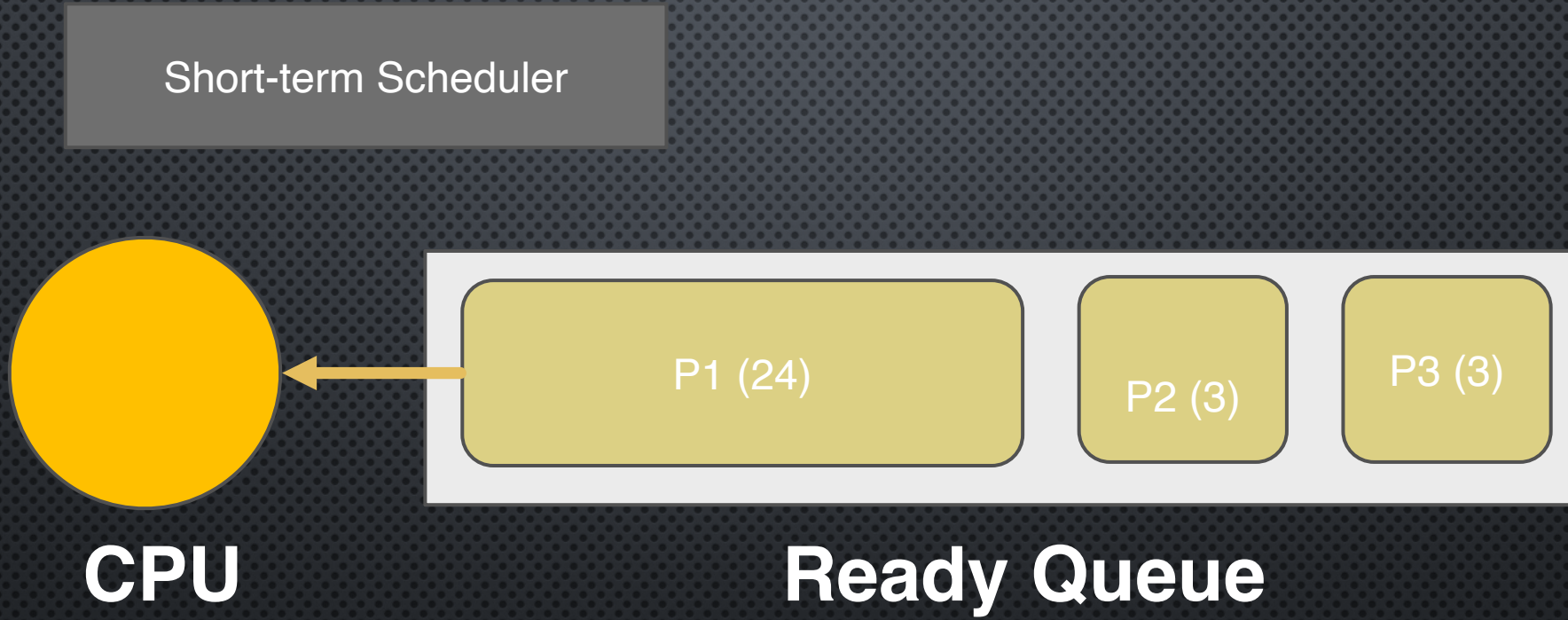
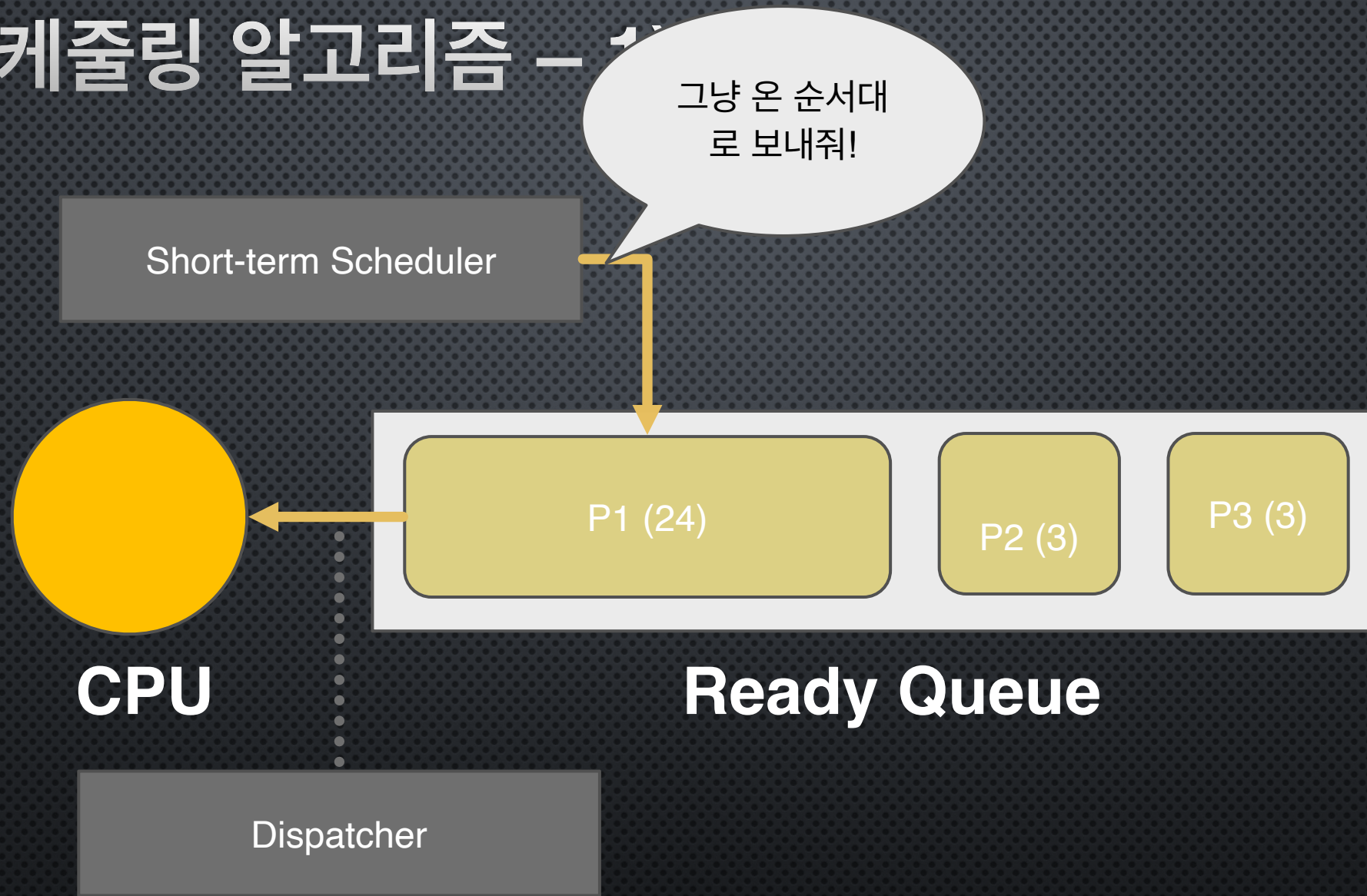


