다면 이로부터 회복하기 위한 알고리즘도 필요하다.

1. 교착상태 탐지기법

교착상태가 발생했는지 검사하여 교착상태에 있는 프로세스와 자원을 찾아내는 것

- 교착상태 발견 알고리즘, 자원할당 그래프 사용

쇼사니(Shoshani)와 코프만(Coffman)이 탐지 알고리즘을 제안

2. 교착상태 회복기법

(1) 프로세스의 종료

교착상태의 모든 프로세스를 중지 시킨다.

교착상태의 프로세스를 하나씩 중지 시킨다.

프로세스를 중지 시키는데 있어서 최소비용으로 중지시키기 위한 방법으로 희생자(프로세스)를 선택한다.

- 희생자 선정 시 고려사항
 - 프로세스의 남은 작업 완료시간.
 - 프로세스들의 우선순위
 - 희생시킬 프로세스의 개수
 - 프로세스가 사용하는 자원의 형태와 개수

(2) 자원 선점

교착상태의 프로세스가 점유 중인 자원을 선점하여 다른 프로세스에게 할당한다.

자원을 선점하여 다른 프로세스에게 할당하고 난 후 교착상태의 프로세스를 정지시킨다.

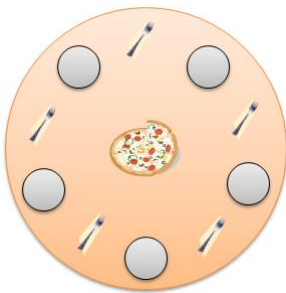
프로세스를 정지시킨 후 재시작 한다.

우선순위가 낮거나, 수행된 경력이 적거나, 사용되는 자원이 적은 프로세스 등을 위주로 해당 프로세스의 자원을 선점한다.

학습내용3 : 기아상태

프로세스가 사용할 수 없는 자원을 계속 기다리면 교착상태의 발생 우려가 있어서, 교착상태를 예방하기 위하여 자원을 할당했을 때 발생하는 기다림의 결과이다.

식사하는 철학자 문제



1. 해결방법

대기중인 프로세스를 발견하고 각 프로세스가 대기한 시간을 조사하고 추적한다.

새로운 기아상태를 발견하면 새로운 작업의 시작을 보류 한다.

- 빈번한 시스템 보류로 인하여 처리량이 감소할 수 있다.

【학습정리】

1. 교착상태의 예방과 회피

교착상태 예방기법

- 상호배제(Mutual Exclusion) 부정
- 점유와 대기(Hold and Wait) 부정
- 비선점(Non-Preemption) 부정
- 환형대기(Circular Wait) 부정

교착상태의 회피기법 (은행가 알고리즘(Banker's Algorithm))

- 프로세스의 시작거부
- 자원할당의 거부

2. 교착상태의 탐지와 회복

교착상태 탐지기법

- 쇼사니(Shoshani)와 코프만(Coffman)의 탐지 알고리즘

교착상태 회복기법

- 프로세스의 종료
- 자원 선점

3. 기아상태

배고픈 철학자