

Lab 1

제목 : 스케줄 시뮬레이터 만들기



과목명 : 운영체제(SW)
교수명 : 최종무 교수님
학 과 : 소프트웨어학과
학 번 : 32183698
이 름 : 이현기
제출일 : 2022 - 4월 08일

1. 목표

이번 Lab1의 목표는 LN3에서 배운 스케줄링 방법을 실제로 구현해보고 결과를 알아보는 것이다. 스케줄링 방법으로는 FCFS, SPN, RR, HRRN, Feedback 방법이 있다.

FCFS(First Come First Serve)는 먼저 오는 프로세스부터 차례대로 처리하는 방법이다. 다른말로 FIFO라고도 한다. SPN(Shortest Process Next)은 짧은 runtime의 프로세스부터 실행하는 것이다. 다른말로 SJF(Shortest Job First)라고도 불린다.

RR(Round Robin)은 프로세스를 완성단계까지 실행하는 것이 아닌 일정 시간(time quantum)만큼만 실행하고 그 후에는 큐에서 다음 프로세스와 교환하는 것이다. time quantum을 작게 하면, 좋은 대응성을 가지지만, 문맥교환이 자주 일어나게 되고, time quantum을 크게 하면, 문맥교환을 작게 하지만, 나쁜 대응성을 가진다. 그래서 이 tradeoff의 밸런스를 맞춰줘야 한다.

HRRN(Highest Response Ratio Next)은 SJF 스케줄링을 보완한 기법으로, 처리시간과 기다리는시간을 고려하여 우선순위를 결정하는데 $\frac{Waiting\ Time + Service\ Time}{Service\ Time}$ 이 식을 통해 우선순위를 결정한다.

Feedback은 MLFQ(Multiple Level Feedback Queue)로 설명할 수 있는데, MLFQ는 여러개의 큐를 이용하여 우선순위를 정하고 제일 위에 있는 큐부터 일정의 time quantum 실행하고 아래 큐로 내려가서 기다리는 것을 말한다. MLFQ를 이용하면 긴 시간이 필요한 일들에게도 CPU를 정당하게 공유할 수 있고 짧은 일들이나 I/O 일들을 빨리 처리할 수 있다. 하지만, 너무 많은 짧은 일들이 있다면, 긴 시간을 필요하는 일들은 CPU 시간을 할당 받을 수 없는 기아상태에 이를 수 있고, 사용자가 스케줄러를 속일 수 있고, 프로그램이 시간이 지날수록 행동을 바꿀 수 있다. 이렇게 다양한 스케줄링 기법들이 있는 것을 알고, 실제로 구현해보는 것이 이번 Lab1의 목표이다.

2. 디자인

우선 스케줄링에 필요한 자료구조 큐를 구현하였고, 큐에서 필요한 enqueue, dequeue, isfull, isempty함수까지 같이 구현하였다. 그리고 FCFS 스케줄링은 구조체로 선언한 프로세스 중에서 arrive_time을 비교하여 가장 작은 arrive_time을 가진 프로세스부터 실행하도록 하였고, 끝나는 시간이 언제인지 출력하게 하였다. RR은 먼저 FCFS에서 한것처럼 가장 작은 arrive_time을 가진 프로세스부터 실행하도록 하였고, quantum=1을 이용하여 1씩 실행하도록 하려 했지만 잘 되지 않았다. SPN은 먼저 arrive time을 비교해 먼저 온 프로세스 중 run time이 가장 짧은 프로세스부터 실행하도록 하였다. 프로세스 이름과 끝난 시간을 출력하도록 구현하였다. 나머지 HRRN, Feedback 스케줄링은 구현하지 못했다.

3. 결과

첫 번째는 LN3 24p에 있는 Workload를 통해 스케줄링을 실행한 결과이다.

```
oslab@oslab:~/Desktop/2022_DKU_OS/lab1_sched$ ./lab1_sched 5
First workload
FCFS
Process: A end_t: 3
Process: B end_t: 9
Process: C end_t: 13
Process: D end_t: 18
Process: E end_t: 20
RR
Process : A end_t : 2
Process : B end_t : 5
Process : C end_t : 3
Process : D end_t : 4
Process : E end_t : 1
SPN
Process: A, end_t: 3
Process: B, end_t: 9
Process: C, end_t: 13
Process: D, end_t: 18
Process: E, end_t: 20
```

두 번째는 임의로 정한 workload에 대한 스케줄링을 실행한 결과이다.

```
Second workload
FCFS
Process: 1 end_t: 2
Process: 2 end_t: 5
Process: 3 end_t: 10
Process: 4 end_t: 14
Process: 5 end_t: 20
RR
Process : 1 end_t : 1
Process : 2 end_t : 2
Process : 3 end_t : 4
Process : 4 end_t : 3
Process : 5 end_t : 5
SPN
Process: 1, end_t: 2
Process: 2, end_t: 5
Process: 3, end_t: 10
Process: 4, end_t: 14
Process: 5, end_t: 20
```

4. 어려웠던 점 및 느낀점

강의를 통해 여러 가지 스케줄링 방법들에 대해 학습하였으나, 이것을 코드로 구현하는 과정이 상당히 어려웠다. 학습하였을 때에는 이해가 잘 되서 금방금방 구현할 수 있을줄 알았는데 막상 코드를 짜려고 하니 어떻게 짜야할지 많이 막막했다. 시간을 들여서 혼자 FCFS 스케줄링과 RR, SPN 스케줄링을 구현해보았지만, 위에 나온 결과를 보면 알 수 있듯이 제대로 된 결과도 출력되지 않았다. RR 스케줄링은 time quantum인 1씩 잘라서 프로세스를 실행하는 부분이 되지 않았고, SPN도 arrive time을 비교하고 run time을 비교하지 않고 FCFS 스케줄링처럼 출력이 되었다. 나머지 HRRN, Feedback 스케줄링은 큐를 이용하여 코드를 짜야하는데 어떻게 이용해야하는지 잘 모르겠다. 솔직히 과제를 늦게 시작한 것도 있지만, 시간이 매우 많이 필요한 과제였다. 다음 lab과제는 이번보다 좀 더 여유를 두고 과제를 시작해야겠다고 느꼈고, 코딩하는 능력을 많이 키워야겠다고 느꼈다.