CPU 스케줄링 알고리즘

CPU 스케줄링 알고리즘 – 1) FCFS



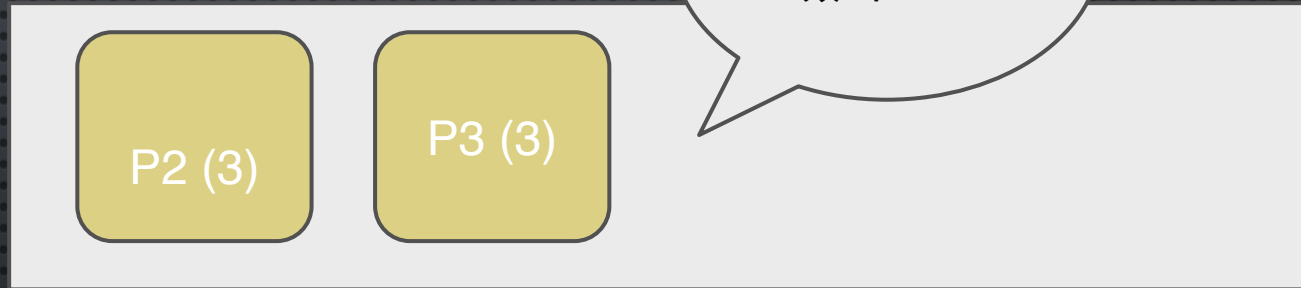
CPU 스케줄링 알고리즘 - 1



CPU 스케줄링 알고리즘 – 1) FCFS



CPU



Ready Queue

$$\text{Avg. Waiting time} = (0+24+27)/3 = 17$$

CPU 스케줄링 알고리즘 – 1) FCFS

FCFS 한줄평

CPU 스케줄링 알고리즘 – 1) FCFS

FCFS 한줄평

- 형평성: ★★★★★

CPU 스케줄링 알고리즘 – 1) FCFS

FCFS 한줄평

- 형평성: ★★★★★
- 효율성: ★★☆☆☆

CPU 스케줄링 알고리즘 – 1) FCFS

FCFS 한줄평

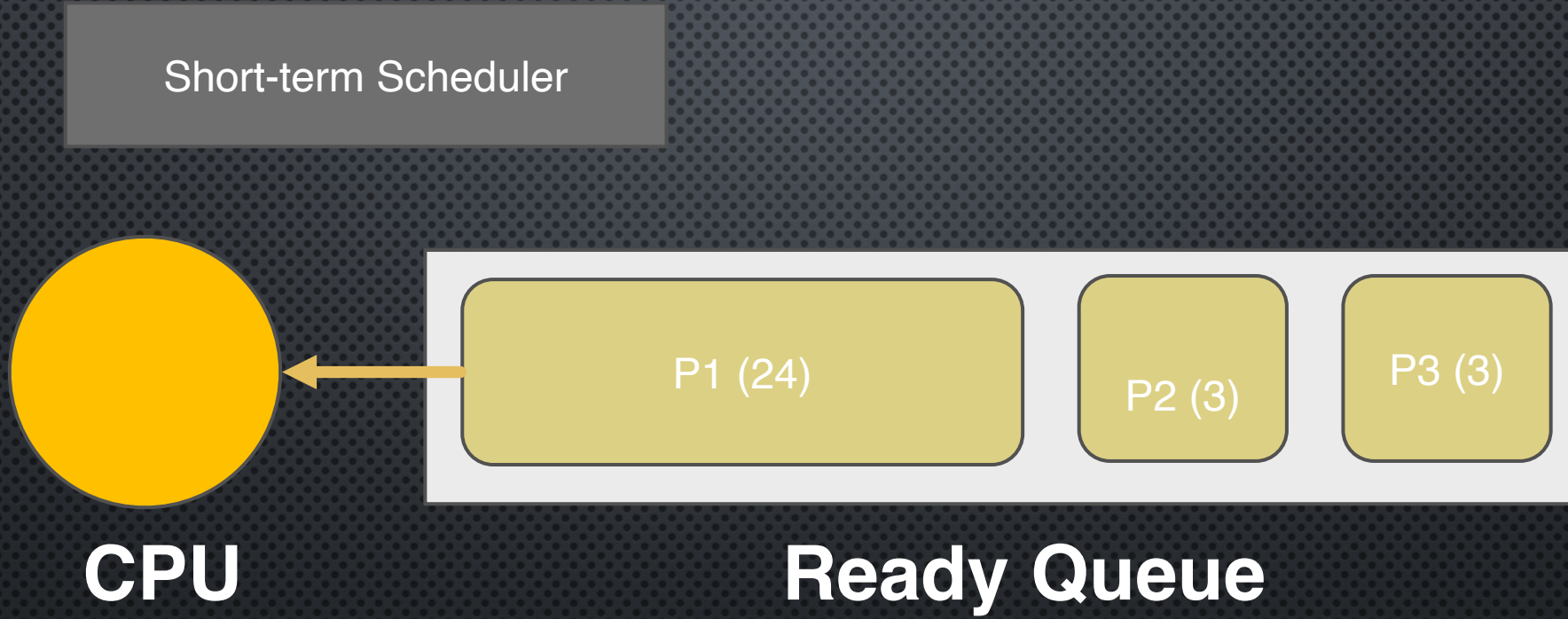
- 형평성: ★★★★★
- 효율성: ★★☆☆☆
- 난이도: ★☆☆☆☆

CPU 스케줄링 알고리즘 – 1) FCFS

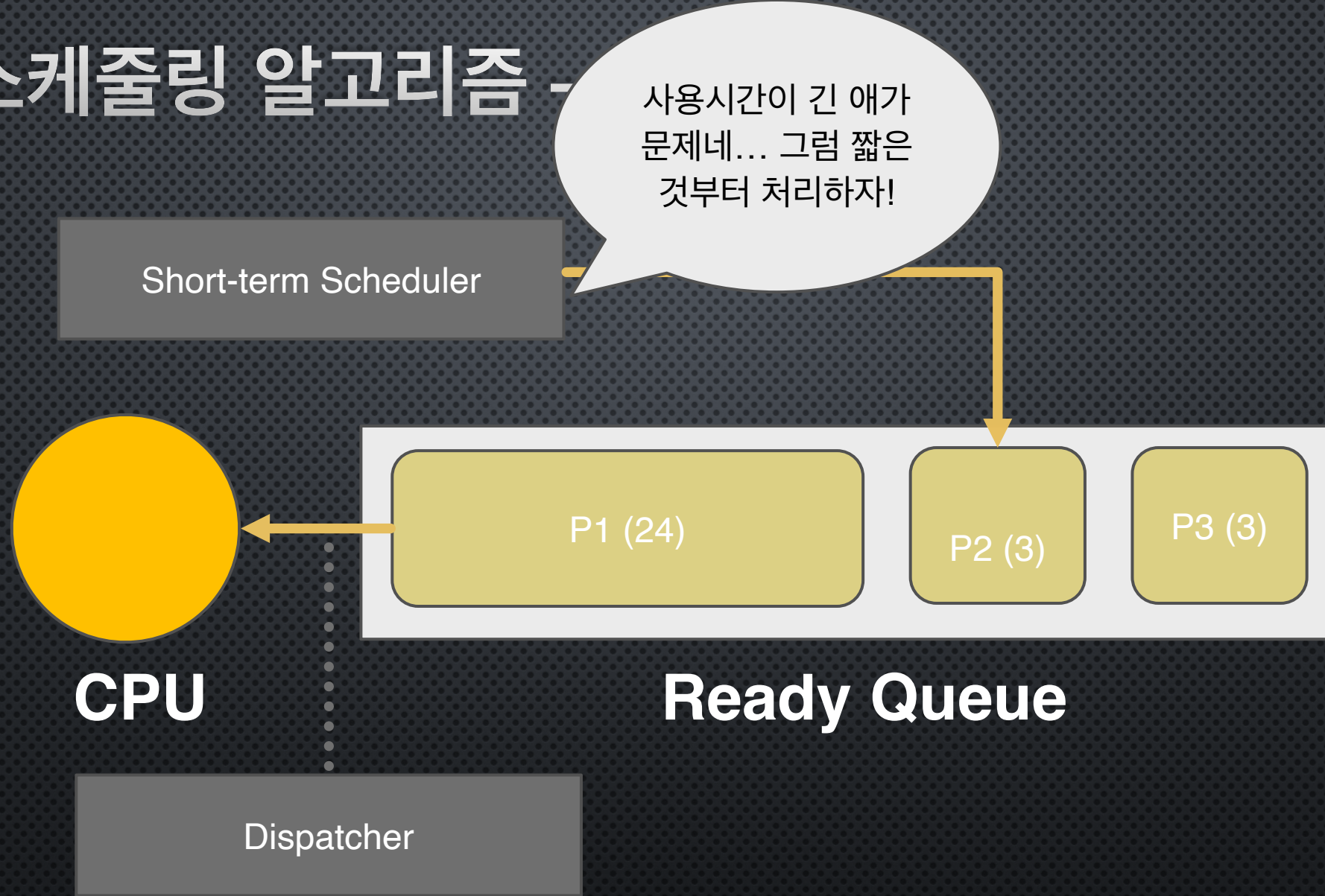
FCFS 한줄평

- 형평성: ★★★★★
 - 효율성: ★★☆☆☆
 - 난이도: ★☆☆☆☆
-
- 상대적으로 CPU burst가 짧은 Interactive job이 많다면 어느 정도 괜찮겠지만.. 글썄?
 - 우리가 스케줄링을 설계한다면 가장 먼저 떠오를 방법 == 현실 세계의 줄서기 문화
 - 긴 CPU burst time의 프로세스로 인한 Convoy effect 발생

