

CPU scheduling의 또 다른 예

- 시스템 내에 CPU는 하나, IO 장치는 무한이라고 가정
- 각 스레드가 처음에 시작되어 대기 큐에 들어가는 시각 즉 arrival time을 고려 (수업 시간에 한 것처럼...)
- 각 스레드가 CPU - IO - CPU - IO ... 작업을 번갈아 한다고 가정
- CPU 작업을 끝낸 스레드는 곧바로 IO 작업을 시작함
- SJF 알고리즘에서 tie일 경우, 대기 큐에 먼저 들어 온 스레드가 우선권 가짐.
- SRTF 알고리즘에서 현재 CPU를 사용 중인 스레드와 대기큐에 있는 다른 스레드가 tie일 경우, 현재 CPU를 사용 중인 스레드에게 우선권 부여
- 어떤 동일한 시점에 대기 큐에 들어가는 스레드가 다수일 경우, 대기큐에 들어가는 순서에 대한 규칙 (우선순위 문제)
 1. 처음으로 큐에 들어가는 스레드
 2. IO를 마친 스레드
 3. Time slice를 다 소진한 스레드
 4. 위 순서대로 조건을 고려해도 tie일 경우, 스레드 번호가 적은 것이 큐에 먼저 들어감

FCFS 알고리즘

Arrival Time

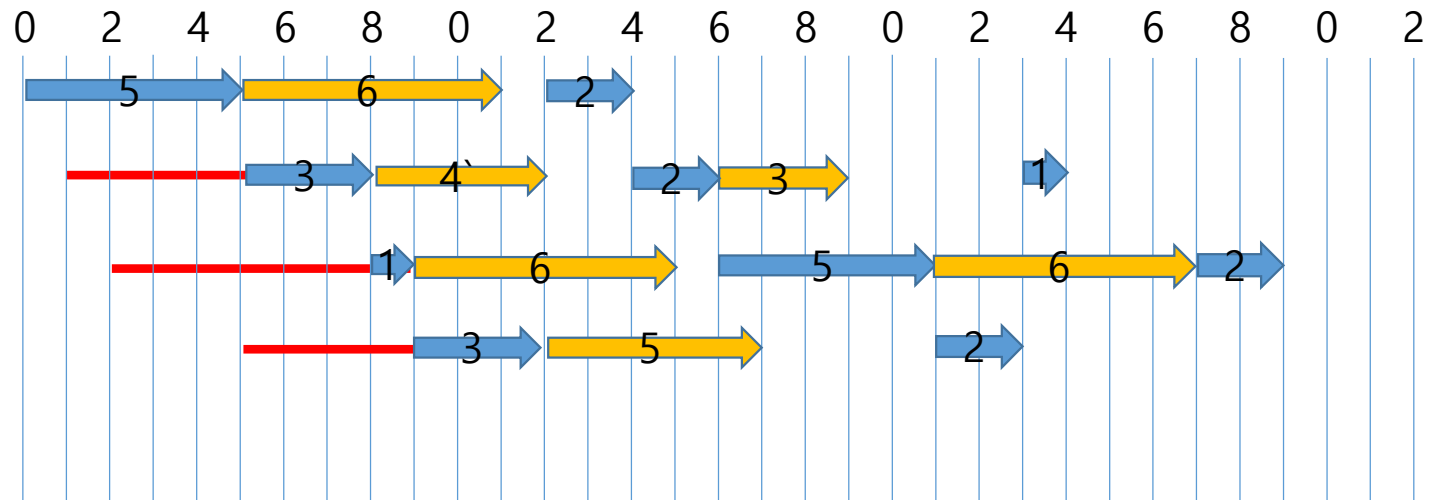


T1 : 0 5-(6)-2

T2 : 1 3-(4)-2-(3)-1

T3 : 2 1-(6)-5-(6)-2

T4 : 5 3-(5)-2



Waiting Time:

T1 : 1
T2 : 10
T3 : 7
T4 : 8

Turnaround Time:

