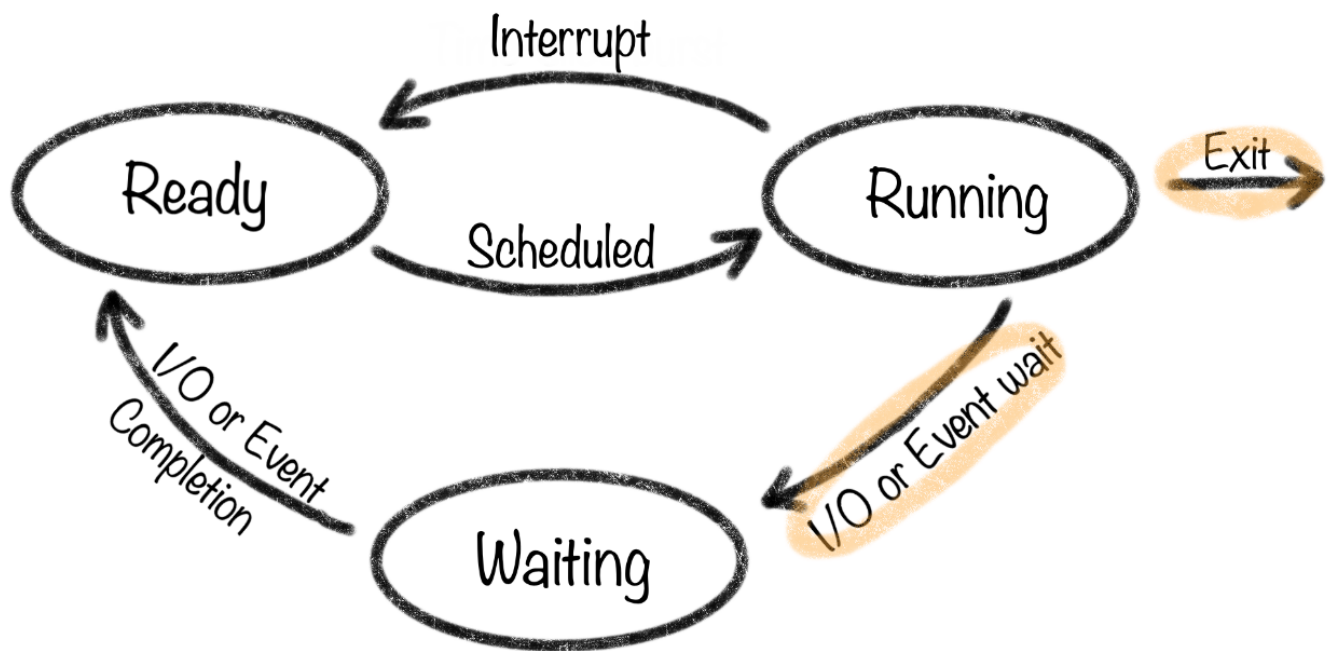


1. ??(Preemptive) ????

- 在 100 个数据块中，只有 10 个数据块被 CPU 访问
- 在 100 个数据块中，只有 10 个数据块被 CPU 访问
- 在 100 个数据块中，只有 10 个数据块被 CPU 访问

□□ □□ □□□ (Priority Scheduling) : □ □□□□ □□□□ □□□ □□

2. ???(Non-preemptive) ????



“ 프로세스 실행 순서 : 프로세스가 실행 중인 상태에서 CPU를 점유하고 있는 동안에 다른 프로세스가 실행될 기회를 갖지 못함 ”

이러한 비선점형 스케줄링 방식에서는 CPU가 한 프로세스를 실행할 때, 그 프로세스가 실행을 완료할 때까지 다른 프로세스가 실행될 기회를 갖지 못함.

이러한 비선점형 스케줄링 방식에는

1. FCFS (First Come First Serve) : 프로세스가 실행 순서에 따라 실행됨

2. SJF (Shortest Job First) : 실행 시간이 짧은 프로세스가 먼저 실행됨

3. Round Robin (RR) : 실행 시간이 긴 프로세스도 실행될 기회를 갖기 위해, 실행 시간을 제한함

4. HRN (Highest Response Ratio Next) : 실행 시간이 길고 대기 시간이 긴 프로세스가 먼저 실행됨

프로세스 실행 순서

프로세스 실행 순서 : 프로세스가 실행 중인 상태에서 CPU를 점유하고 있는 동안에 다른 프로세스가 실행될 기회를 갖지 못함. 프로세스 실행 순서 : 프로세스가 실행 중인 상태에서 CPU를 점유하고 있는 동안에 다른 프로세스가 실행될 기회를 갖지 못함.

1.FCFS

- ☐ ☐
- ☐ ☐ ☐ ☐

02. 다음과 같은 3개의 작업에 대하여 FCFS 알고리즘을 사용할 때, 임의의 작업 순서로 얻을 수 있는 최대 평균 반환 시간을 T, 최소 평균 반환 시간을 t라고 가정했을 경우 T-t의 값은?

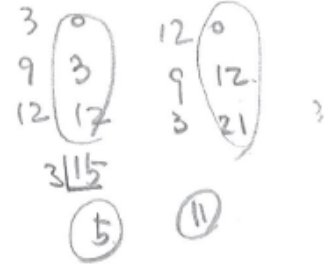
프로세스	실행시간
P1	9
P2	3
P3	12

- ① 3
③ 5

- ② 4
④ 6

차이?

$$11 - 5 = 6$$



[정답] ④

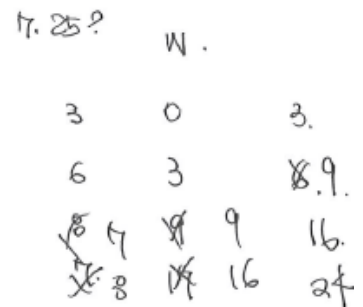
- ☐ ☐ ☐ t ☐ ☐ ☐ ☐ (P2 → P1 → P3)
- $t = (0 + 3 + 12) / 3 = 15 / 3 = 5$
- ☐ ☐ ☐ T ☐ ☐ ☐ ☐ (P3 → P1 → P2)
- $T = (0 + 12 + 21) / 3 = 33 / 3 = 11$
- $T - t = 11 - 5 = 6$

2.SFJ

- ☐ ☐
- ☐ ☐ ☐ ☐ ☐
- ☐ ☐ / ☐ ☐ / ☐ ☐ ☐ ☐ ☐

03. SJF(Shortest Job First) 스케줄링에서 다음과 같은 작업들이 준비상태 큐에 있을 때 평균 반환시간과 평균 대기시간은?

프로세스	실행시간
P-1	6
P-2	3
P-3	8
P-4	7



□□□□

□□□□

□□□□

□□□□

P-2

3

0

3

P-1

6

3

9

P-4

7

9

16

P-3

8

16

24

$$28 / 4 = 7$$

$$52 / 4 = 13$$

04. 다음과 같은 Task List에서 SJF방식으로 Scheduling할 경우 Task 2의 종료 시간을 구하면?