CPU 스케줄링 알고리즘 – 2) SJF



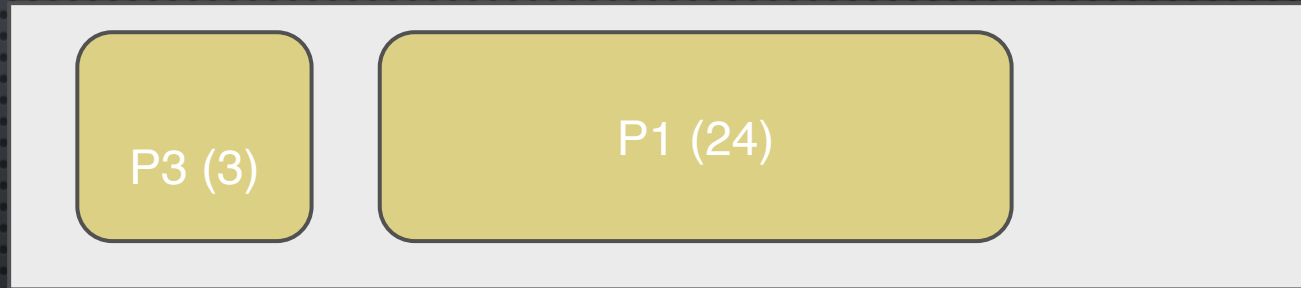
CPU 스케줄링 알고리즘



CPU 스케줄링 알고리즘 – 2) SJF



CPU



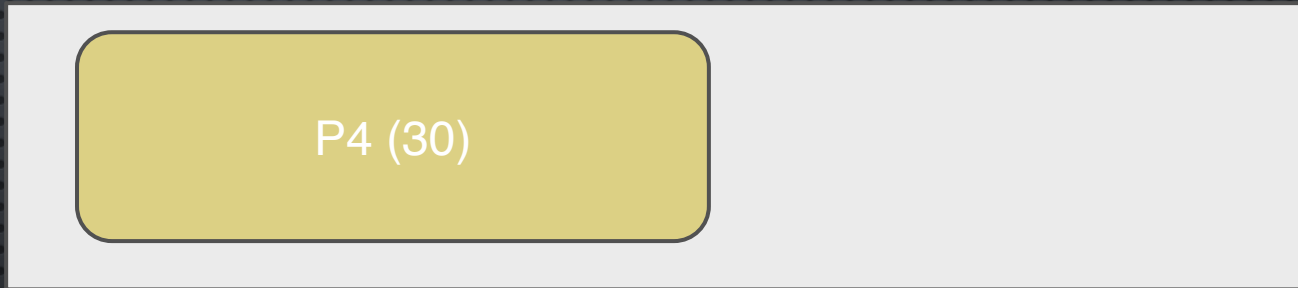
Ready Queue

$$\text{Avg. Waiting time} = (0+3+6)/3 = 3$$

CPU 스케줄링 알고리즘 – 2-1) SRTF



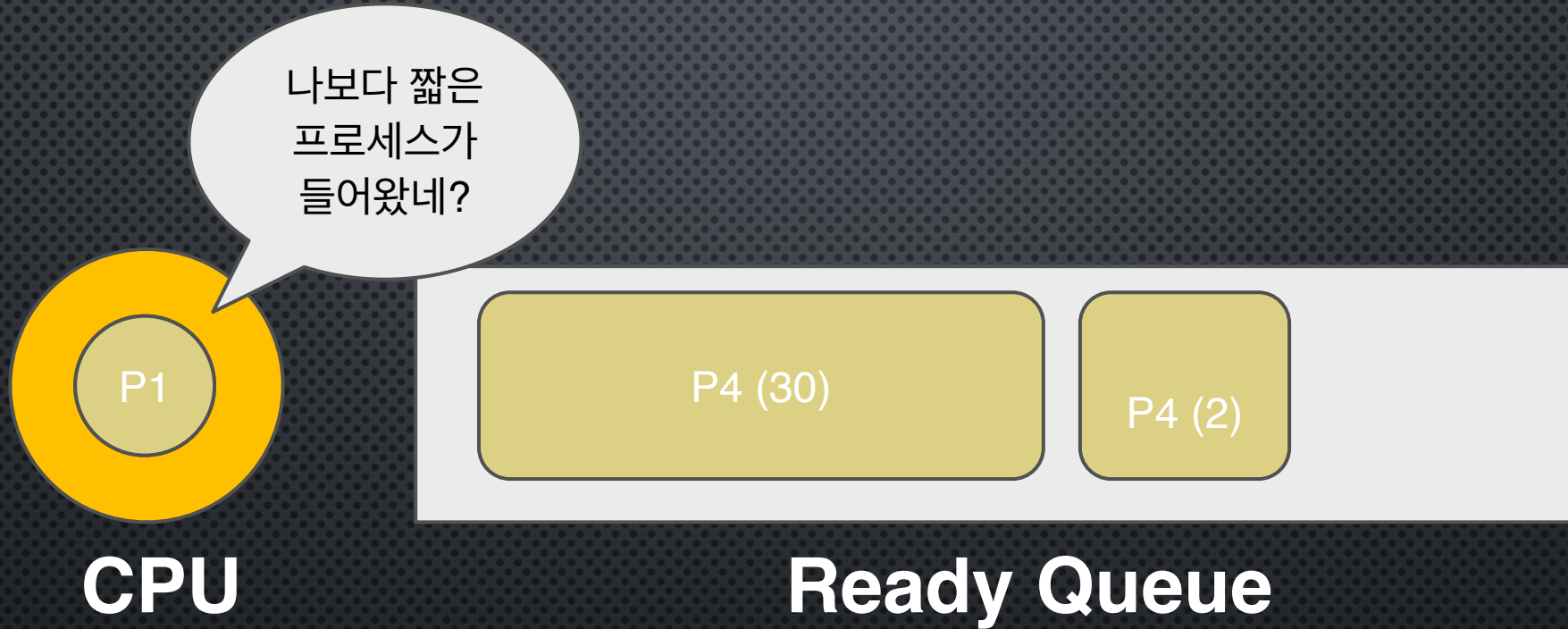
CPU



Ready Queue

현재 Ready Queue에 존재하는 프로세스 중 가장
CPU burst time이 짧은 프로세스부터 **선점적으로** 실행시키는 방법

CPU 스케줄링 알고리즘 – 2-1) SRTF

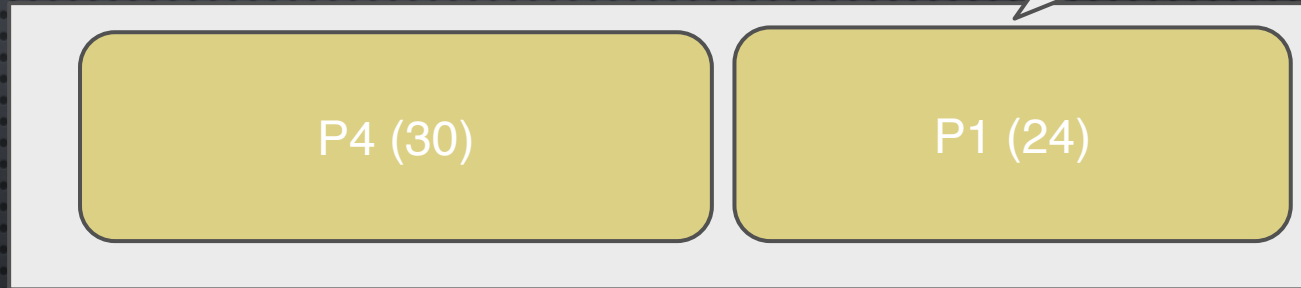


현재 Ready Queue에 존재하는 프로세스 중 가장 CPU burst time이 짧은 프로세스부터 **선점적으로** 실행시키는 방법

CPU 스케줄링 알고리즘 – 2-1) SRTF



CPU



Ready Queue

불공평한 세
상..

