

2023년 2학기 스터디그룹 주간학습보고서

| | | | |
|-------|---------------|------|-------------------|
| 그룹 명 | MaveOS | | |
| 날 짜 | 10월 18일 | 시 간 | 12시 00분 ~ 14시 45분 |
| 스터디장소 | 50주년 기념관 516호 | 회차 | 5 회차 모임 |
| 수강과목 | 운영체제 | 담당교수 | 강지훈 |
| 스터디 | 리더: 한아림 | | |
| 참석자 | 그룹원: 손효림, 이경화 | | |

학습문제

<CPU 스케줄링2>

- 스케줄링 알고리즘
- 스레드 스케줄링

학습문제 해결과정

<CPU 스케줄링>

#1. 스케줄링 알고리즘

1) 라운드 로빈 스케줄링

- 스케줄링 방식 : FCFS에서 선점이 추가된 스케줄링 방식
- 타임 슬라이스 : 실행의 최소 단위 시간 간격(타임 슬라이스)이 프로세스마다 할당됨
- 레디 큐 : 라운드 로빈 스케줄러의 레디 큐는 FIFO 큐로 동작함

2) CPU 버스트 vs 타임 슬라이스

- CPU 버스트 < 타임 슬라이스
- > 프로세스가 CPU를 자발적으로 방출
- > 레디 큐에 있는 다음 프로세스 실행
- CPU 버스트 > 타임 슬라이스
- > CPU 사용 시간을 체크하는 타이머가 작동하여 운영체제에 의해 인터럽트 발생
- > 실행하던 프로세스는 레디 큐의 Tail에 넣어짐
- > 레디 큐에 있는 다음 프로세스 실행

3) 라운드 로빈 스케줄링의 특징

- 일반적으로 연속으로 두 번 이상의 시간 할당을 받는 프로세스는 없음
- 레디 큐에 N개의 프로세스가 있고 타임 슬라이스가 q이면, 각 프로세스는 자신의 다음 타임 슬라이스가 할당될 때까지 $(N-1) \times q$ 시간 이상을 기다리지 않음
- 성능이 타임 슬라이스의 크기에 크게 의존함
- 컨텍스트 스위칭 시간과 비교해 타임 슬라이스가 더 클 것을 원함

4) 프로세스 집합의 총 처리 시간

- 타임 슬라이스의 크기에 좌우됨
- 대부분의 프로세스가 단일 타임 슬라이스 내에 CPU 버스트를 끝낸다면 평균 총 처리 시간이 개선됨

5) 우선순위 스케줄링

- 우선순위 스케줄링이란? 각 프로세스에 우선순위를 부여해 가장 높은 우선순위를 가진 프로세스부터 CPU를 할당하는 방식
- 스케줄링 종류 : 우선순위 스케줄링은 선점형 혹은 비선점형으로 작동할 수 있음
- 주요 문제 : 무기한 Blocking 발생 → 오랫동안 대기하는 프로세스의 우선순위를 점진적으로 증가시키는 '노화' 방법으로 해결

#2. 스레드 스케줄링

1) 사용자 수준 스레드 스케줄링 vs 커널 수준 스레드 스케줄링

- 사용자 수준 스레드 스케줄링 : 스레드 라이브러리가 스케줄링함
- 커널 수준 스레드 스케줄링 : 커널이 스케줄링함
- CPU 상에서 실행되기 위해서는 사용자 수준 스레드가 커널 수준 스레드와 매핑되어야 함

2) 스케줄러 활성화

- 스케줄러 활성화란? 사용자 스레드와 커널 스레드의 매핑을 위해 커널과 스레드 라이브러리가 서로 협력하는 것
- LWP(Light Weight Process) : 사용자와 커널 스레드 사이의 중간 자료 구조
- Scheduler Activation : 사용자 스레드 라이브러리와 커널 스레드 간의 통신 방법 중 하나

3) LWP

- 가상 프로세서 : 사용자 스레드 라이브러리에서 LWP는 가상 프로세서로 인식됨
- 커널 스레드와의 부속 : 각 LWP는 커널 스레드에 부속되어 있으며, 프로세서에서 스케줄링하는 대상이 바로 이 커널 스레드
- Block : 커널 스레드가 block되면 LWP도 같이 block되고, 이에 따라 사용자 수준 스레드도 block됨.