<Task List> 도착시간 0일 때 71만 도 착했기 때문에 실행시간 6부터 실행된다.

Task	도착시간	실행시간
Task 1	0	6
Task 2	1	3
Task 3	2	4

6 3

Task 2 → 3 → 1 X.

Task 1 → 2 → 3 0

① 3

② 6

③ 9

④ 13

8?.

1 3 대.
2 4 4.
0 6

[정답] ③

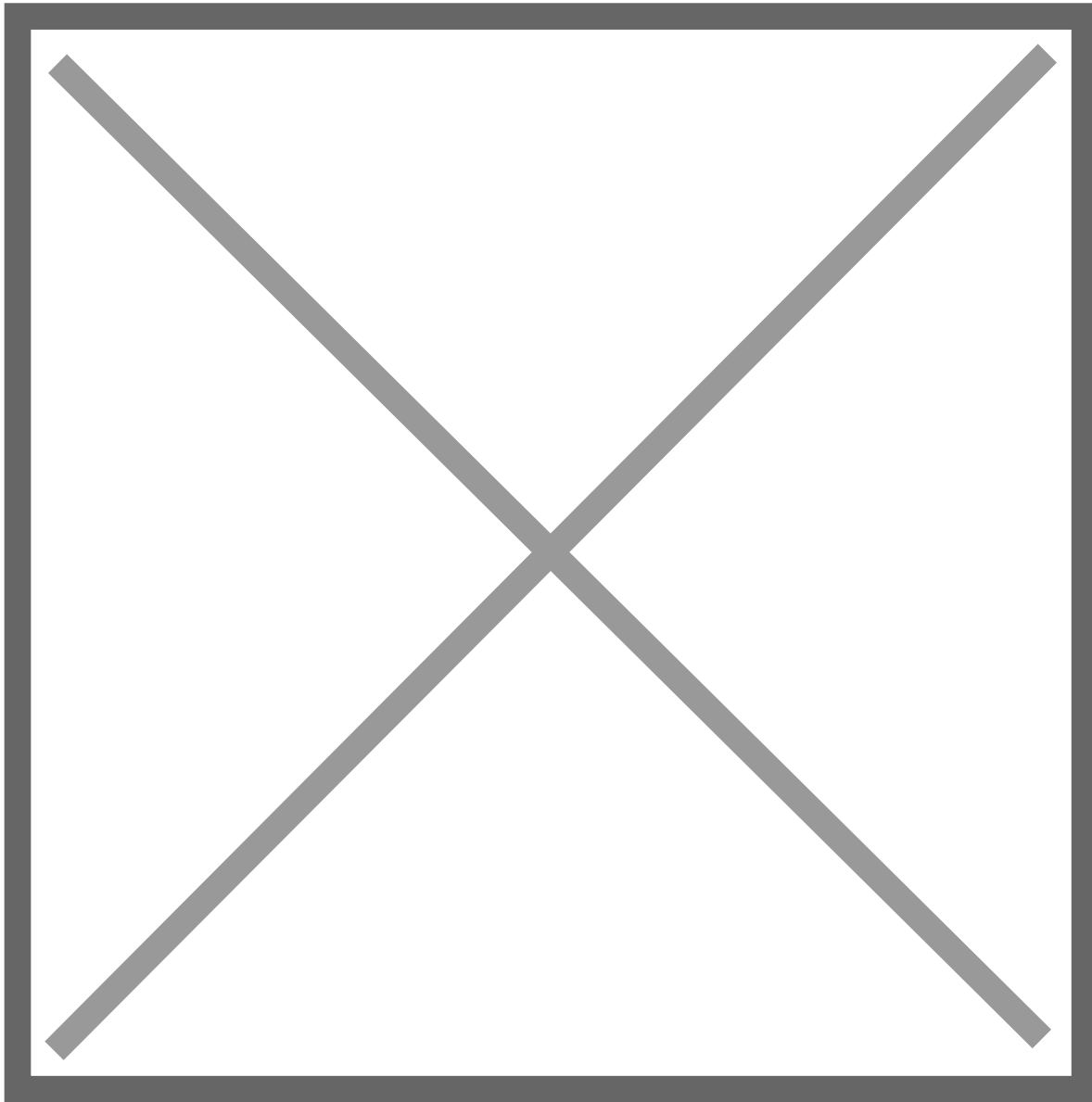
Task2의 실행 시간 = 2 * (Waiting Time) + 2 * (Burst Time) = 6 + 3 = 9

이 실행 시간 0에서 T1의 실행 시간까지 기다리는 시간은 T1의 실행 시간입니다. (이 ?는 T1의 실행 시간입니다)

3.HRN

- HRN은 1
- HRN은 실행 중인 프로세스의 수, 실행 중인 프로세스의 실행 시간, 실행 중인 프로세스의 실행 시간
- $HRN = 1 + \frac{실행 시간}{실행 시간}$
- → HRN은 실행 중인 프로세스의 수 !

이 HRN은 실행 중인 프로세스의 수

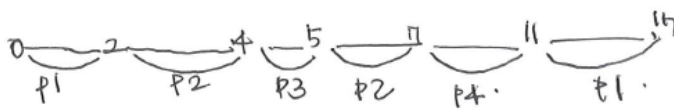


- T_{wait} (Waiting Time) = $T_{turnaround} - T_{service}$
- $T_{service}$ (Service Time) = B , B Burst Time
- SJF (SJF) \rightarrow T_{wait} is minimized, \rightarrow $T_{turnaround}$ is minimized, \rightarrow $T_{service}$ is minimized (starvation) \rightarrow T_{wait} !