현재 Ready Queue에 존재하는 프로세스 중 가장
CPU burst time이 짧은 프로세스부터 **선점적으로** 실행시키는 방법

CPU 스케줄링 알고리즘 – 2) SJF

SJF(SRTF) 한줄평

CPU 스케줄링 알고리즘 – 2) SJF

SJF(SRTF) 한줄평

- 형평성: ★☆☆☆☆

CPU 스케줄링 알고리즘 – 2) SJF

SJF(SRTF) 한줄평

- 형평성: ★☆☆☆☆
- 효율성: ★★★★★

CPU 스케줄링 알고리즘 – 2) SJF

SJF(SRTF) 한줄평

- 형평성: ★☆☆☆☆
- 효율성: ★★★★★
- 난이도: ★★★★★

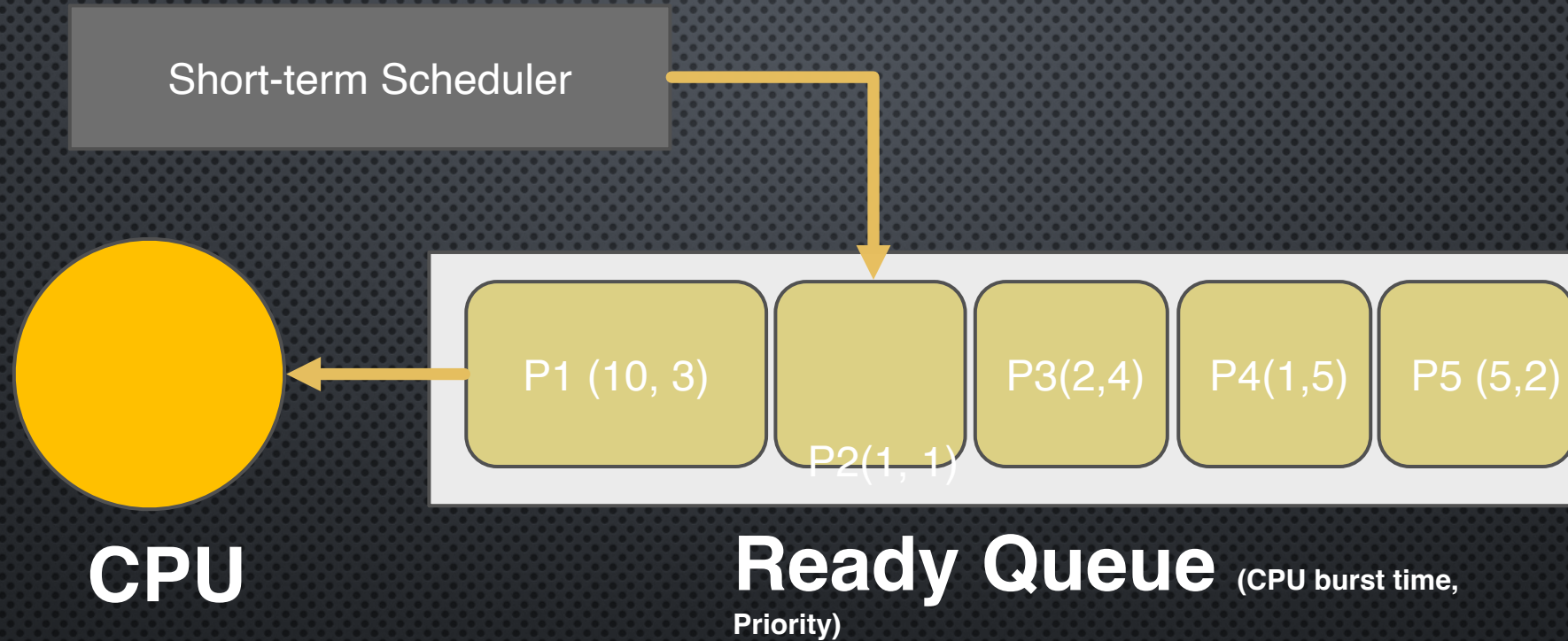
CPU 스케줄링 알고리즘 – 2) SJF

SJF(SRTF) 한줄평

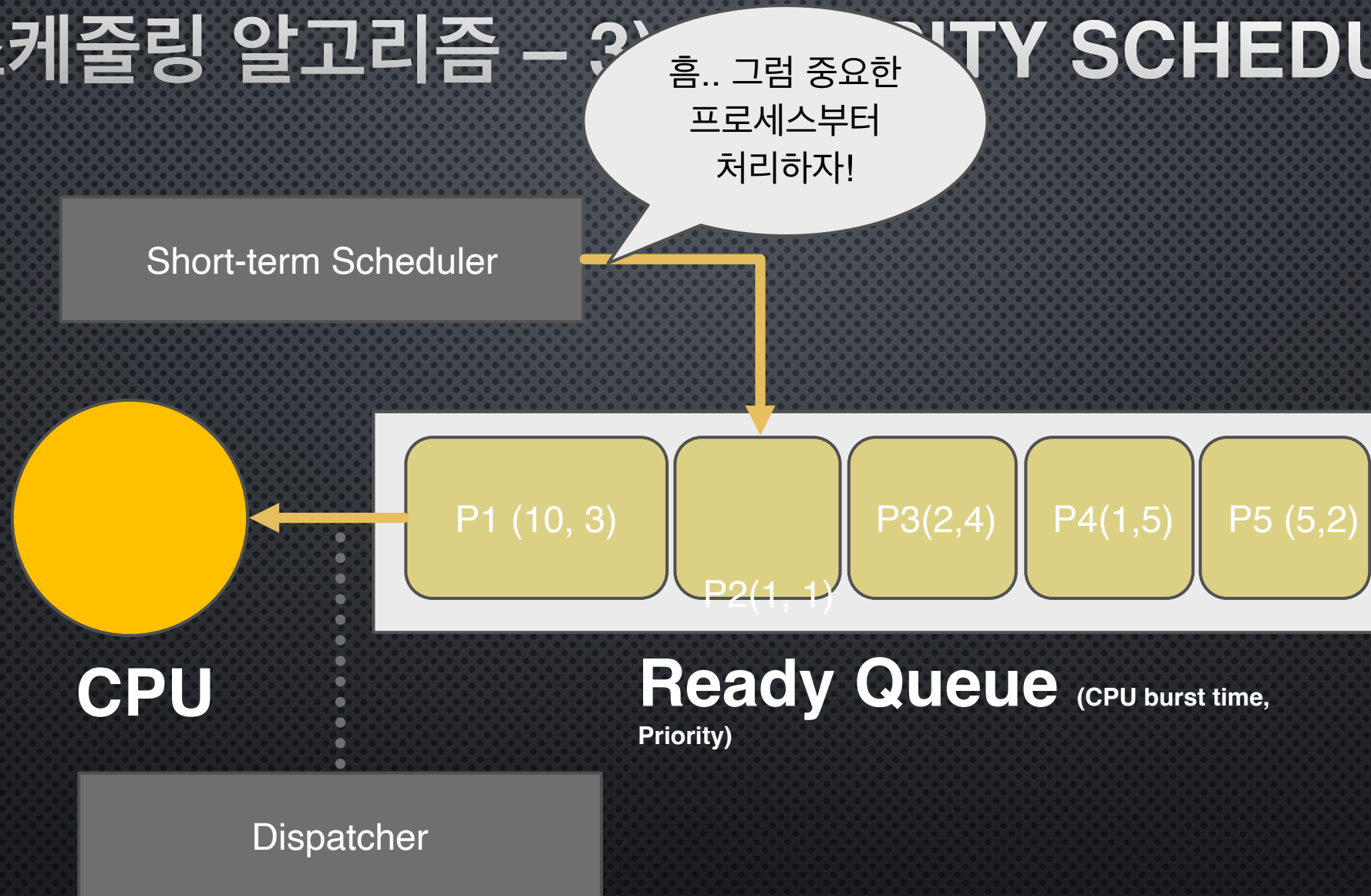
- 형평성: ★☆☆☆☆
- 효율성: ★★★★★
- 난이도: ★★★★★

- Avg. waiting time을 중요한 스케줄링 척도로 생각했을 때 가장 최고 성능을 내는 스케줄링
- SRTF의 경우 min. Average waiting time을 보장한다.
- 하지만 긴 CPU burst time의 프로세스는 계속 실행되지 못하고 starvation에 빠지는 문제가 생김.
- 추가적으로, ready queue에 들어온 프로세스의 CPU burst time을 알 수 없으므로 구현이 불가능함.

CPU 스케줄링 알고리즘 – 3) PRIORITY SCHEDULING



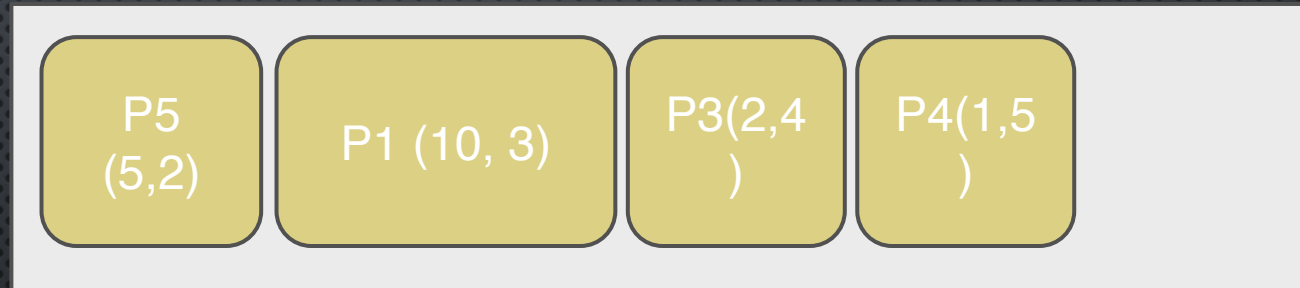
CPU 스케줄링 알고리즘 – 3) PRIORITY SCHEDULING



SCHEDULING



CPU



Priority Queue(heap)

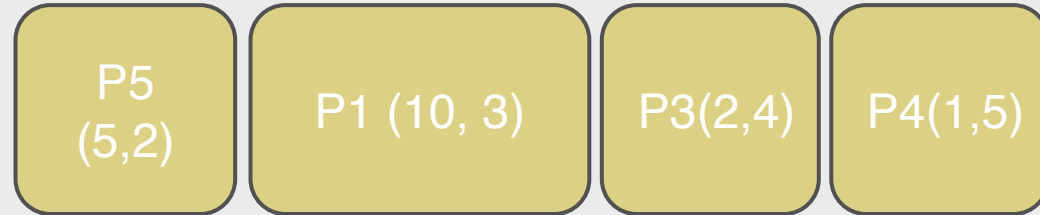
**Avg. Waiting time = $(0+1+6+16+18)/5 = 8.2$
(SJF의 경우 3.4)**

SCHEDULING



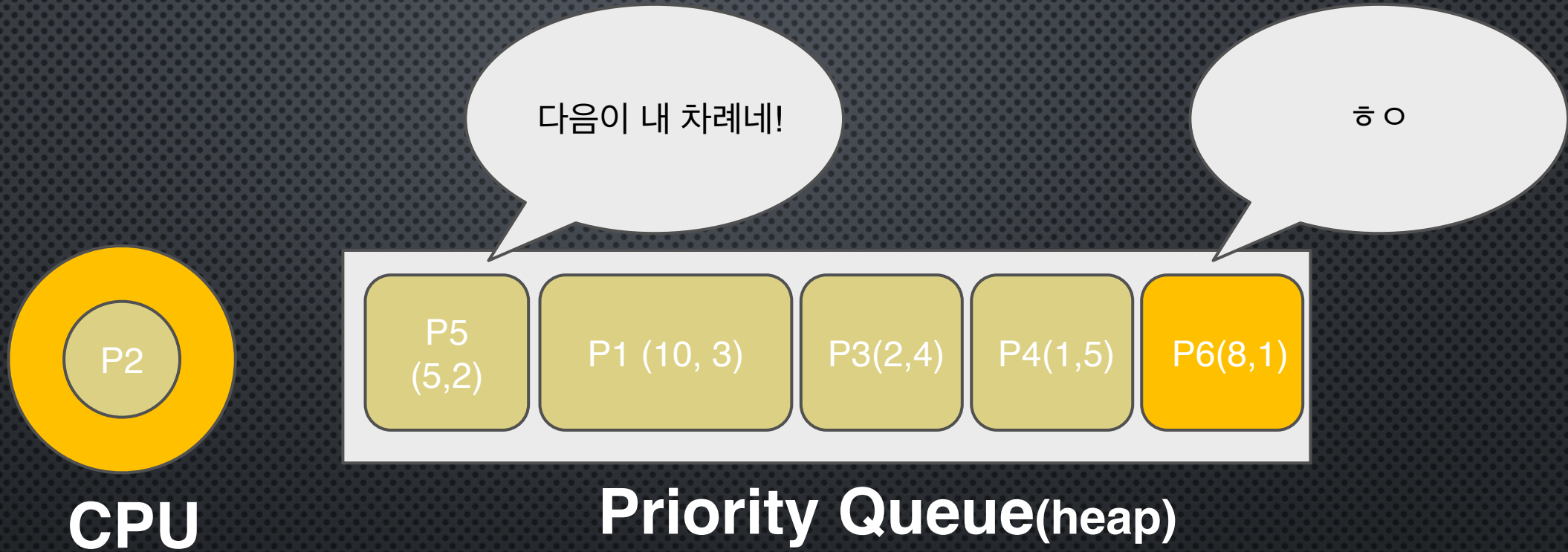
CPU

다음이 내 차례네!



