Chapter 5

— 例题与作业

北京邮電大學

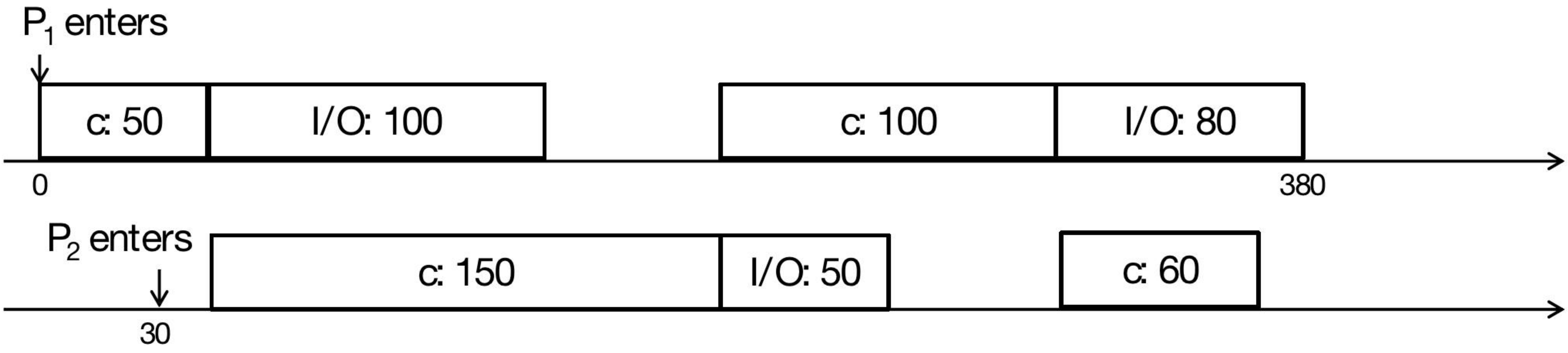
Examples and Exercises

- Examples
  - 例1. 并发进程吞吐量
  - 例2. 优先级/调度顺序 vs 吞吐量/利用率
  - 例3. HRRF scheduling
  - 例4. Scheduling in real-time systems
  - 例5. Job scheduling + process scheduling /长期调度 + 短期调度
  - **例6. 作业调度+进程调度+对外设竞争+对主存竞争**
  - 例7. 时间片长度 vs CPU利用率
  - 例8. （反例）response time分析
- Exercises
  - 2 CPU、3进程调度
  - 两级队列反馈调度

Example 1 并发进程吞吐量

- In a multiprogramming batch system, there are two processes P1 and P2 concurrently running on a single CPU. P2 enters into the system 30ms later than P1, and their executing traces, i.e. alternating sequences of CPU bursts and I/O bursts, are as follows
  - P_1: computing, 50ms → I/O operation, 100ms → computing, 100ms → I/O operation, 80ms
  - P_2: computing, 150ms → I/O operation, 50ms → computing, 60ms
- It is assumed that time costs of CPU scheduling and process switch are omitted, illustrate the executing traces of P_1 and P_2 by charts, and compute:
  - 1) the maximal throughput for completing these two processes
  - 2) the waiting times for P_1 and P_2 respectively

01 Example 1



- throughput=2/(50+150+100+80)=2/380ms
- waiting time for P_1 : 50ms

  waiting time for P_2 : (50-30) + 50= 70ms

扩展：作业，多核多线程，3个进程/线程，2个CPU

Example 2 优先级/调度顺序 vs 吞吐量/利用率

- 单CPU系统正在执行三个进程P_1、P_2和P_3，各进程的计算（CPU）时间和I/O时间比例如下

进程	计算时间	I/O时间
P_1	90%	10%
P_2	50%	50%
P_3	15%	85%

- 为提高系统资源利用率，合理的进程优先级设置应为__B__，为什么?
  - A. $P_1>P_2>P_3$ B. $P_3>P_2>P_1$
    C. $P_2>P_1=P_3$ D. $P_1>P_2=P_3$
- 注：优先级高的进程优先执行，假设3个进程的I/O操作可并行执行

