

CPU Scheduling

CS 06.

CPU Scheduling 정의

프로세스가 생성되어 실행될 때
필요한 시스템의 여러 자원을
해당 프로세스에게 할당하는 작업을 의미

CPU Scheduling 목적

- 공정한 스케줄링
- 처리량의 극대화
- 응답시간 최소화
- 균형있는 자원 사용
- 실행의 무한 연기 배제

CPU Scheduling Criteria (성능 척도)

- **프로세서 이용률 (CPU Utilization)**

- 시간당 CPU를 사용한 시간의 비율
- 프로세서를 실행상태로 항상 유지하여 유휴상태가 되지 않도록 한다.
가능하면 입출력(I/O) 중심의 작업보다 프로세서 중심의 작업을 실행해야한다.

- **처리율 (Throughput)**

- 시간당 처리한 작업의 비율
- 단위 시간당 완료되는 작업 수가 많도록 짧은 작업을 우선 처리하거나 인터럽트 없이 작업을 실행한다.

- **반환시간 또는 소요시간 (Turnaround Time)**

- CPU burst time(쓰고 나갈때까지의 시간, 누적되지 않음)
- 작업이 시스템에 맡겨져서 메인 메모리에 들어가기까지의 시간, 준비 큐에 있는 시간, 실행시간, 입출력시간 등 작업 제출 후 완료되는 순간까지의 소요시간이 최소화되도록 일괄 처리 작업을 우선 처리한다.

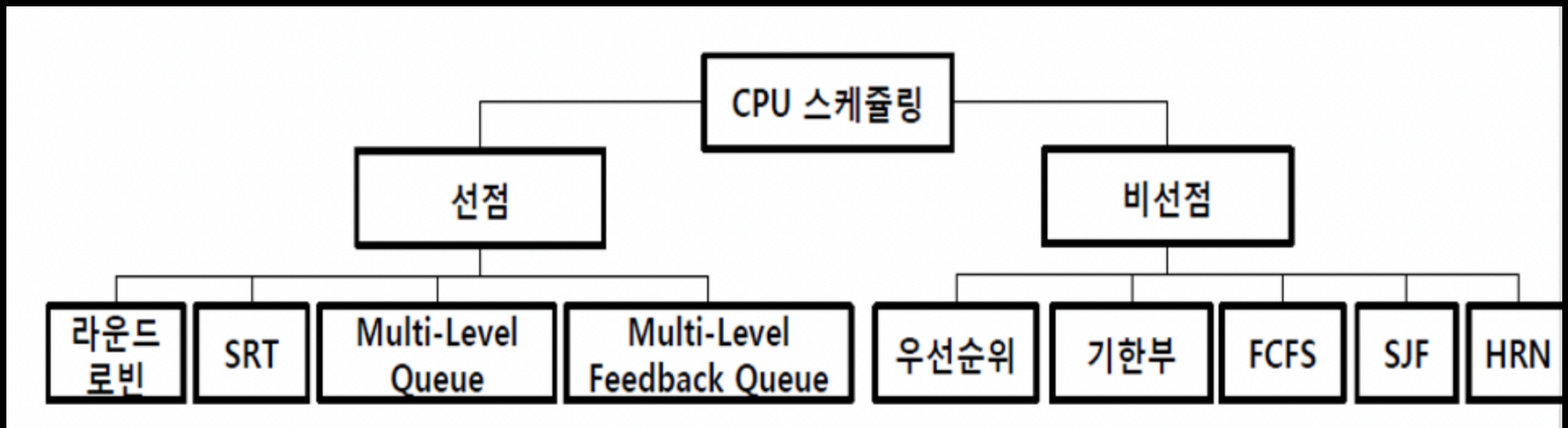
- **대기시간 (Waiting Time)**

- 대기열에 들어와 CPU를 할당받기까지 기다린 시간
- 작업의 실행시간이나 입출력시간에는 실제적인 영향을 미치지 못하므로 준비 큐에서 기다리는 시간이 최소화되도록 사용자 수를 제한한다.

- **반응시간 또는 응답시간 (Response Time)**

- 대기열에서 처음으로 CPU를 얻을때까지 걸린시간
- 반응시간은 의뢰한 시간에서부터 반응이 시작되는 시간까지의 간격으로 대화형 시스템에서 중요한 사항이다.
따라서 대화식 작업을 우선 처리하고 일괄 처리 작업은 대화식 작업의 요구가 없을때까지 처리한다.

CPU Scheduling의 종류



CPU Scheduling의 종류

구분	선점 (Preemptive)	비선점 (Non-Preemptive)
개념	P1이 CPU점유, P2가 CPU점유가능	P1이 CPU점유, P2는 대기
장점	빠른 응답, 시분할 시스템에 적합	응답시간 예상 가능, 공정한 처리
단점	오버헤드 발생 (Context Switching)	짧은 작업도 긴 대기 발생

비선점형 방식(non-preemptive) 알고리즘 종류

- **FCFS (First Come First Service)**

- 도착한 순서대로 처리 (일괄처리)
- 짧은 작업이 긴 작업을 기다리게 됨
- Convoy effect 발생

- **SJF (Shortest Job First)**

- 가장 짧은 소요시간의 프로세스 먼저 실행
- FCFS 보다 평균 대기 시간 감소
- 긴 프로세스가 짧은 프로세스에게 계속 작업 순위가 밀리게 되어 실행되지 못하는 기아현상(Starvation)발생

- **우선순위 (Priority)**

- 프로세스에 우선순위 부여하여 우선순위에 따라 할당
- SJF 단점 보완

선점형 방식(preemptive) 알고리즘 종류

- **라운드 로빈 (RR, Round Robin)**

- 동일한 할당 시간을 주고 그 시간 안에 끝나지 않으면 다시 준비큐로 이동
- FCFS 알고리즘을 선점 형태로 변형한 기법

- **SRF (Shortest Remaining Time First)**

- SJF의 선점형 스케줄링 방식
- 남은 프로세스의 burst time보다 더 짧은 process가 도착하면 CPU를 빼앗음
- 프로세스가 새로 들어올때마다 갱신됨

- **다단계 큐 (MLQ, Multi Level Queue)**

- 우선 순위에 다른 준비 큐 여러 개 사용
- 스케줄링 부담은 적지만, 유연성이 떨어짐