Priority Queue(heap)

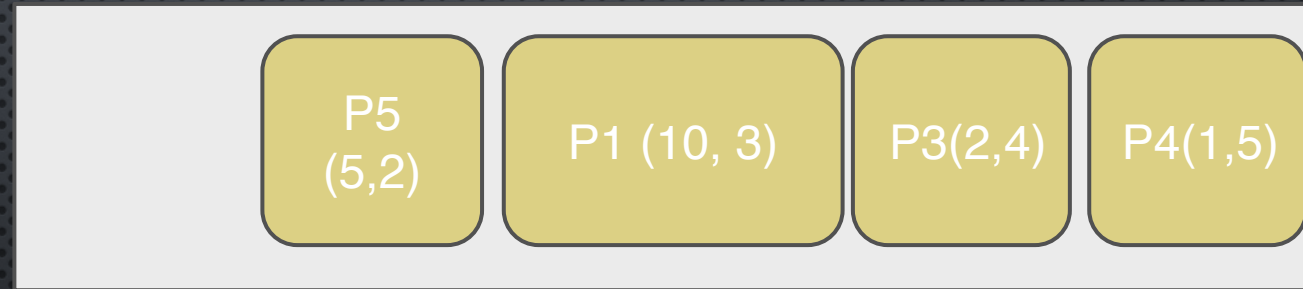
SCHEDULING



SCHEDULING

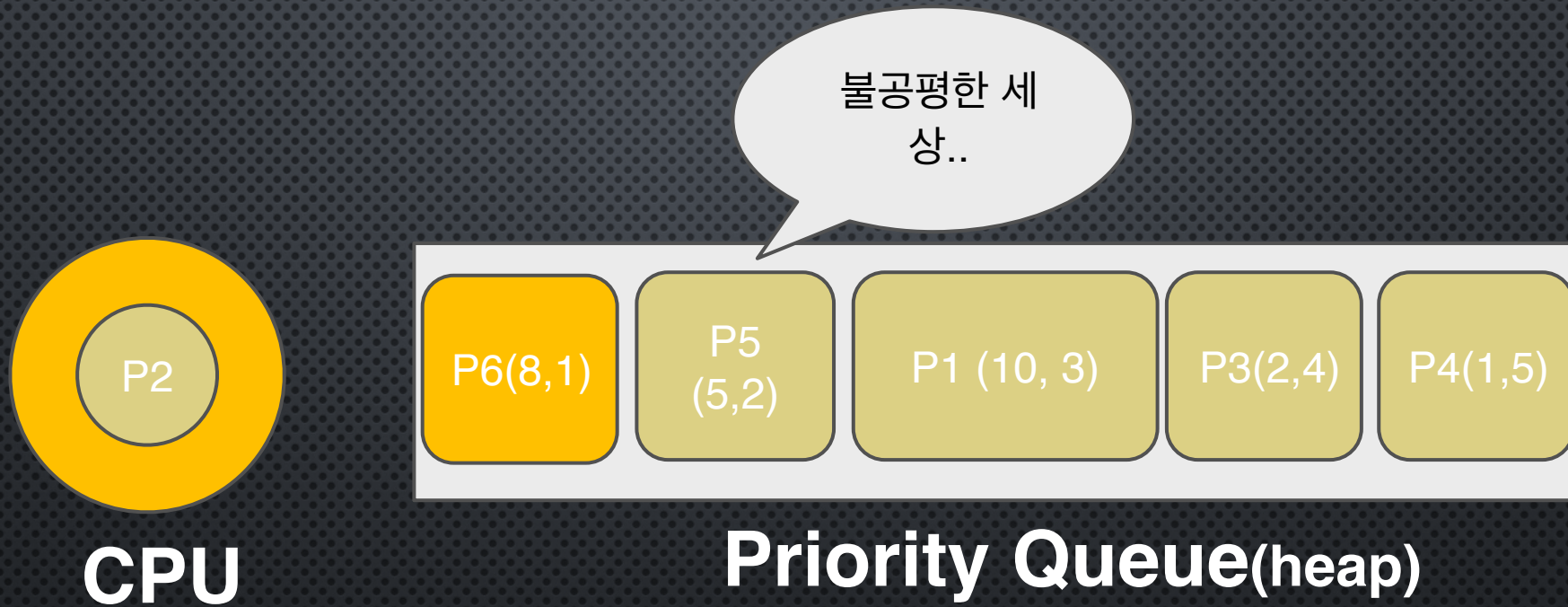


CPU



Priority Queue(heap)

SCHEDULING



위와 같은 비선점형 뿐만 아니라, SRTF와 같이 선점형으로도 우선순위 스케줄링을 구현할 수 있다.

