

Scheduling Algorithms

- **FCFS (First-Come First-Served)**
- **SJF (Shortest-Job-First)**
- **SRTF (Shortest-Remaining-Time-First)**
- **Priority Scheduling**
- **RR (Round Robin)**
- **Multilevel Queue**
- **Multilevel Feedback Queue**

FCFS (First-Come First-Served)

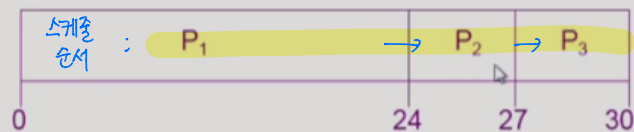
⇒ 온 순서대로 실행!

→ Example:

Process	Burst Time
P_1	24
P_2	3
P_3	3

→ 프로세스의 도착 순서 $P_1 \rightarrow P_2 \rightarrow P_3$

스케줄 순서를 Gantt Chart로 나타내면 다음과 같다



→ Waiting time for $P_1 = 0$; $P_2 = 24$; $P_3 = 27$

→ Average waiting time: $(0 + 24 + 27)/3 = 17$

시간 많이 걸리는 프로세스는 많이 기다림. (여러번 기다림)
 시간 적게 걸리는 프로세스는 적게 기다림.

Round Robin (RR) ⇒ 공정한 CPU 스케줄링.

⇒ 할당된 시간만큼 CPU 쓰고 다음 프로세스에게 CPU 전달.
 ⇒ 시간이 많이 걸리는 프로세스는 time quantum 만큼 CPU 쓰고 뺏기고
 그걸 기다리다가 자기차례 되면 또 할당시간만큼 CPU 쓰고 뺏긴다!

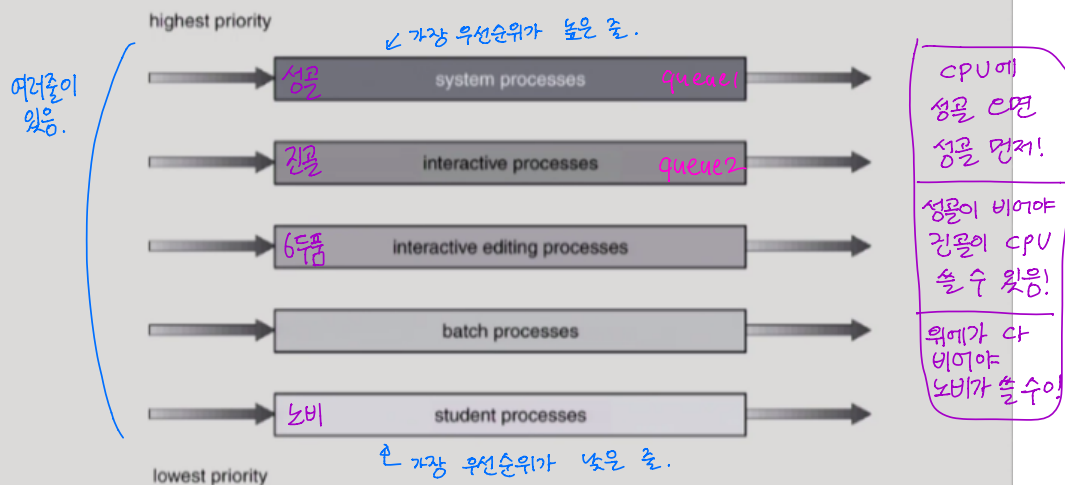
- ➔ 각 프로세스는 동일한 크기의 할당 시간(**time quantum**)을 가짐
(일반적으로 10-100 milliseconds).
- ➔ 할당 시간이 지나면 프로세스는 선점(**preempted**)당하고 **ready queue**
의 제일 뒤에 가서 다시 줄을 선다
- ➔ n 개의 프로세스가 ready queue에 있고 할당 시간이 **q time unit**인 경
우 각 프로세스는 최대 **q time unit** 단위로 CPU 시간의 $1/n$ 을 얻는다.
⇒ 어떤 프로세스도 **$(n-1)q$ time unit** 이상 기다리지 않는다.
- ➔ Performance
 - ✓ q large \Rightarrow FCFS
 - ✓ q small \Rightarrow context switch 오버헤드가 커진다

Multilevel Queue

⇒ 계급제..