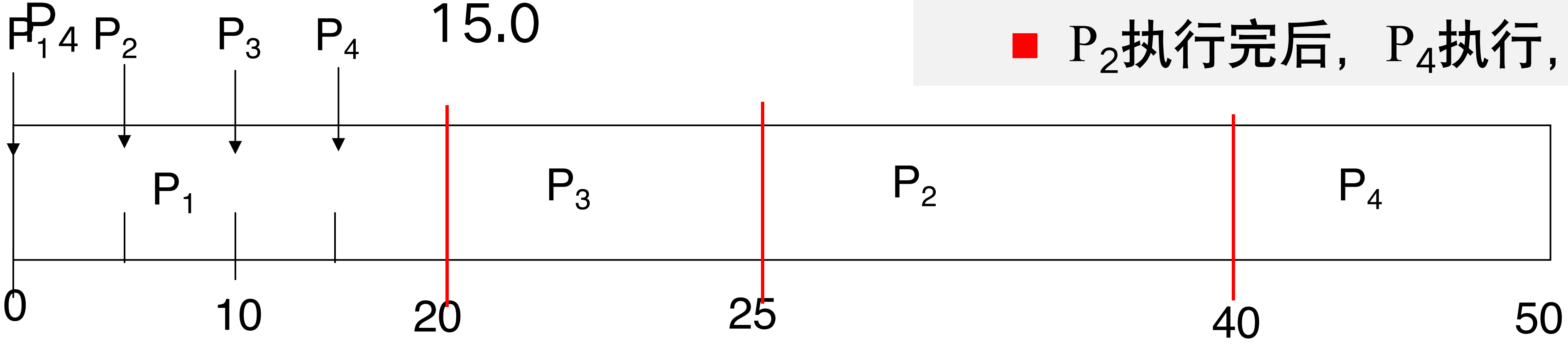
02 优先级/调度顺序 vs 吞吐量/利用率

- 原答案
  - 为了合理地设置进程优先级，应将进程的CPU时间和I/O时间做综合考虑
  - 对于优先级调度算法，一般来说，I/O型作业的优先权是高于计算型作业的优先权，这是由于I/O操作需要及时完成，它没有办法长时间保存所要输入/输出的数据，所以考虑到系统资源利用率，要选择I/O繁忙型作业有更高的优先级
  - 未必，**能否画出甘特图？？**
- 正确答案：**从CPU竞争导致的进程等待角度**
  - 3个进程的I/O操作可并行，但只能竞争使用1个CPU。当1个进程处于CPU burst占用CPU时，其它进程只能等待CPU或在I/O burst下进行I/O操作
  - 为了减少由于CPU竞争导致的进程等待，根据SJF算法原理，应优先安排CPU burst短的进程，因此调度顺序/优先级为P_3、P_2、P_1

Example 3 HRRF调度

Process	Arrival Time	Burst Time
P_1	0.0	20
P_2	5.0	15
P_3	10.0	5
P_4	15.0	10

- 开始时只有P_1, P_1被选中，执行时间20
- P_1执行完后，P_2、P_3、P_4的响应比分别为
  $1 + 15/15$、$1 + 10/5$、$1 + 5/10$
  ，P_3被选中，执行时间5
- P_3执行完后，P_2、P_4的响应比分别为
  $1 + 20/15$、$1 + 10/10$
  ，P_2被选中，执行时间15
- P_2执行完后，P_4执行，执行时间10



P_1:20,1+0/20

P_2: 15,1+15/15
P_3: 5, 1+10/5
P_4: 10, 1+5/10

P_2: 15,1+20/15
P_4: 10, 1+10/10

P_4: 10, 1+25/10

Example 4 Scheduling in Real-time Systems

- The task or process deadline in real time systems
  - the time limit before which the task/process should be selected by scheduler, run on CPU and complete processing, in order to reacts to and handles the external event in time

Events	Critical process	Arrival Time	burst time	deadline	priority number
1	P_1	0.0	4.0	7.00	3
2	P_2	3.0	2.0	5.50	1
3	P_3	4.0	2.0	12.01	4
4	P_4	6.0	4.0	11.00	2

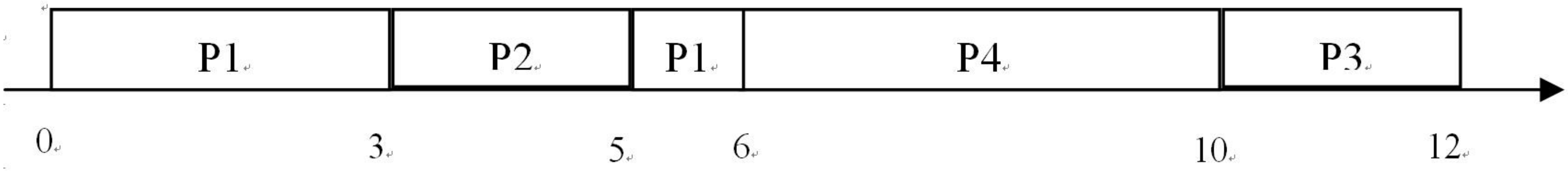
Note: a small priority number means higher priority

Example 4

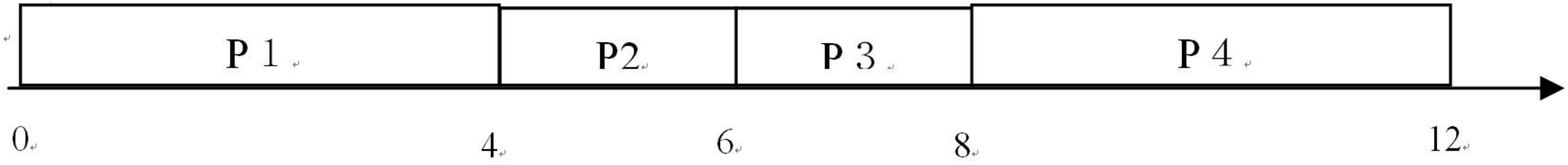
- Suppose that FCFS scheduling is employed,
  - give the Gantt charts illustrating the execution of these processes
  - give the turnaround time and waiting time of each process
  - give the average turnaround time and the waiting time
  - — which event will be treated with in time (i.e. which event will not finish after its deadline) ? and which not ?