SCHEDULING

Priority Scheduling 한줄평

SCHEDULING

Priority Scheduling 한줄평

- 형평성: ★★★★★

SCHEDULING

Priority Scheduling 한줄평

- 형평성: ★★★★★
- 효율성: ★★★★★

SCHEDULING

Priority Scheduling 한줄평

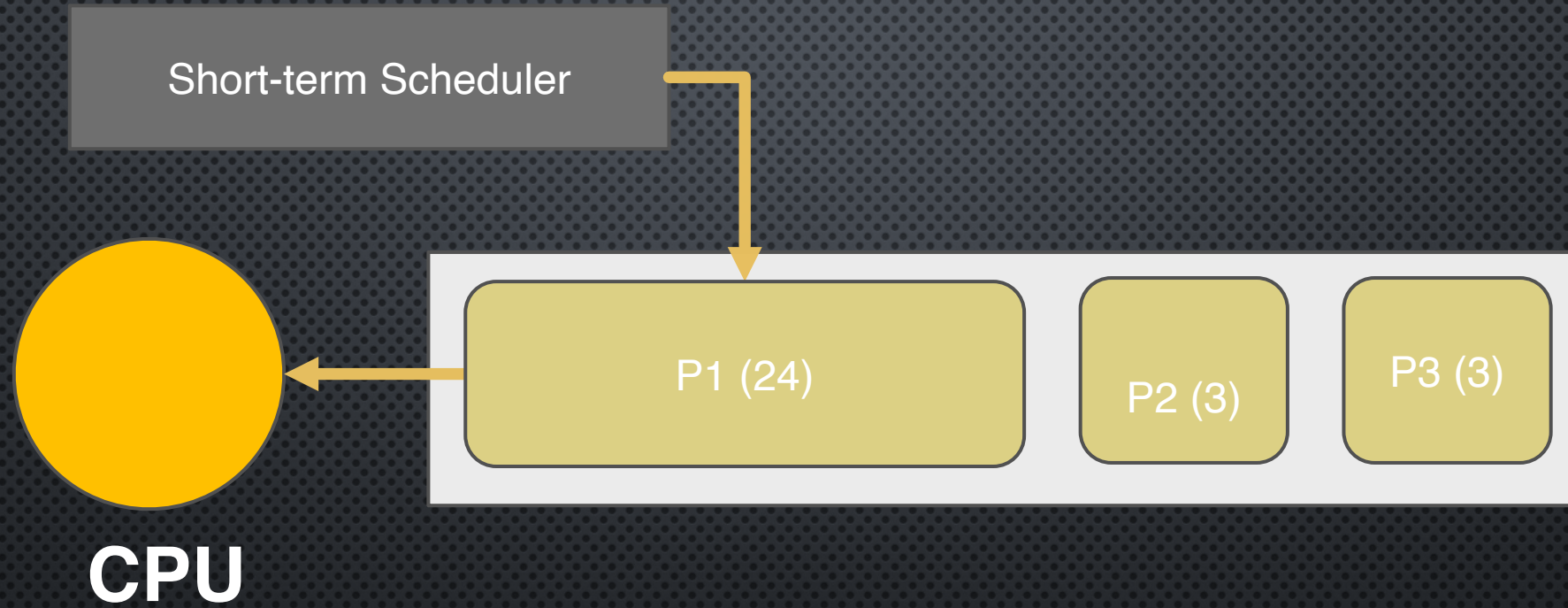
- 형평성: ★★★★★
- 효율성: ★★★★★
- 난이도: ★★★★★

SCHEDULING

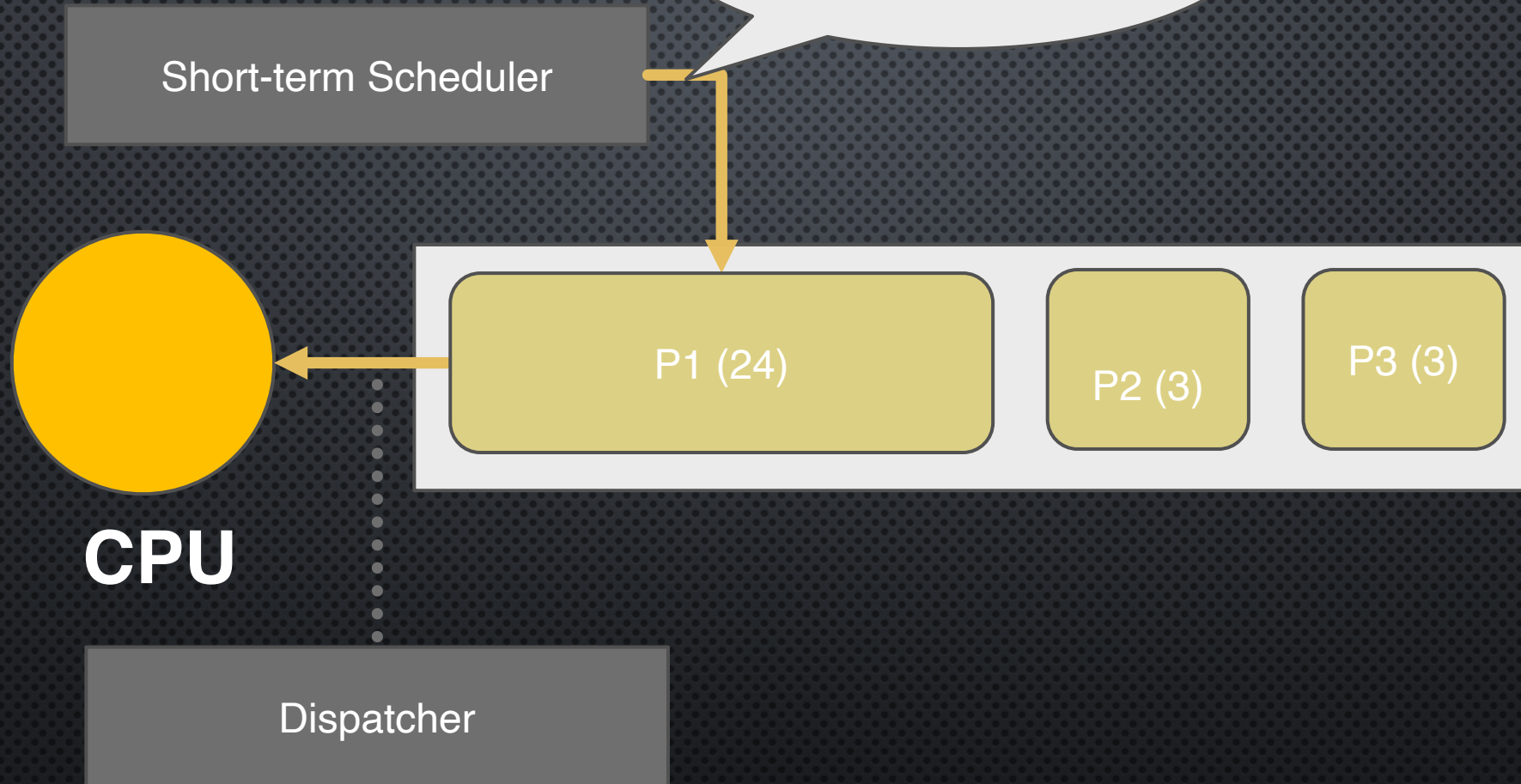
Priority Scheduling 한줄평

- 형평성: ★★★★★
 - 효율성: ★★★★★
 - 난이도: ★★★★★
-
- 단독으로 사용되기 보다는 Multi-level queue 등의 스케줄링에서 보조적으로 사용되는 경향이 있다.
 - SJF도 우선순위 스케줄링의 특수 케이스 중 하나로 생각할 수 있다.
 - 마찬가지로 낮은 우선순위의 프로세스는 계속 실행되지 못하고 starvation에 빠지는 문제가 생김.
 - 대기 시간에 비례하여 우선순위를 증가시키는 aging 기법을 사용하여 극복할 수 있다.