01. 다음 표는 단일 CPU에 진입한 프로세스의 도착 시간과 처리하는 데 필요한 실행 시간을 나타낸 것이다. 프로세스 간 문맥 교환에 따른 오버헤드는 무시한다고 할 때, SRT(Shortest Remaining Time) 스케줄링 알고리즘을 사용한 경우 네 프로세스의 평균 반환시간(Turnaround Time)은?

프로세스	도착 시간	실행 시간
P1	0	8 / 6
P2	2	4 X 2
P3	4	1 X
P4	6	4

- ① 4.25 ② 7
③ 8.75 ④ 10



평균시간 = 총실행시간 - 도착시간 [정답] ②

$$P1 = 17 - 0 = 17$$

$$P2 = 7 - 2 = 5$$

$$P3 = 5 - 4 = 1$$

$$P4 = 11 - 6 = 5$$

02. 다음 표는 단일 CPU에 진입한 프로세스의 도착 시간과 처리하는 데 필요한 실행 시간을 나타낸 것이다. 프로세스 간 문맥 교환에 따른 오버헤드는 무시한다고 할 때, SRT(Shortest Remaining Time) 스케줄링 알고리즘을 사용한 경우 네 프로세스의 평균 반환시간(Turnaround Time)은?

프로세스	도착 시간	실행 시간
P1	0	7 5
P2	②	4 X 2
P3	4	1 X
P4	5	4

$$P1 = 16 - 0 = 16$$

$$P2 = 7 - 2 = 5$$

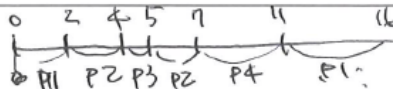
$$P3 = 5 - 4 = 1$$

$$P4 = 11 - 5 = 6$$

$$28 / 4 = 7$$

- ① 4.25
③ 8.75

- ② 7
④ 10



[정답] ②

5.RR(Round Robin)

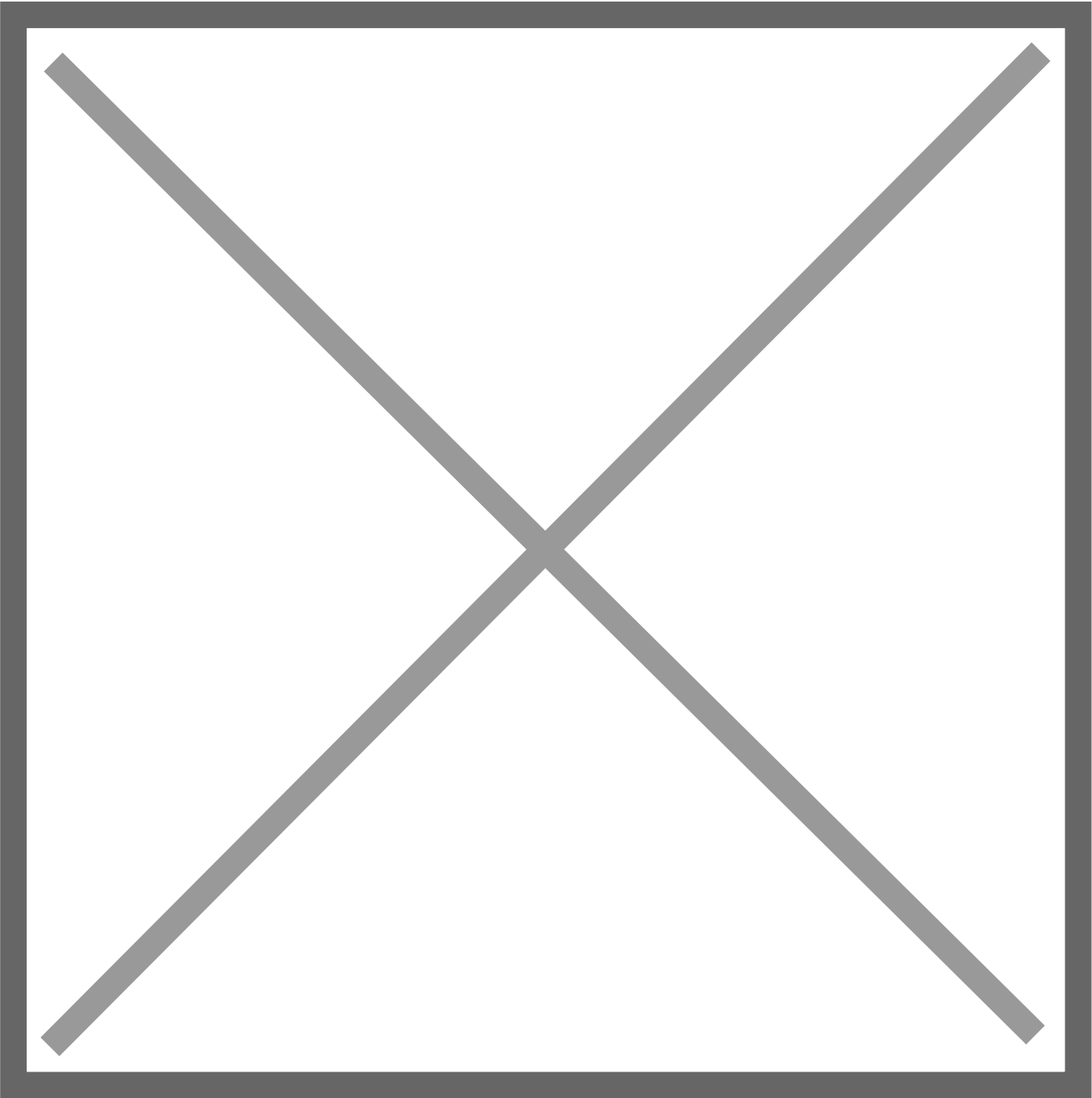
-
- (Time Quantum)
- (FIFO) ()
-
-
- + +
-
- **3** **3**
- : **CPU** / / !
- (Waiting Time, WT) = (FT) - (AT) - (BT)
- (Turnaround Time, TAT) = (FT) - (AT)