- Suppose that priority-based preemptive scheduling is employed
  - give the Gantt charts illustrating the execution of these processes
  - give the turnaround time and waiting time of each process
  - give the average turnaround time and the waiting time
  - — which event will be treated with in time? and which not ?

- Priority-based preemptive
  - all processes are treated with in time



- FCFS
  - e1, e3 are treated with in time



deadline到达时，如果进程没有结束，应继续执行，不会被强行终止

!!
各章典型例题第5章v2

Example 5 Job scheduling + process scheduling

- Job scheduling + process scheduling /长期调度 + 短期调度
- In a computer system, the users submit to the system their computational tasks as jobs, and all these jobs are then stored as the standby jobs on the disk
- The job scheduler (also known as long-term scheduler) selects the standby jobs on the disk, creates new processes in memory for them, and then starts executing these processes. Each job’s ID is the same as that of the process created for it, for example, J_i and P_i
- When the number of concurrent processes in memory is lower than three, the job scheduler takes the FCFS algorithm to select a standby job on the disk to create a new process. Otherwise, the processes should wait on the disk.
- For the processes in memory, the process scheduler (also known as short-term scheduler) takes the non-preemptive priority-based algorithm to select a process and allocates the CPU to it

05 Example 5

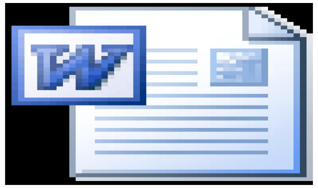
- It is assumed the system costs resulting from job and process scheduling are omitted
- Consider the following set of Jobs J_1, J_2, J_3 , J_4 and J_5. For $1 \leqslant i \leqslant 5$, the arrival time of each J_i , the length of the CPU burst time of each process P_i, and the priority number for each J_i/P_i are given as below, and a smaller priority number implies a higher priority

Job	Arrival Time	Burst Time (minute)	Priority Number
J_1	14:00	40	4
J_2	14:20	30.01	2
J_3	14:30	50.01	3
J_4	14:50	20.01	5
J_5	15:05	10.01	5

05 Example 5

- Illustrate the execution of each job/process by charts.
- What is the turnaround time of each job?
- What is the waiting time of each job?

Note: The waiting time of a job includes the time it waits on the disk and that it waits in memory



!!

各章典型例题第5章-v2

tournaround time

J_1 : $T_1 = 40$ (min)

J_2 : $T_2 = 20 + 30.01$
$= 50.01$ (min)

J_3 : $T_3 = 40.01 + 50.01$
$= 90.02$ (min)

J_4 : $T_4 = 70.02 + 20.01$
$= 90.03$ (min)

J_5 : $T_5 = 5.01 + 70.02 +$
$10.01 = 85.04$ (min)

waiting time

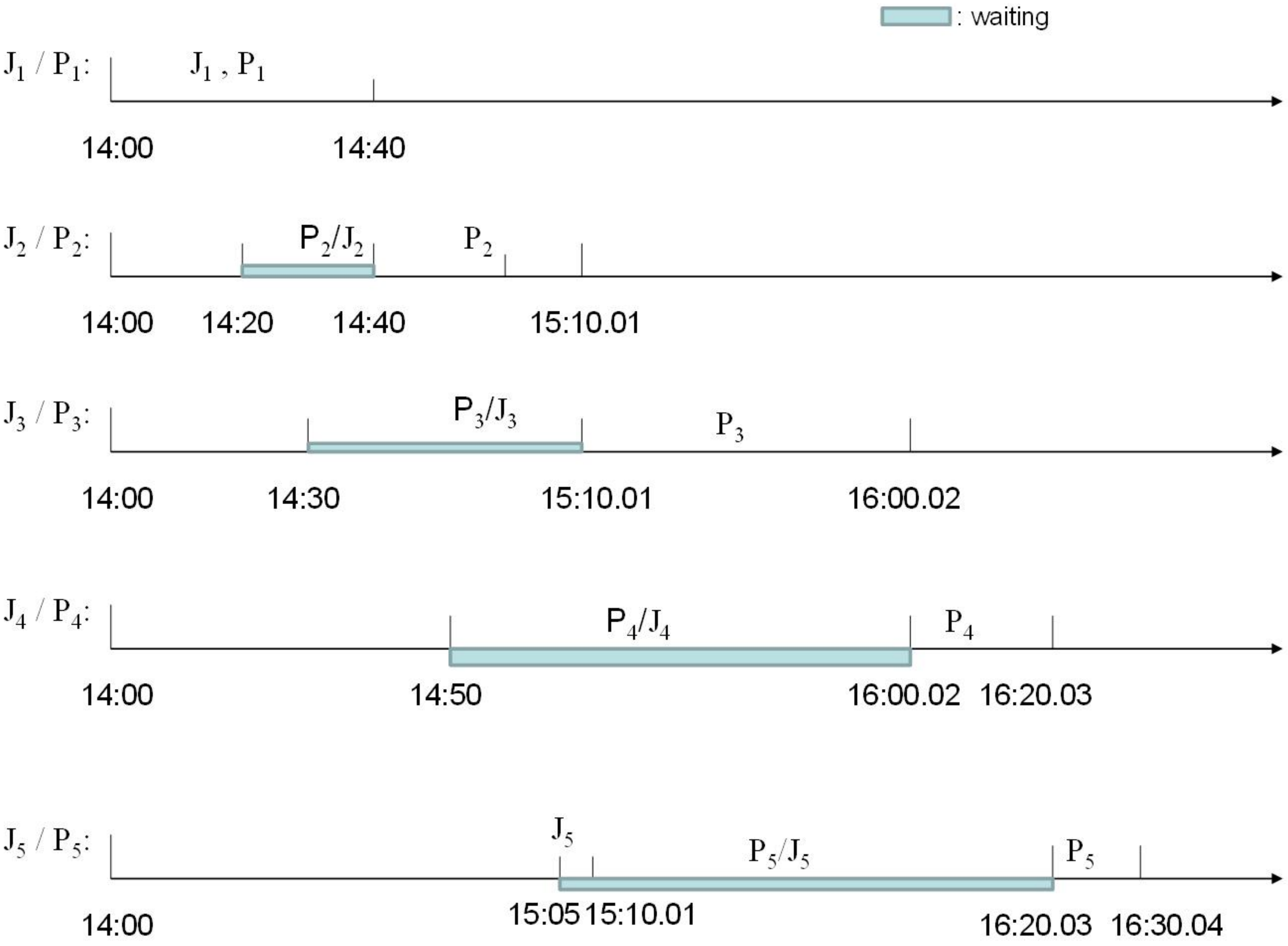
J_1 : $W_1 = 0$ (min)