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN



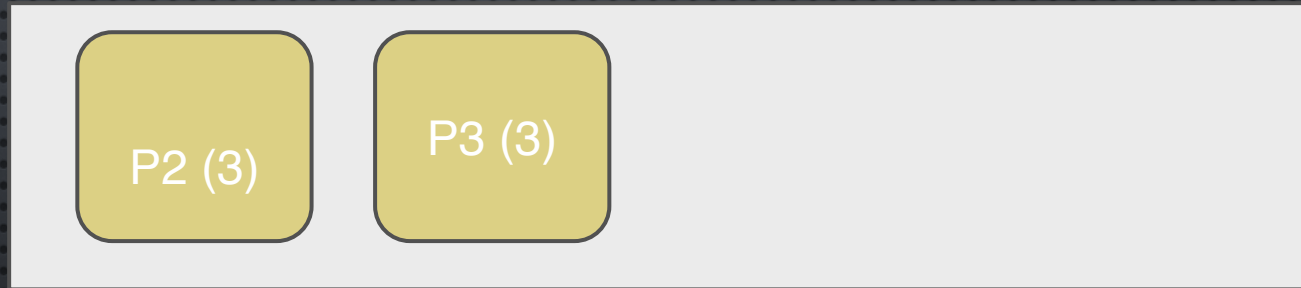
CPU 스케줄링 알고리즘 - 4) Round Robin



CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN



CPU

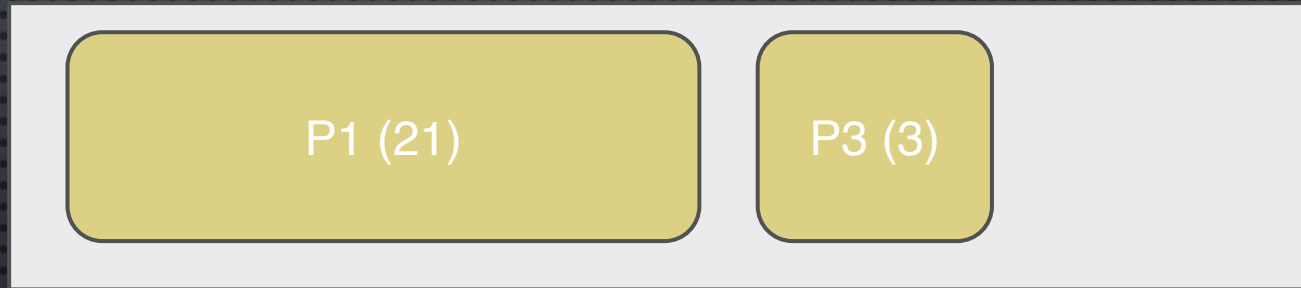


Ready Queue

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN



CPU

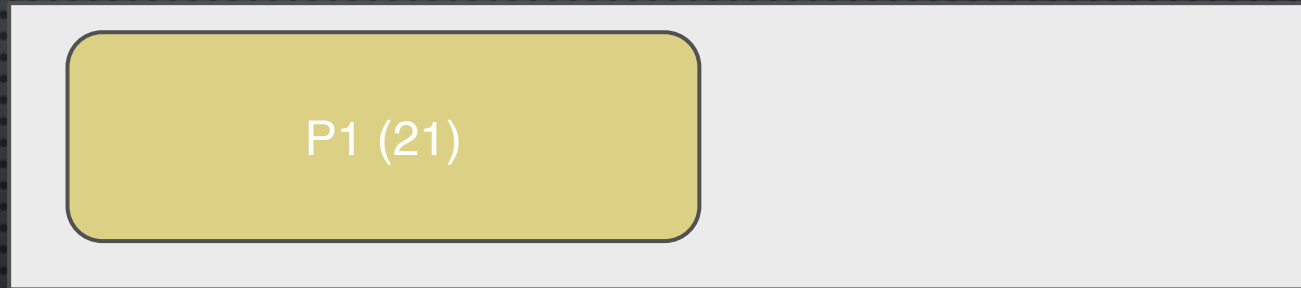


Ready Queue

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN

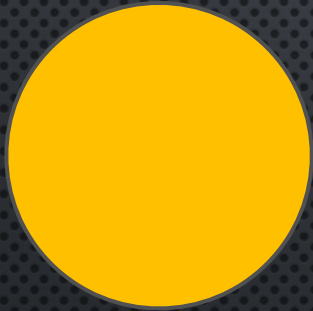


CPU

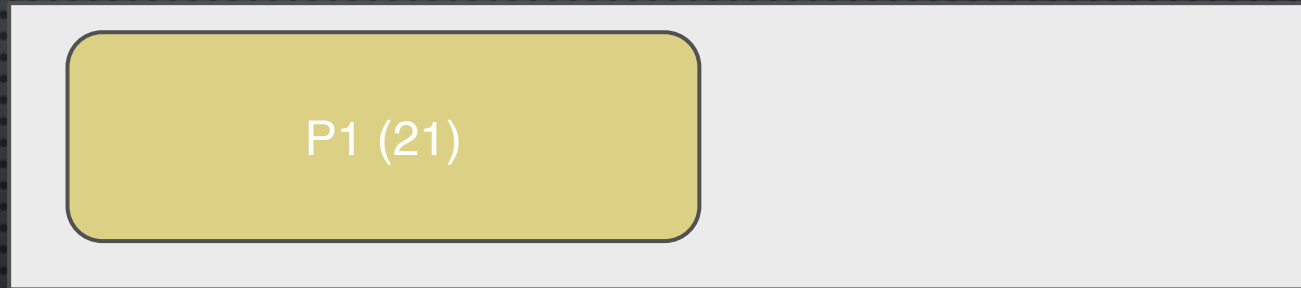


Ready Queue

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN



CPU



Ready Queue

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN

Round Robin 한줄평

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN

Round Robin 한줄평

- 형평성: ★★★★★

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN

Round Robin 한줄평

- 형평성: ★★★★★
- 효율성: ★★★☆☆

CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN

Round Robin 한줄평

- 형평성: ★★★★★
- 효율성: ★★★☆☆
- 난이도: ★★☆☆☆

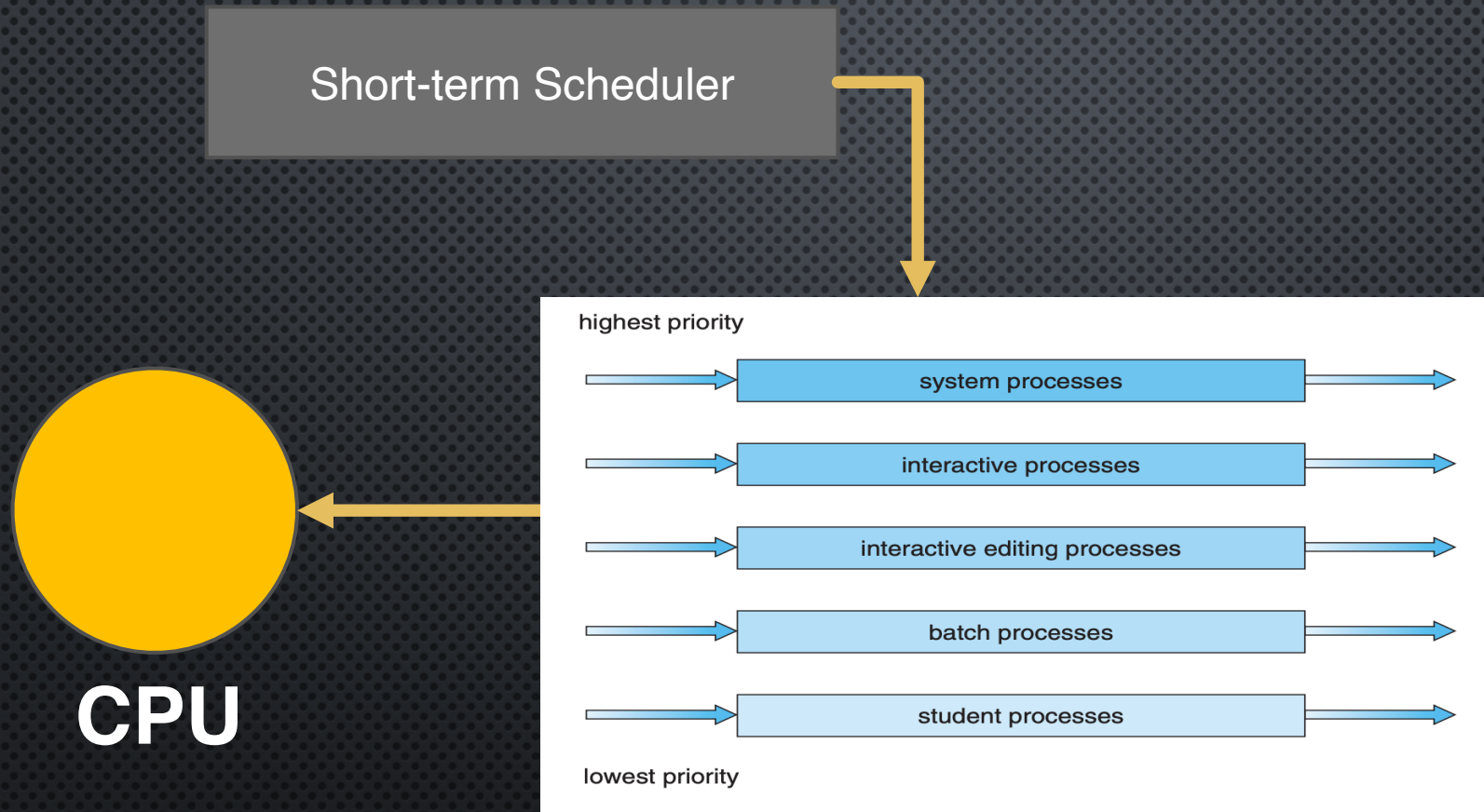
CPU 스케줄링 알고리즘 – 4) ROUND-ROBIN

Round Robin 한줄평

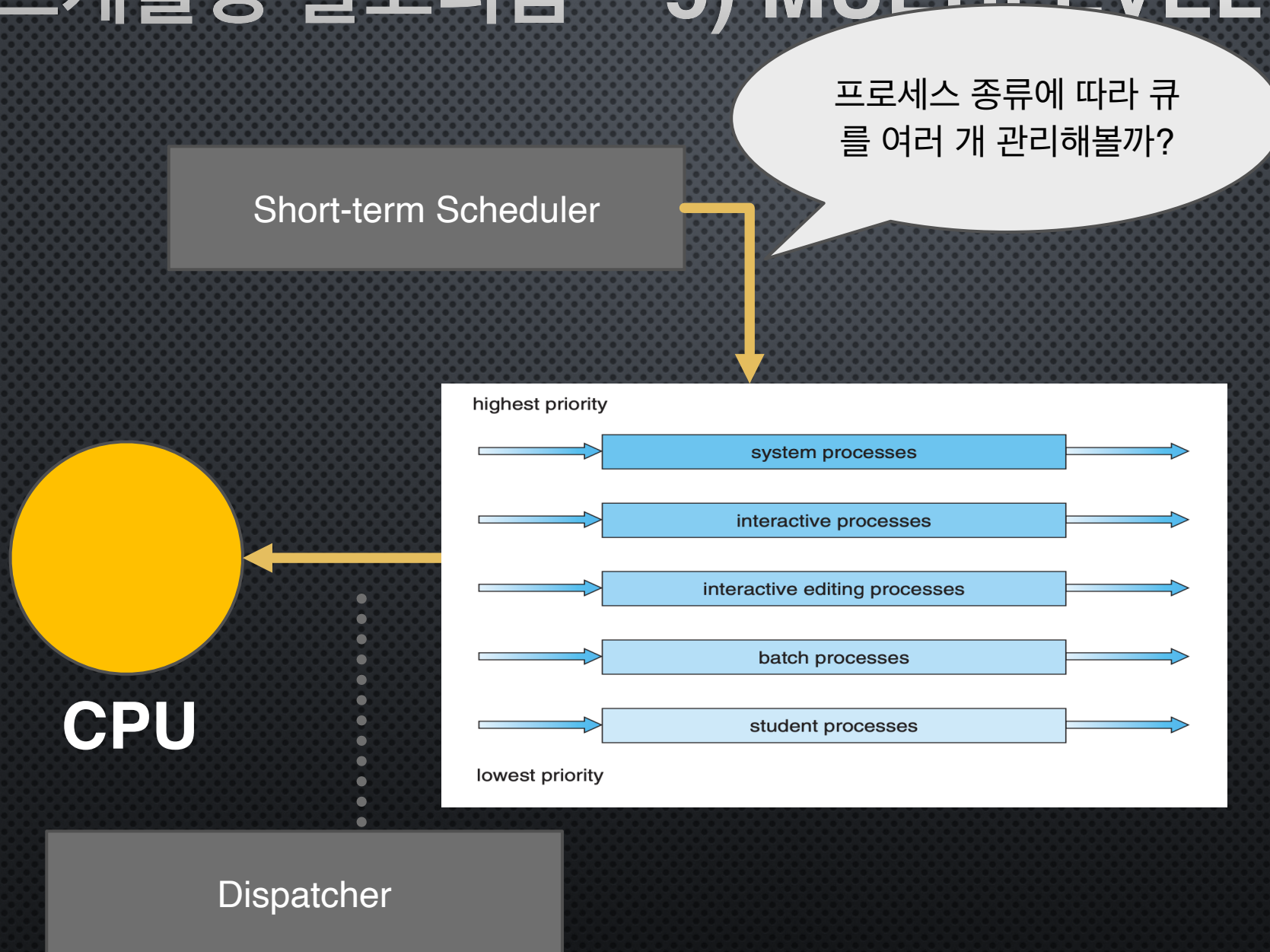
- 형평성: ★★★★★
- 효율성: ★★★★★
- 난이도: ★★★★★

- 모든 프로세스에게 동등한 우선순위를 주고 돌아가며 실행되는 방식
- Response time 측면에서 가장 효율적이다.
- 어느 프로세스도 $(n-1) * (\text{time quantum})$ 이상 기다리지 않는다.
- I/O bound job과 CPU bound job이 섞여있는 상황에서도 균일한 성능을 낸다.

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE



CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE



CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

- 각 프로세스는 기준에 따라 한 queue에 영원히 배정된다.
- 각 큐는 각자만의 스케줄링 방식을 채택한다. (RR+FCFS)
- Q. 그럼 큐를 나누는 기준은 무엇일까?

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

- 각 프로세스는 기준에 따라 한 queue에 영원히 배정된다.
- 각 큐는 각자만의 스케줄링 방식을 채택한다. (RR+FCFS)
- Q. 그럼 큐를 나누는 기준은 무엇일까?

메모리 크기

프로세스 우선순위

프로세스 종류

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

- 각 프로세스는 기준에 따라 한 queue에 영원히 배정된다.
- 각 큐는 각자만의 스케줄링 방식을 채택한다. (RR+FCFS)
- Q. 그럼 큐를 나누는 기준은 무엇일까?

Priority	Meaning
IDLE_PRIORITY_CLASS	기준 우선순위 4
NORMAL_PRIORITY_CLASS	기준 우선순위 9
HIGH_PRIORITY_CLASS	기준 우선순위 13
REALTIME_PRIORITY_CLASS	기준 우선순위 24
ABOVE_NORMAL_PRIORITY_CLASS	NORMAL_PRIORITY_CLASS보다 높고 HIGH_PRIORITY_CLASS보다 낮은 우선순위. 단, Windows NT and Me 이하(포함해서)에서는 지원되지 않는 우선순위이다.
BELOW_NORMAL_PRIORITY_CLASS	IDLE_PRIORITY_CLASS보다 높고 NORMAL_PRIORITY_CLASS보다 낮은 우선순위. 단, Windows NT and Me 이하(포함해서)에서는 지원되지 않는 우선순위이다.