⇒ 태어난 계급 극복X

⇒ 성물은 성물줄에 줄선다.
(노비) (노비줄)

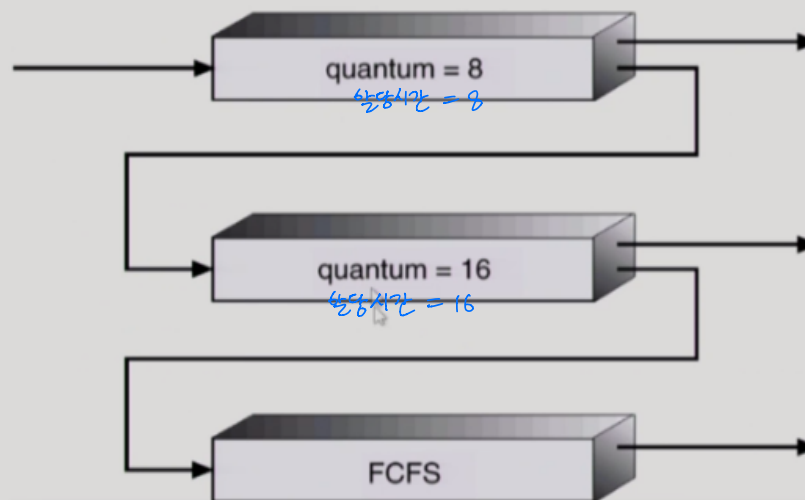


Multilevel Queue

- Ready queue를 여러 개로 분할 ⇒ ready queue를 여러 줄로 분할 ⇒ multilevel queue!
- ✓ **foreground (interactive)** ⇒ 사람이!
- ✓ **background (batch – no human interaction)** ⇒ 사람 교류 없어 only CPU만!
- ✱ 각 큐는 독립적인 스케줄링 알고리즘을 가짐
- ✓ **foreground – RR**
- ✓ **background – FCFS** (context switch overhead를 줄임.)
- 큐에 대한 스케줄링이 필요
- ✓ **Fixed priority scheduling** ⇒ 우선순위 높으면만 CPU 준다!
 - serve **all from foreground then from background.**
 - Possibility of **starvation.** (성공 제라하고 전부 starvation의 가능성이 있음.)
- ✓ **Time slice** ← starvation을 막기위해.
 - 각 큐에 CPU time을 적절한 비율로 할당
 - Eg., **80% to foreground in RR, 20% to background in FCFS**
 - 40% ⇒ 우선순위가 높은 줄에 주고 } ⇒ starvation을 막을 수 있음.
 - 20% ⇒ 우선순위가 낮은 줄에 줌 }

Multilevel Feedback Queue

- ⇒ 실행 가능.
- ⇒ 시간 짧게 걸리는 process가 우선!



Multilevel Feedback Queue

- 프로세스가 다른 큐로 이동 가능
- 에이징(aging)을 이와 같은 방식으로 구현할 수 있다
- Multilevel-feedback-queue scheduler를 정의하는 파라미터들
 - ✓ Queue의 수
 - ✓ 각 큐의 scheduling algorithm
 - ✓ Process를 상위 큐로 보내는 기준
 - ✓ Process를 하위 큐로 내쫓는 기준
 - ✓ 프로세스가 CPU 서비스를 받으려 할 때 들어갈 큐를 결정하는 기준

Example of Multilevel Feedback Queue

- Three queues:
 - ✓ Q_0 – time quantum 8 milliseconds
 - ✓ Q_1 – time quantum 16 milliseconds
 - ✓ Q_2 – FCFS
- Scheduling
 - ✓ new job이 queue Q_0 로 들어감
 - ✓ CPU를 잡아서 할당 시간 8 milliseconds 동안 수행됨
 - ✓ 8 milliseconds 동안 다 끝내지 못했으면 queue Q_1 으로 내려감
 - ✓ Q_1 에 줄서서 기다렸다가 CPU를 잡아서 16 ms 동안 수행됨
 - ✓ 16 ms에 끝내지 못한 경우 queue Q_2 로 쫓겨남