J_2 : $W_2 = 20$ (min)

J_3 : $W_3 = 40.01$ (min)

J_4 : $W_4 = 70.02$ (min)

J_5 : $W_5 = 75.03$ (min)



Example 6 作业调度+进程调度+对外设竞争+对主存竞争

- 某多道程序设计系统供用户使用的主存为100K，磁带机2台，打印机1台。采用可变分区内存管理，采用静态方式分配外围设备（以先申请先满足、非抢占方式分配磁带机、打印机），忽略作业I/O时间。现有作业序列如下：

作业号	进入输入井时间	运行时间	主存需求量	磁带需求量	打印机需求量
1	8:00	25分钟	15K	1	1
2	8:20	10分钟	30K	0	1
3	8:20	20分钟	60K	1	0
4	8:30	20分钟	20K	1	0
5	8:35	15分钟	10K	1	1

Example 6

- 作业调度采用FCFS策略，优先分配主存低地址区且不准移动已在主存的作业，在主存中的各作业平分CPU时间（时间片轮转法）。现求
  - 作业被调度的先后顺序
  - 全部作业运行结束的时间
  - 作业平均周转时间
  - 最大作业周转时间

!!

各章典型例题第5章-v2

- 注：本题综合了作业调度、进程调度、对外设竞争、对主存竞争

- 作业调度选择的作业顺序为：作业1、作业3、作业4、作业2、作业5
- 全部作业运行结束的时间为9:30
- 周转时间：

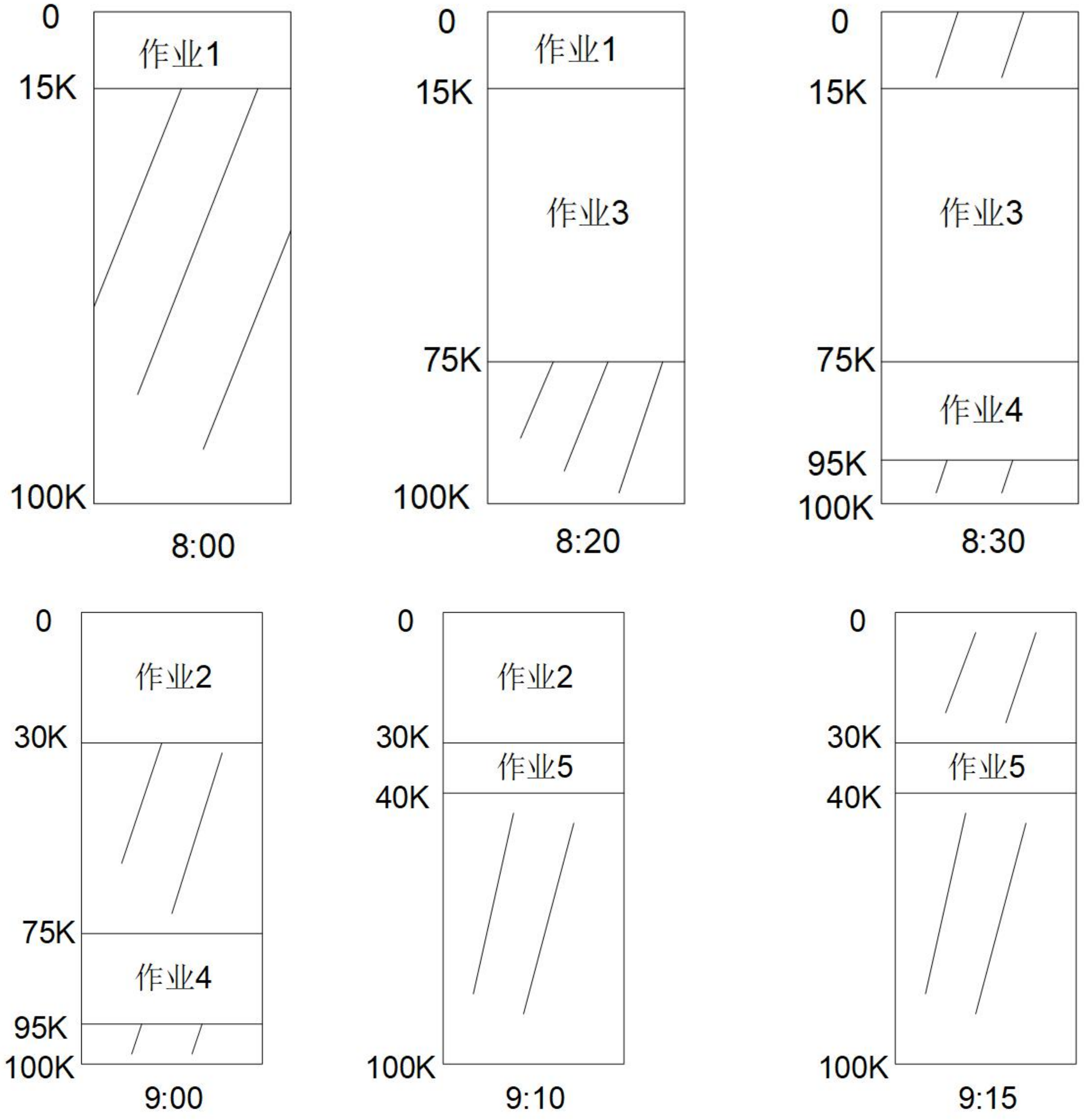
  作业1为30分钟，

  作业2为55分钟，

  作业3为40分钟，

  作业4为40分钟，

  作业5为55分钟
- 平均作业周转时间为44分钟
- 最大作业周转时间为55分钟



Example 7 时间片长度 vs CPU利用率

- 11个并发tasks/processes
  - 10个I/O bound tasks，由“1ms CPU burst + 10ms I/O burst”组成
  - 1个CPU bound task，完全由CPU burst组成

- 分别考察时间片=1ms、10ms时，各自的CPU利用率

I/O bound tasks t1~t10:

CPU burst　　I/O burst

CPU bound task t11:

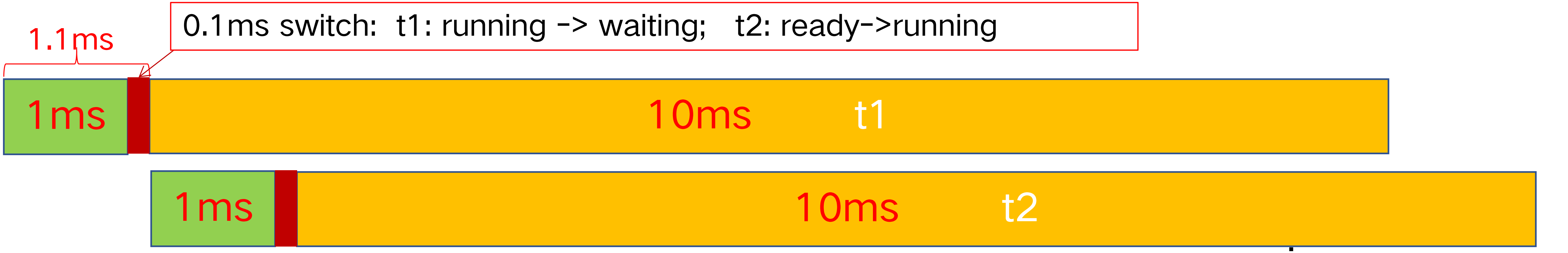
Long-run task：

1. I/O bound task是由多个“1ms CPU burst + 10ms I/O burst”组成，生命周期/周转时间/执行时间远超1个时间片
2. CPU bound task的执行时间超过1个时间片

考察11个进程t1~t11被调度一轮（每个进程获得一个时间片运行）后，用户态下执行时间占比

时间片=1ms

考察11个任务并发地被调度执行一次:
对每隔1ms，进行一次耗时0.1ms的context切换，即在(1+0.1)ms中有1ms在用户态下运行

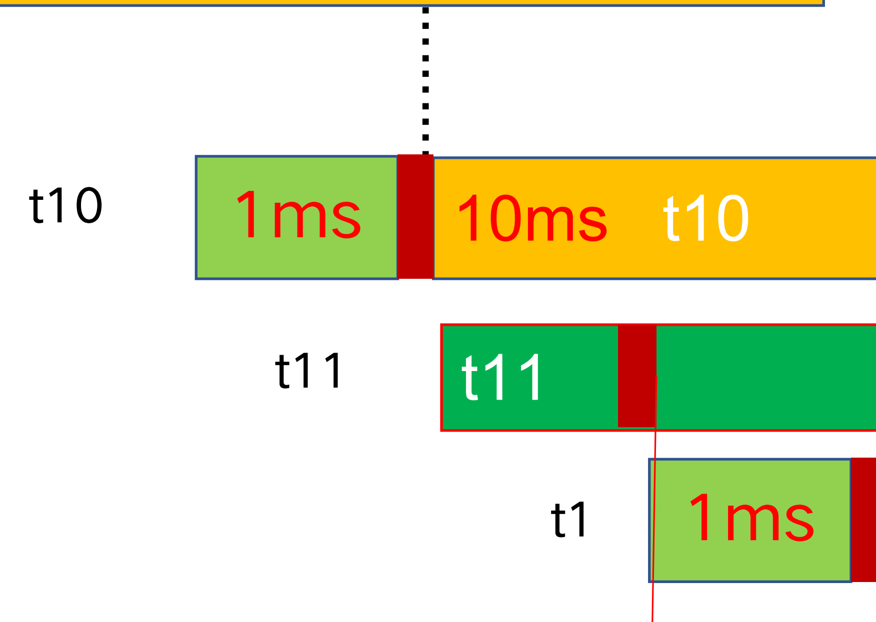


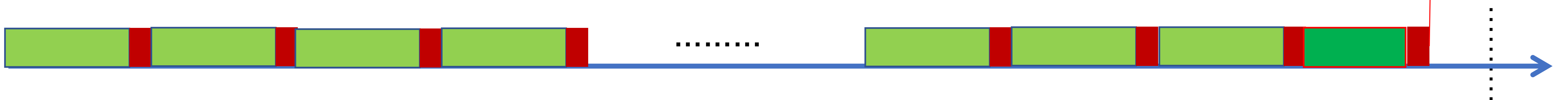
CPU 利用率

$$= \frac{\text{用户态执行时间}}{\text{用户态执行时间}+\text{内核态switch时间}}$$

$$= \frac{1ms\,\text{时间片}\times 11}{(1ms\,\text{时间片}+0.1ms\ \text{switch时间})\times 11}$$

$$= \frac{1ms\times 11}{(1ms+0.1ms)\times 11} = 90.91\%$$