[표 9-1] 프로세스의 우선순위 계층

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

Multilevel Queue 한줄평

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

Multilevel Queue 한줄평

- 형평성: ★★☆☆☆

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

Multilevel Queue 한줄평

- 형평성: ★★☆☆☆
- 효율성: ★★★★★

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

Multilevel Queue 한줄평

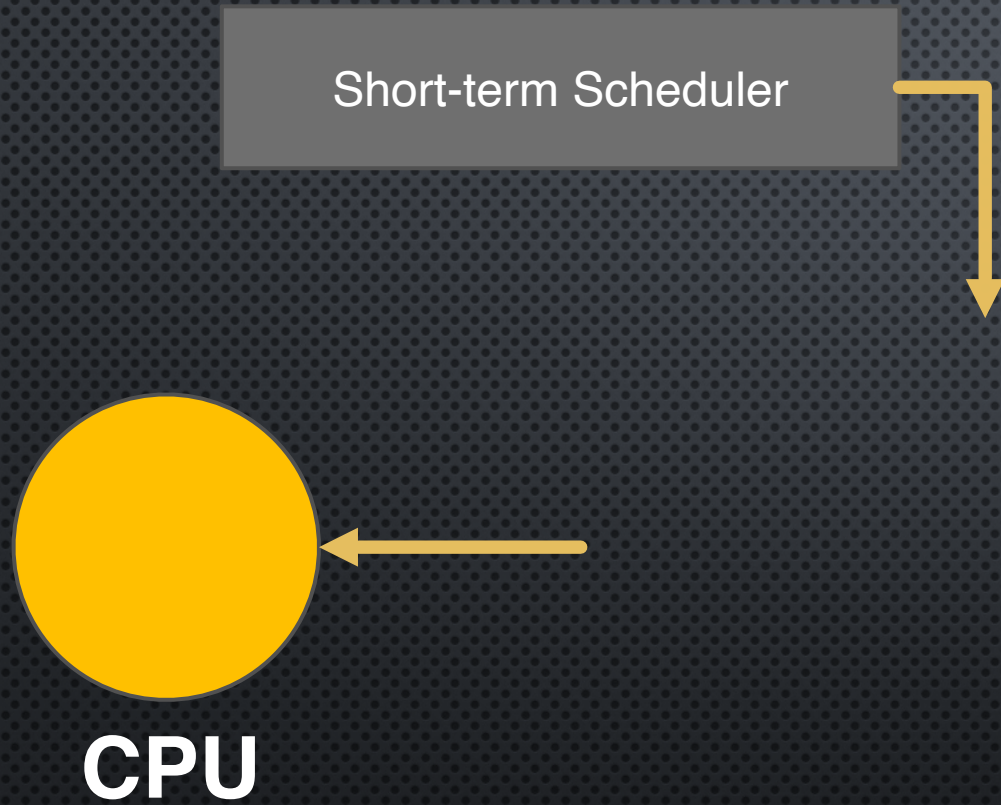
- 형평성: ★★☆☆☆
- 효율성: ★★★★★
- 난이도: ★★★★★

CPU 스케줄링 알고리즘 – 5) MULTILEVEL QUEUE

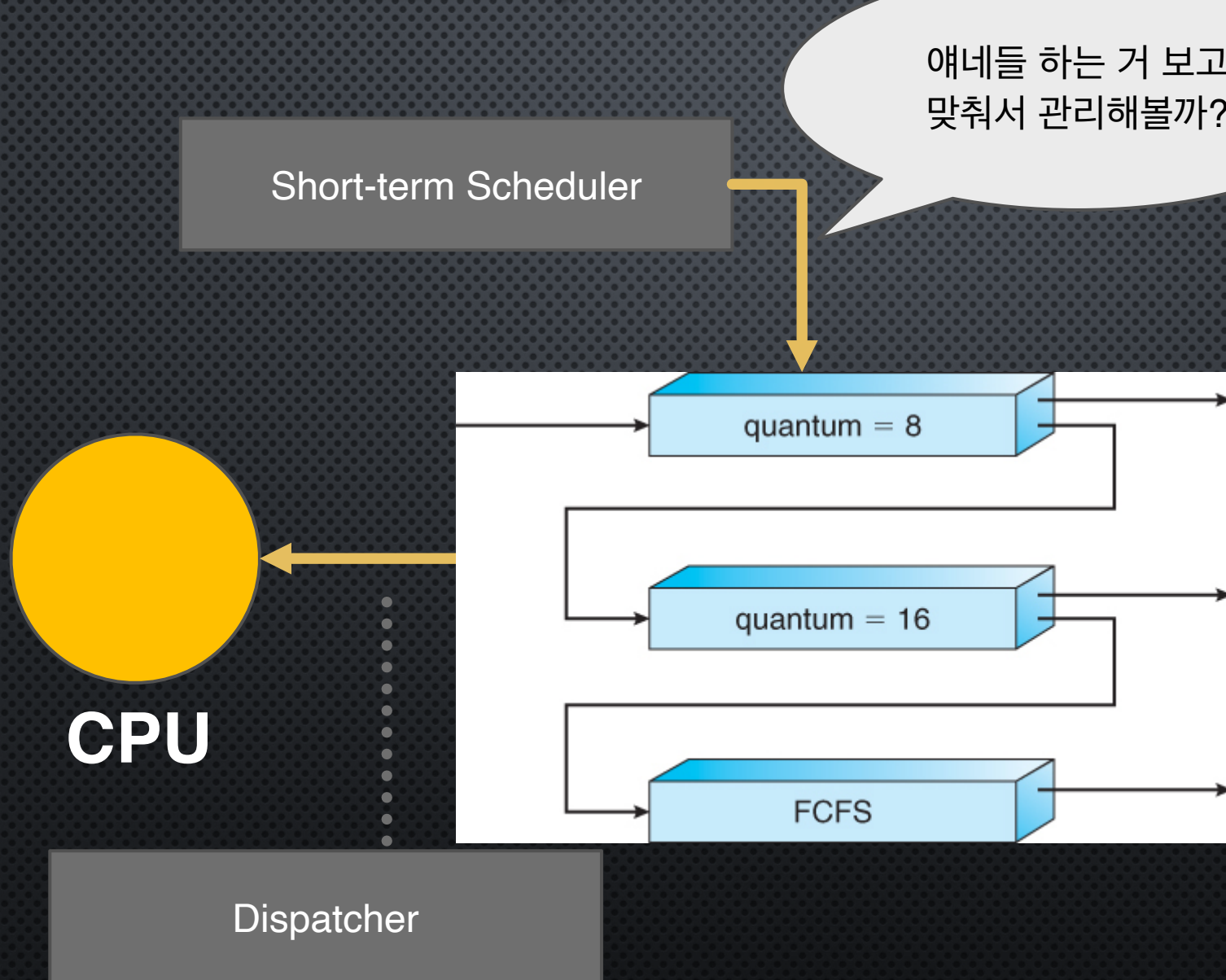
Multilevel Queue 한줄평

- 형평성: ★★☆☆☆
 - 효율성: ★★★★★
 - 난이도: ★★★★★
-
- 프로세스의 특성에 맞게 분배하여 효율성을 살린 스케줄링
 - 한번 배정받은 큐에 계속 있기 때문에 큐 배정에 대한 오버헤드가 적다.
 - 프로세스를 분류하고 여러 개의 큐를 관리하기 때문에 구현하기 어렵다.
 - 우선순위가 높은 큐부터 무조건 실행되는 방식으로 구현 시 starvation이 발생한다.
 - 큐 마다 Time slice를 제공하는 방식으로 해결할 수 있다.

CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE



CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE



CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE

- MQ 스케줄링과 비슷한 컨셉인데, 프로세스의 큐 간 이동이 가능하다.
- 마찬가지로 각 큐는 각자만의 스케줄링 방식을 채택한다.
- 보통 높은 우선순위끼리는 RR 스케줄링, 낮은 우선순위는 FCFS를 채택한다.
- Q. 그럼 어떤 프로세스가 높은 우선순위를 가져야할까?

CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE

Multilevel Feedback Queue 한줄평

CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE

Multilevel Feedback Queue 한줄평

- 형평성: ★★★★★

CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE

Multilevel Feedback Queue 한줄평

- 형평성: ★★★★★
- 효율성: ★★★★★

CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE

Multilevel Feedback Queue 한줄평

- 형평성: ★★★★★
- 효율성: ★★★★★
- 난이도: ★★★★★

CPU 스케줄링 알고리즘 – 6) MULTILEVEL FEEDBACK QUEUE

Multilevel Feedback Queue 한줄평

- 형평성: ★★★★★
 - 효율성: ★★★★★
 - 난이도: ★★★★★
-
- MQ 스케줄링의 inflexibility라는 단점을 보완한 스케줄링
 - 기본적으로 Round-Robin을 따르되, CPU bound job의 우선순위를 떨어뜨리는 식으로 관리한다.
 - 큐 간 이동, 큐 배정, 가장 효율적인 스케줄링 결정 등 구현하기 매우 어렵다.
 - 현재 Windows, Mac OS 에서 사용되는 스케줄링 방법

감사합니다!