4) 경쟁 범위

- 프로세스 경쟁 범위(PCS)
 - > 다대일이나 다대다 모델의 경우 스레드 라이브러리는 사용자 수준의 스레드를 가용한 LWP에서 스케줄링 함
 - > 동일한 프로세스에 속한 스레드들 사이에서 CPU를 경쟁
 - > 일반적으로 우선순위에 따라 수행됨
- 시스템 경쟁 범위(SCS)
 - > 커널이 CPU 상에서 어느 커널 스레드를 스케줄링할 것인지 결정
 - > SCS 스케줄링에서의 CPU에 대한 경쟁은 시스템상의 모든 스레드 사이에서 일어남
 - > 일대일 모델을 사용한 Windows, Linux 등의 시스템은 오직 SCS만 사용해 스케줄링 함

학습성찰

학습내용 이해도

94 %

학습활동 돌아보기

(좋았던 점, 아쉬운 점
활동모습 사진 추가)

- 한아림

이번 스터디 시간에는 CPU의 다양한 스케줄링 알고리즘에 대한 학습을 진행하였다. 스터디 초반에는 라운드 로빈 스케줄링에 대한 내용을 다루었는데, 이를 통해 레디 큐 및 타임 슬라이스, 그리고 CPU 버스에 대한 개념을 정립할 수 있었다. 이후에는 우선 순위 스케줄링 방식에 대해 공부하였는데, 이를 통해 각 스케줄링 방식의 장점과 특징에 대해 파악할 수 있었다. 특히 우선 순위 스케줄링의 경우 무기한 블로킹 문제가 발생할 수 있다는 확률에 대해서도 배울 수 있었는데, 컴퓨터 세계에서는 생각치도 못한 부분에서 문제가 발생할 수 있다는 것을 다시금 깨달을 수 있었다. 이후 스레드 스케줄링 방식, LWP와 스케줄러 활성화 방식에 대한 내용에 대해 공부하며 CPU와 운영체제에 대한 이해도를 높였다. 이제 스터디 중반으로 진입함에 따라 점점 심오한 내용들이 나오는데, 이렇게 어려운 학습 내용들을 팀원들과 배워가며 끝까지 최선을 다해 스터디에 참여해야겠다고 스스로 결의를 다질 수 있었다.

- 손효림

오늘은 CPU 스케줄링에 대한 내용을 학습했다. CPU 스케줄링의 개념, 기준, 알고리즘과 스레드, 다중 프로세스의 스케줄링과 실시간 CPU 스케줄링 그리고 운영체제의 예와 알고리즘 평가에 대한 내용이다. 수업시간에 진도를 많이 나가지 않아 피피티를 제작하기 조금 힘들었다. 하지만 예습도 상당히 유익했다. 스터디가 학습에 상당한 도움이 되고 있기 때문에 중간고사에서 좋은 점수를 받을 수 있을 것 같은 기분이 든다.

- 이경화

이번 주차에는 지난 주차에 이어 CPU 스케줄링을 다루었다. 스케줄링 알고리즘에 대해서는 지난 주제에서 공부했던 FCFS와 SJF에 이어 라운드 로빈 스케줄링과 우선순위 스케줄링에 대해 공부했는데, FCFS, SJF와의 공통점과 차이점을 비교해가며 공부하니 이해가 더 잘 되었다. 더불어 프로세스 실행의 가장 작은 단위인 스레드의 스케줄링에 대해서도 공부했는데, '스케줄러 활성화'라는 개념이 어렵게 느껴졌다. 수업시간에는 잘 이해하지 못했었는데 스터디 시간을 통해 배운 내용을 정리하다보니 개념이 정리가 되었다. 매주 교안을 만들어 스터디에 활용하고 있는데, 분량을 최대한 간소하게 하면서도 핵심을 포함하도록 교안을 구성하려고 스터디원 전체가 노력하고 있다. 덕분에 중요한 부분에 대해서는 더 확실하게 짚고 넘어가고, 덜 중요한 내용에 대해서는 가볍게 훑고 넘어갈 수 있어서 머릿속에서 지식이 잘 구조화되는 것을 느낀다. 아쉬웠던 점은 딱히 없었다. 지금과 같이 효과적인 스터디 방식을 잘 유지하면 충분히 만족할 수 있는 결과를 얻을 수 있을 것 같다.



다음 학습계획

일정 : 11월 1일 12:00~14:45 (50주년기념관 516호)