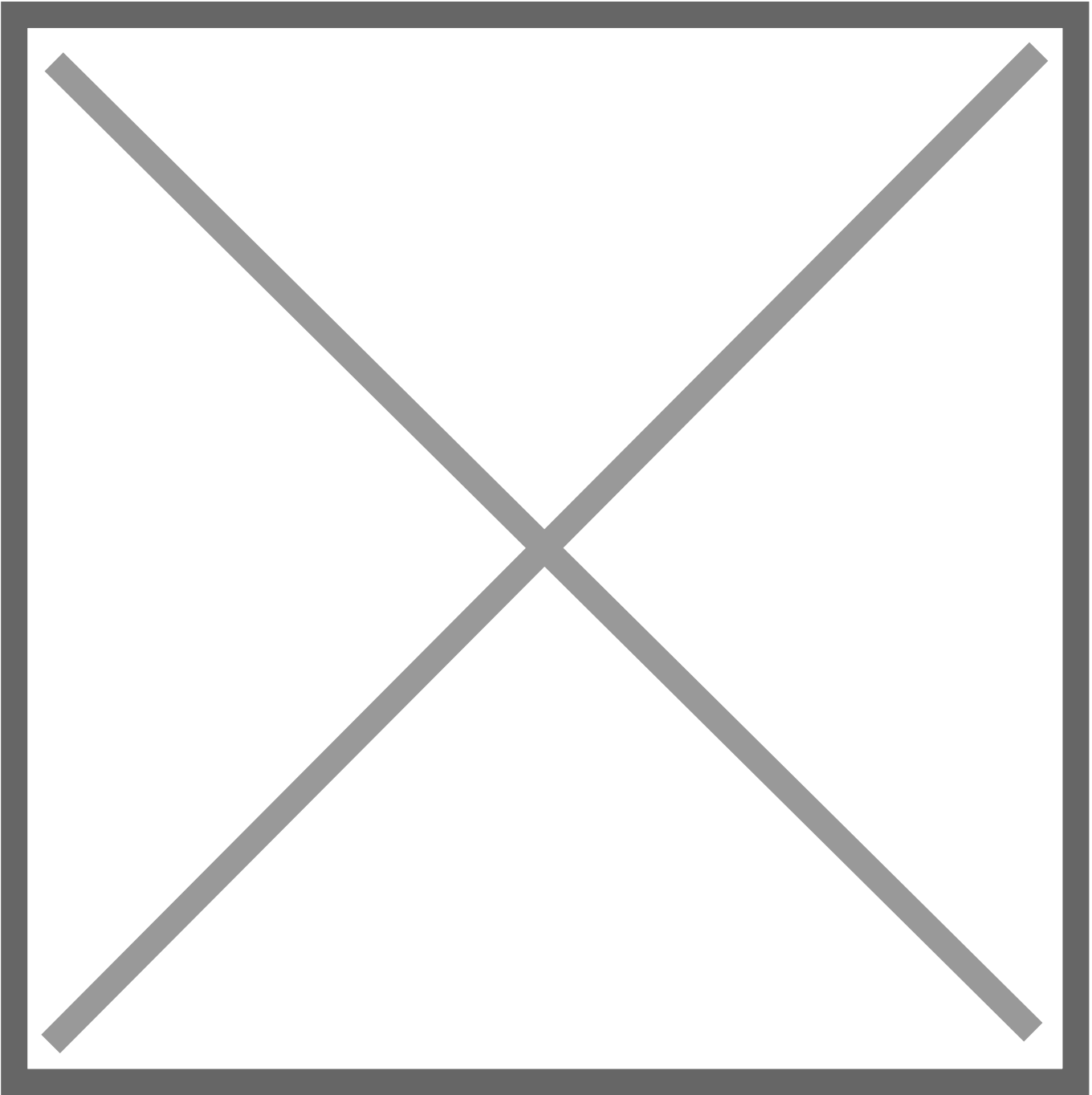
01. 라운드로빈(Round-Robin) 방식으로 스케줄링 할 경우, 입력된 작업이 다음과 같고 각 작업의 CPU 할당 시간이 4시간일 때, 모든 작업을 완료하기 위한 CPU의 사용 순서를 옳게 나열하시오.