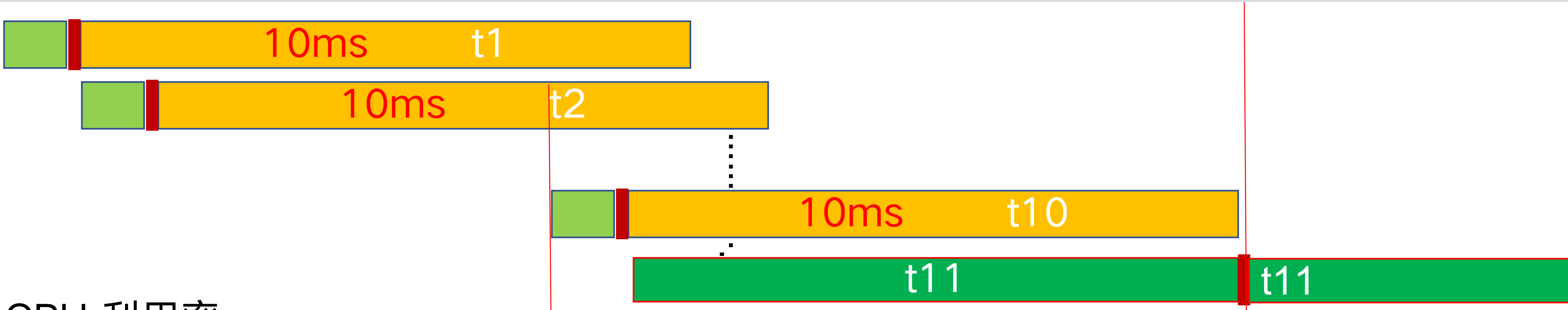


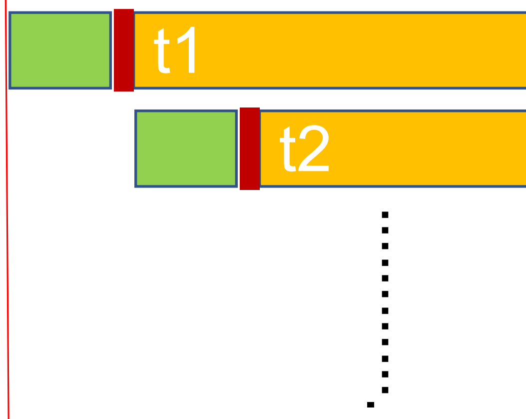


时间片=10ms，考察11个任务并发地被调度执行一次

对t1~t10，每隔1ms!!!(进入wating态I/O操作)，进行一次耗时0.1ms的context切换

对t11，每隔10ms，进行一次耗时0.1ms的context switch





CPU 利用率

$$= \frac{\text{用户态下执行时间}}{\text{用户态执行时间}+\text{内核态switch时间}}$$

$$= \frac{\text{t1~t10的10个CPU burst} + \text{分配给t11的1个时间片}}{\text{用户态下执行时间}+\text{t1~t11的switch次数(11次)}*0.1ms}$$

$$= \frac{1ms*10+10ms}{(1+0.1)ms*10+(10ms+0.1ms)} = \frac{20ms}{21.1ms} = 94.79\%$$

Example 8. （反例）response time分析

- 在交互式系统中，若用户数为10，为保证**响应时间？** ≤100ms，忽略其他系统开销，则操作系统应将时间片设为

  A．≤1ms　　　　B．≤10ms

  C．≤100ms　　　D．≤1000ms

  - 答案：B?