T1 : 14
T2 : 23
T3 : 27
T4 : 18

SJF 알고리즘

Arrival Time

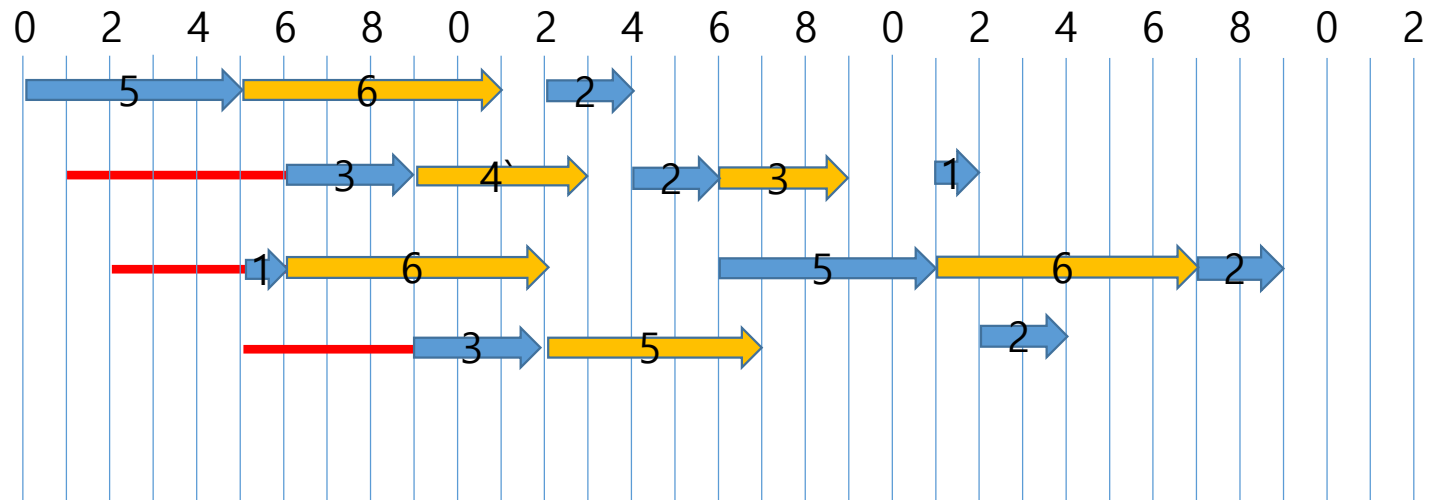


T1 : 0 5-(6)-2

T2 : 1 3-(4)-2-(3)-1

T3 : 2 1-(6)-5-(6)-2

T4 : 5 3-(5)-2



Waiting Time:

T1 : 1

T2 : 8

T3 : 7

T4 : 9

SRTF 알고리즘

이 시점에서 남은 시간 측면에서
T1과 T3가 tie이지만 현재 CPU를 쓰는
스레드에게 우선권 부여

Arrival Time

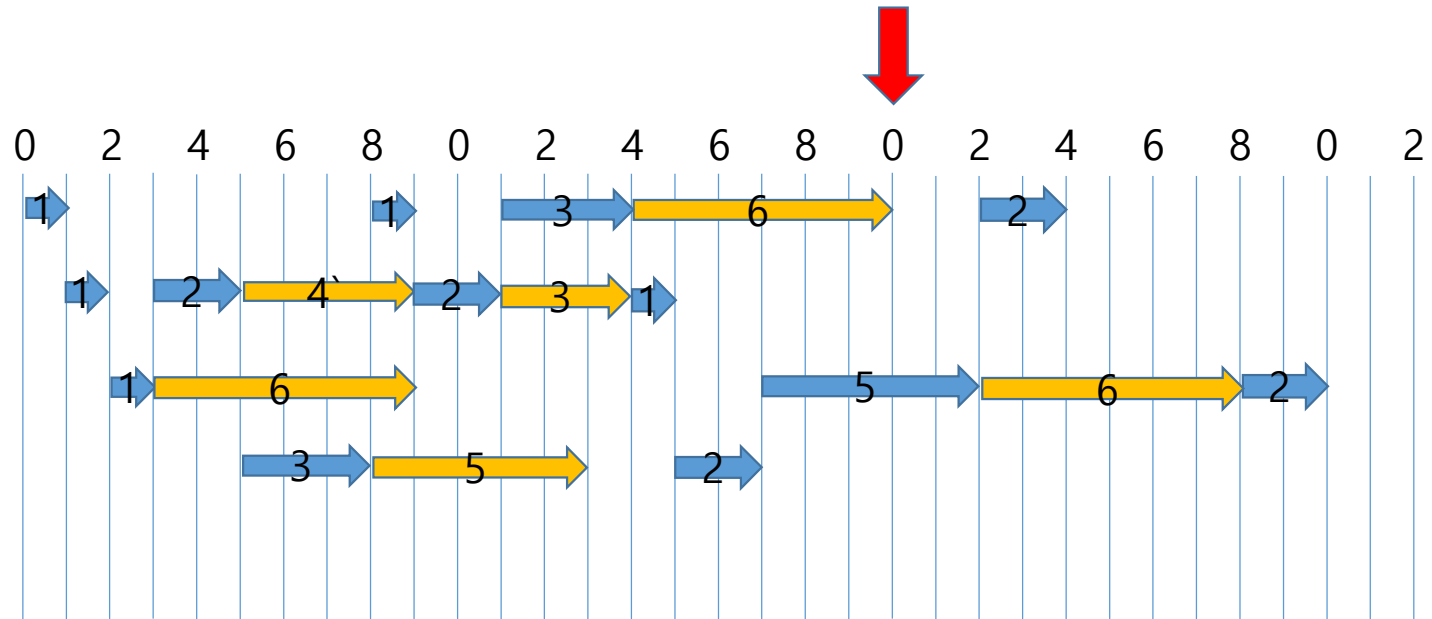


T1 : 0 5-(6)-2

T2 : 1 3-(4)-2-(3)-1

T3 : 2 1-(6)-5-(6)-2

T4 : 5 3-(5)-2



Waiting Time:

T1 : 11

T2 : 1

T3 : 8

T4 : 2

RR 알고리즘 (time slice = 2)

Arrival Time



T1 : 0 5-(6)-2

T2 : 1 3-(4)-2-(3)-1

T3 : 2 1-(6)-5-(6)-2

T4 : 5 3-(5)-2