작업	입력시간	수행시간
A	10:00	5시간
B	10:30	10시간
C	12:00	15시간

A. ~~5~~ ~~9~~.

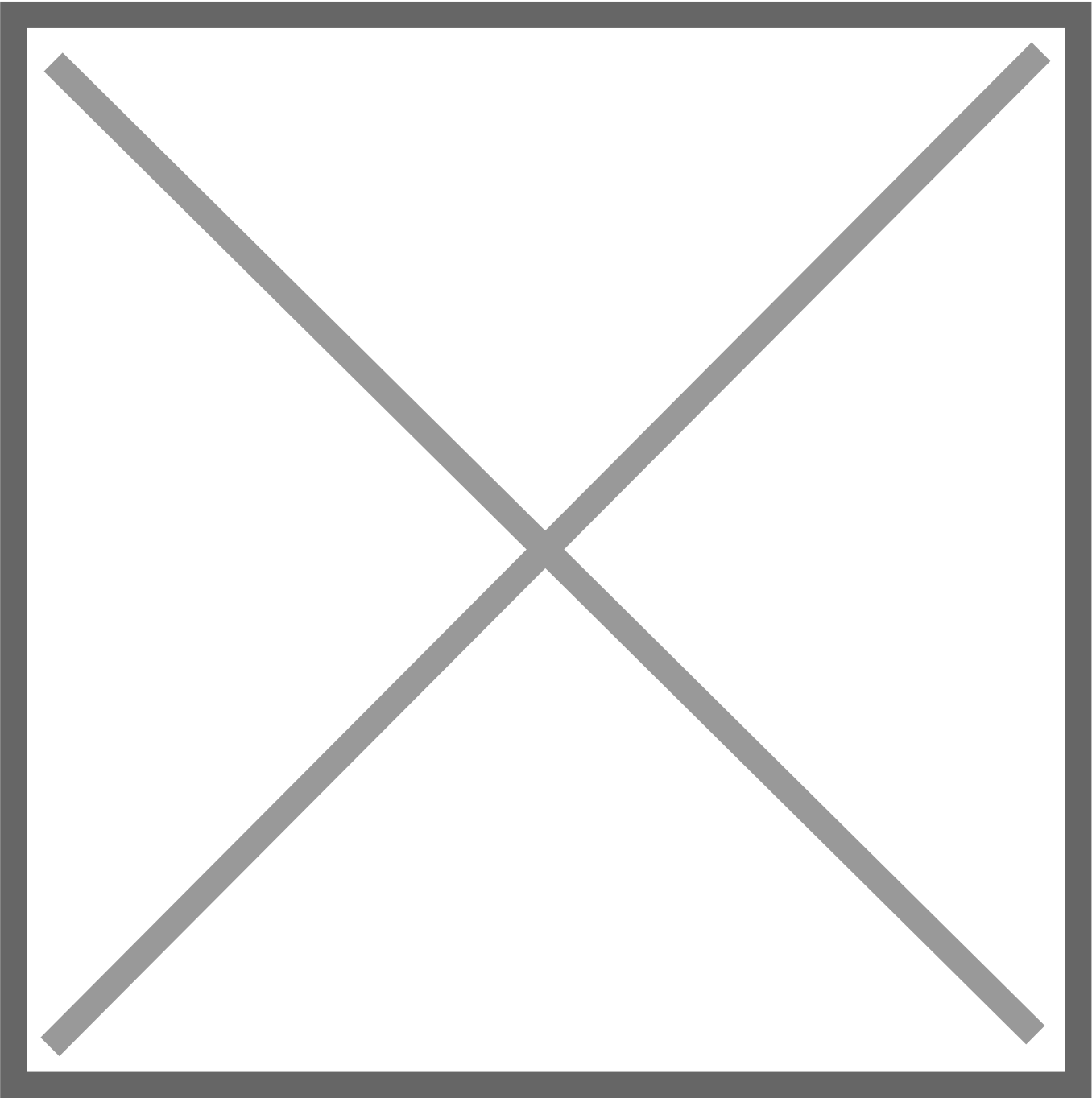
B. ~~10~~ ~~6~~ (2)