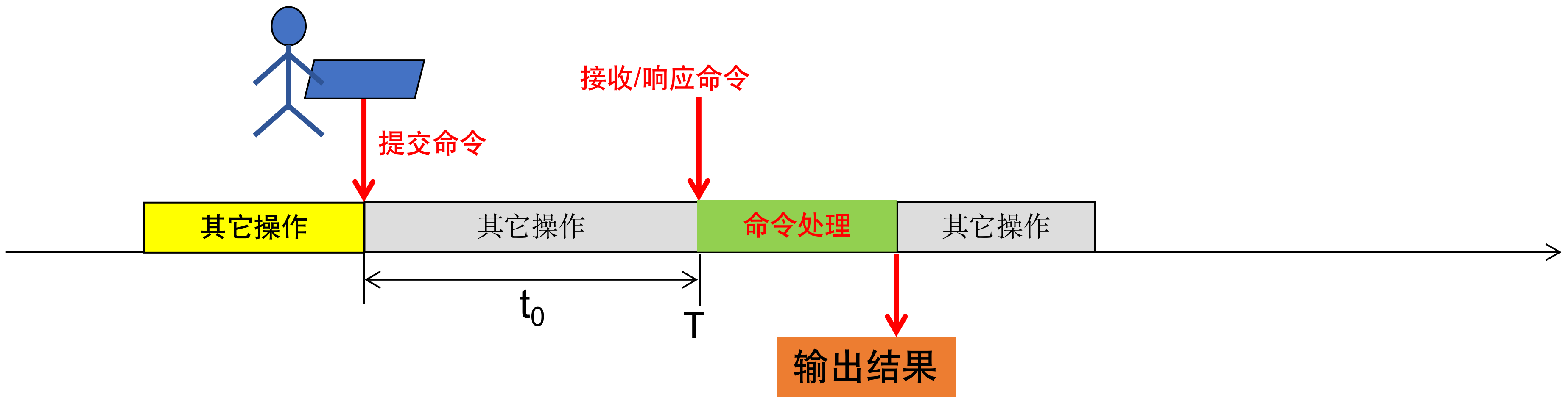
- 答案不严谨
  - 前提条件不足，对概念“响应时间”理解有误

08 Example 8

- 进程响应时间取决于2个因素
  - ■ 单进程P_0（进程独占CPU）情况下，进程自身的业务处理逻辑

—e.g. 用户通过键盘，敲入控制命令，进程在T之后接收、响应命令，

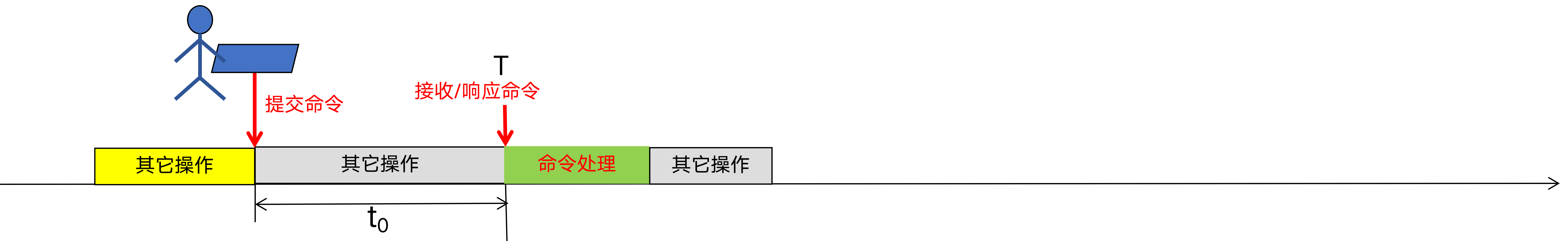
响应时间$t_0 \geq 0$



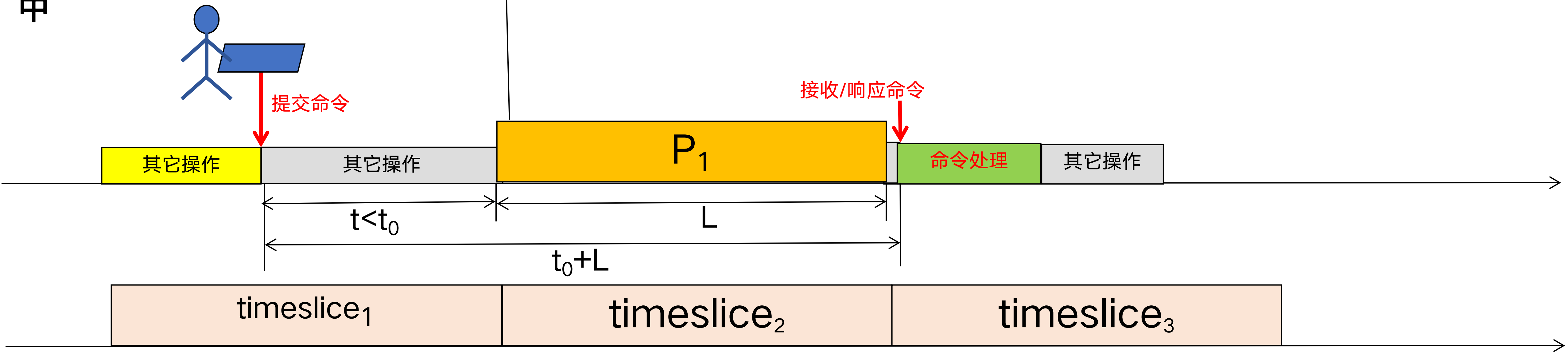
■ 多进程并发环境下，OS调度机制、其它进程执行情况

—假设：n=2个进程: P_0、P_1，分时调度，时间片长度为L，

(1) 单进程P_0独占CPU，响应时间t_0