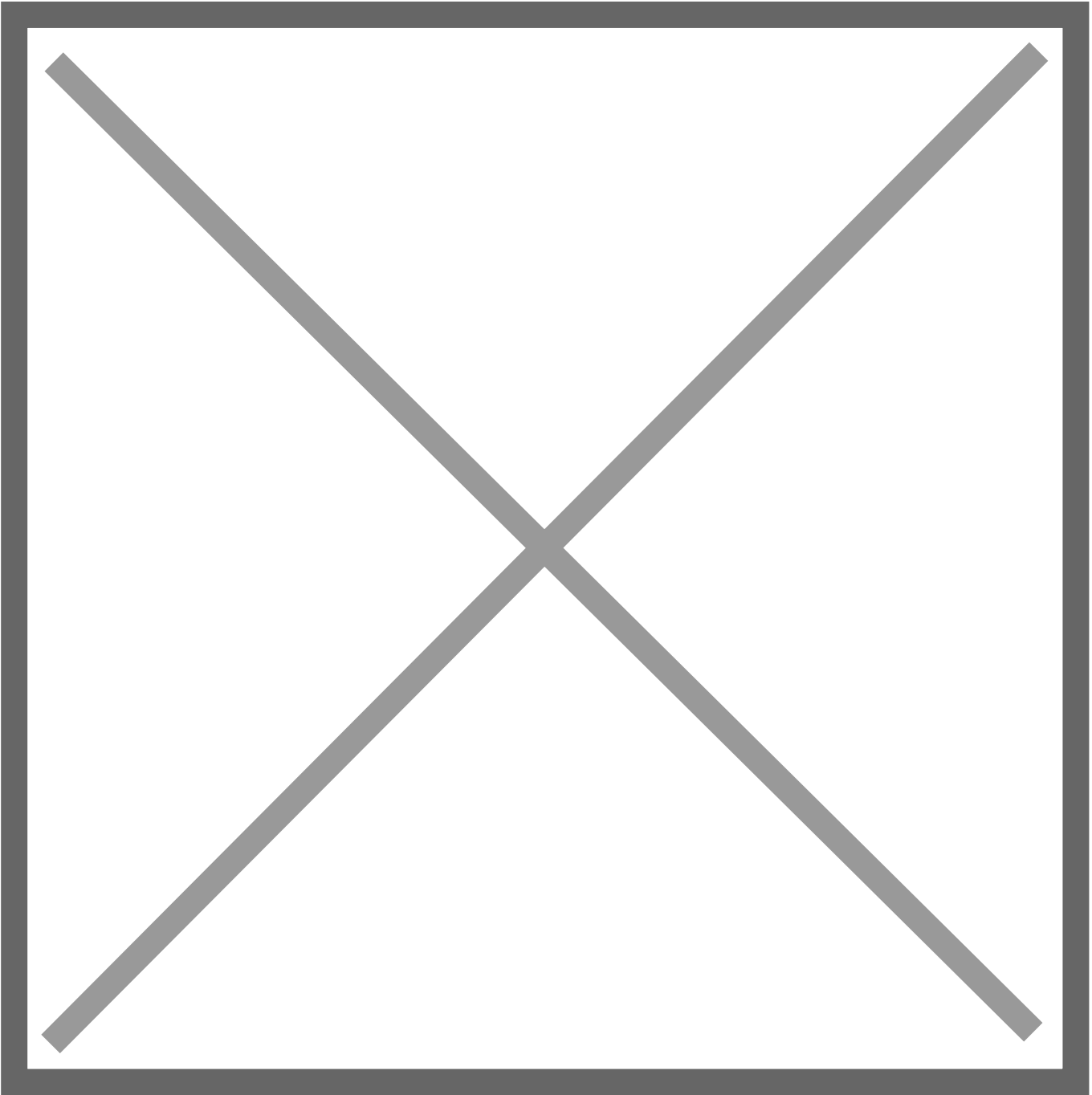




□□ □□□□ □□

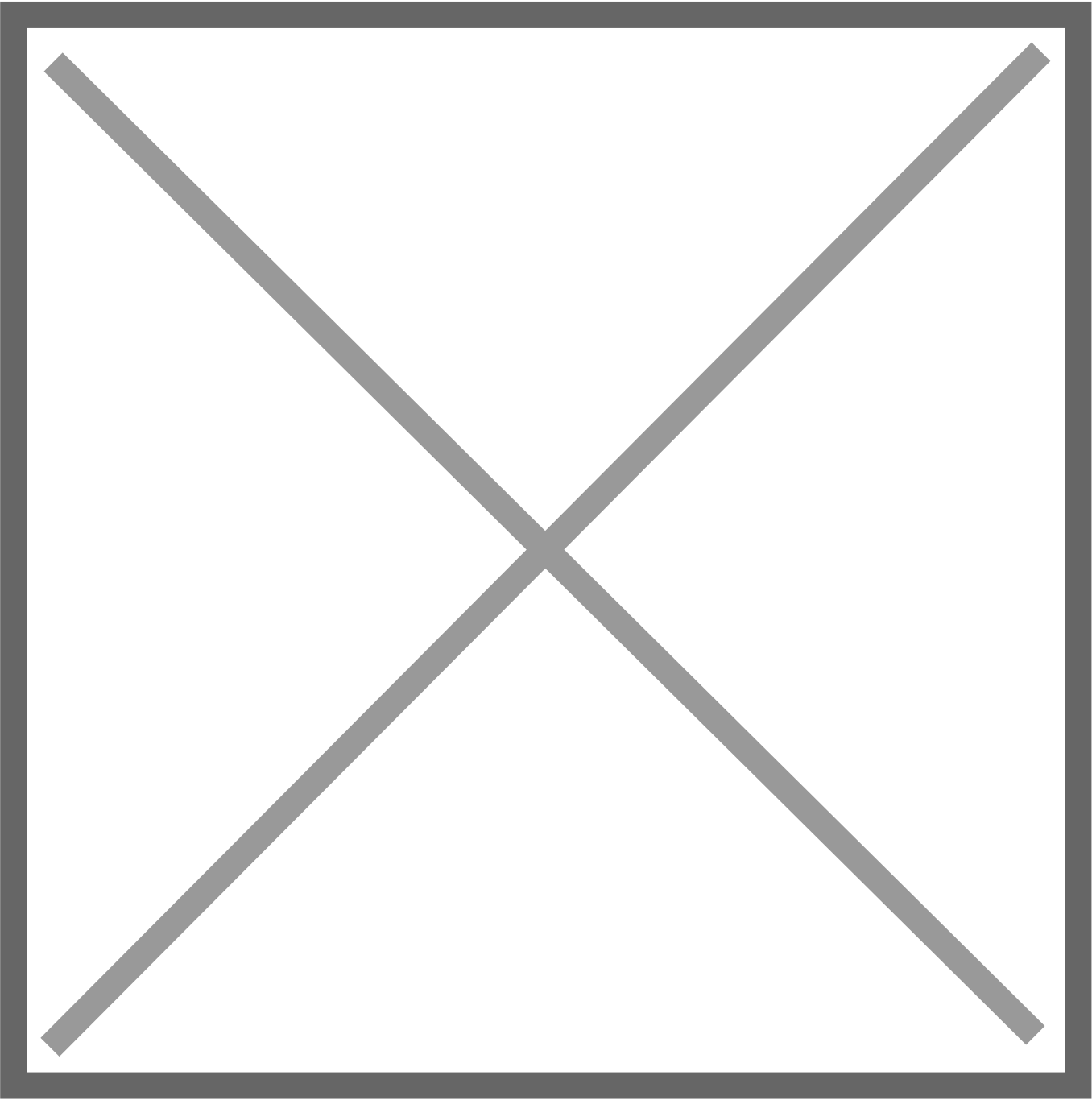
“ □□ □□ (Waiting Time, WT) = □□ □□ (FT) - □□ □□ (AT) - □□ □□ (BT)



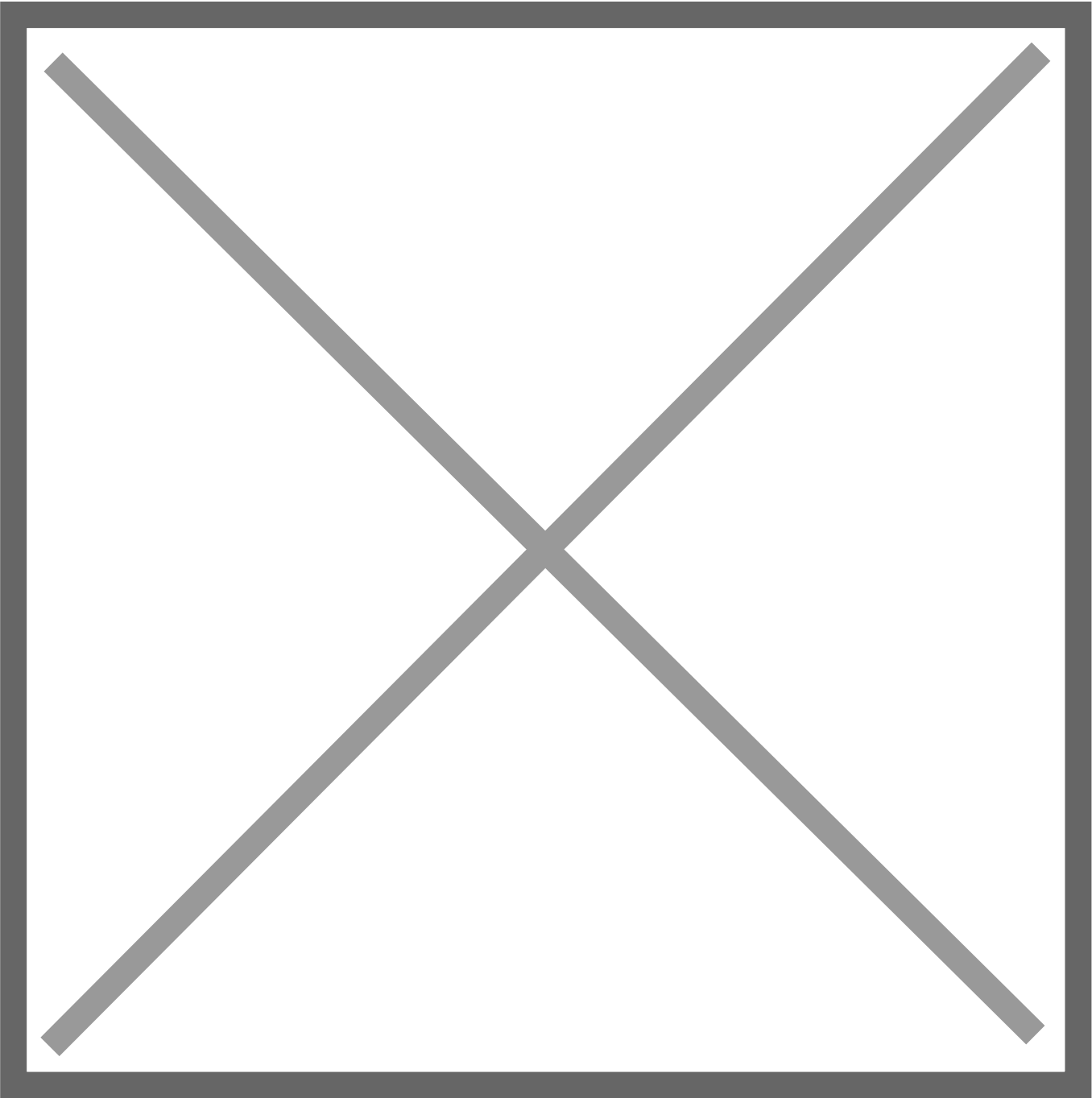


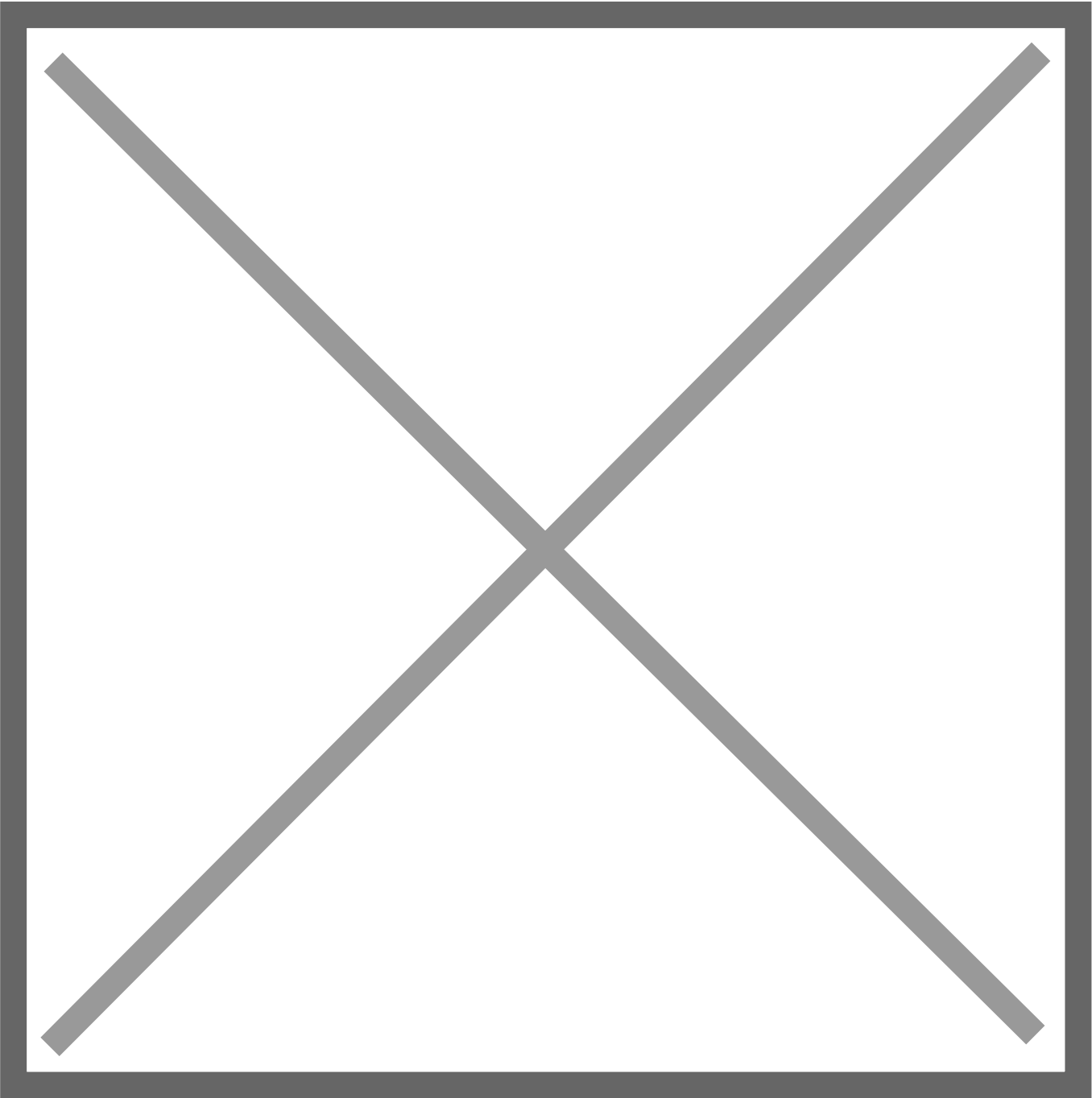
□□ □□□□ □□

“ □□ □□ (Turnaround Time, TAT) = □□ □□ (FT) - □□ □□ (AT)

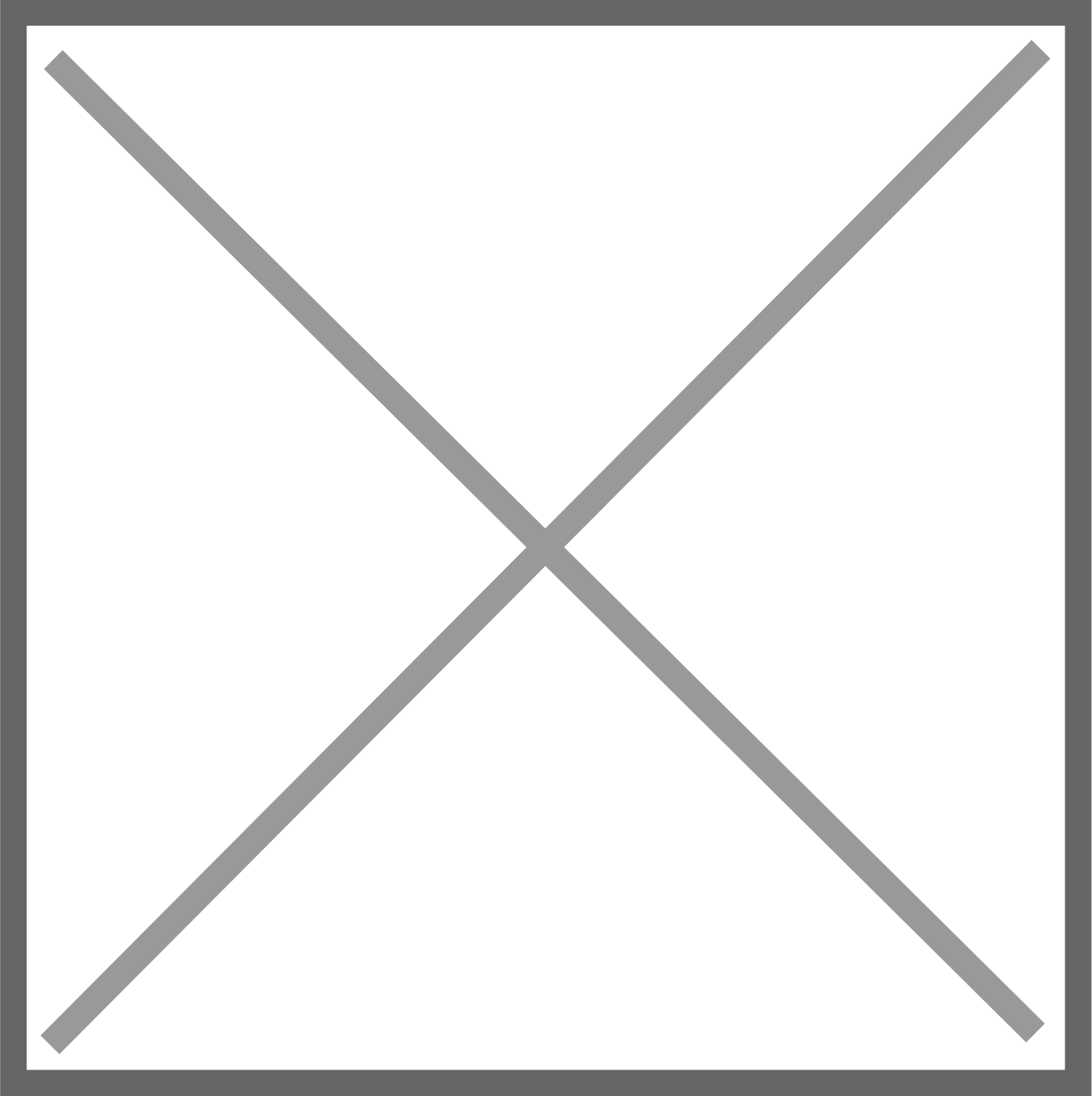


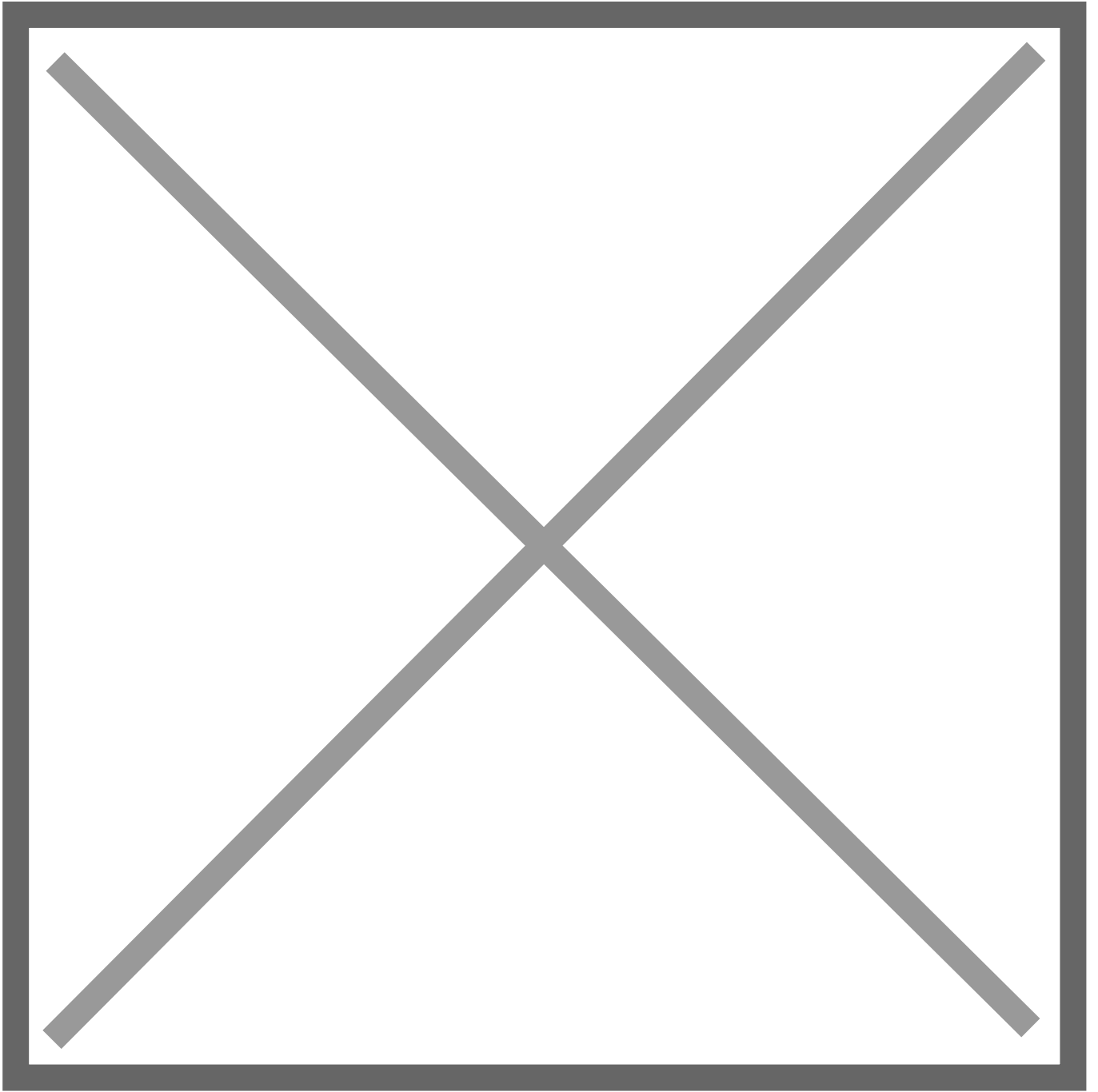
- 11 111 11111 11 0111 1111 11 (111 11 1111 111 11)











□□ □□

1. □□□□ □□□□□□ □□□ □□
2. <https://buly.kr/8eljGok>