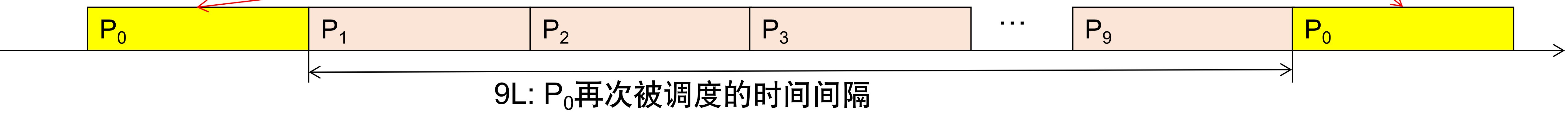


(2) P_0、 P_1分时并发，响应时间t_0+L，**用户提交请求与P_0相应请求分别在2个不同时间片中**

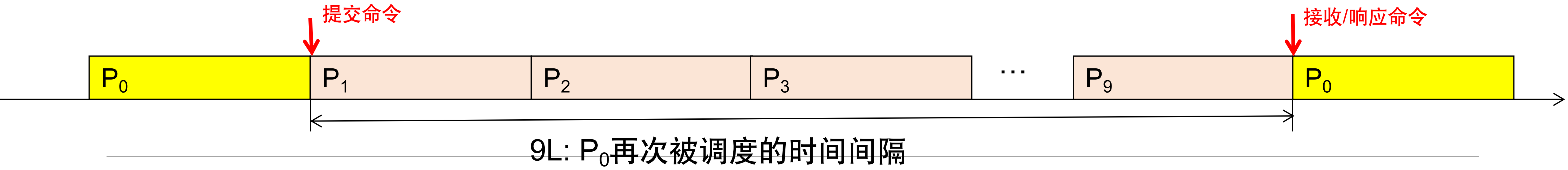


08 反例答案分析

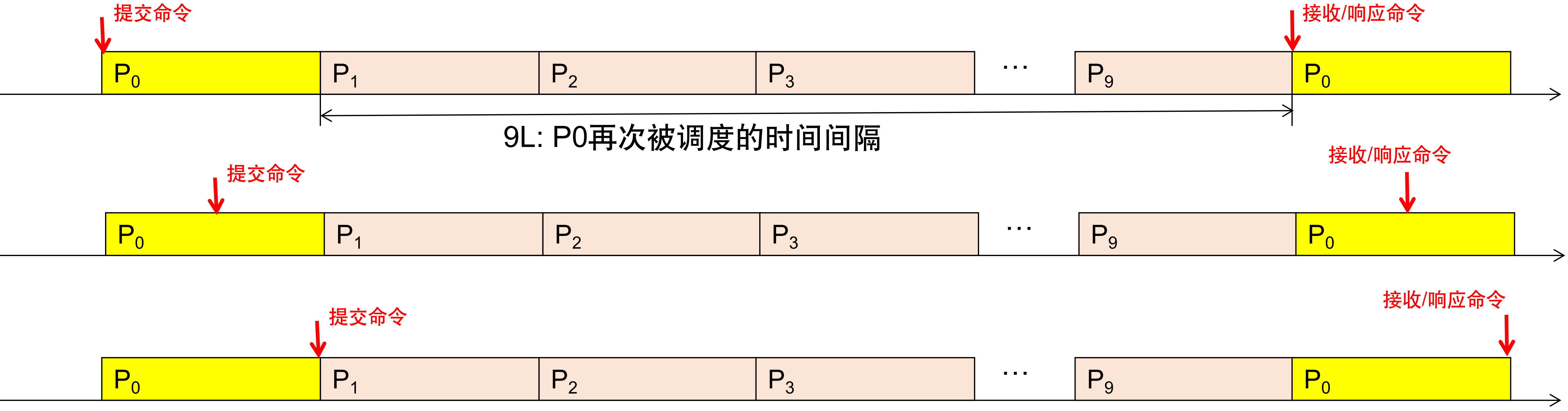
- 并发进程数n=10，要求P_0的响应时间≤100ms，时间片长度L ≤10ms ？
  - 与t_0有关
  - 假设：**用户提交请求与P_0响应请求跨时间片，分别在分配给P_0的2个不同时间片中完成，即0≤t_0 ≤ 2L**



- 情况1：t_0=0，响应时间=9L，9L≤100ms, L ≤100/9ms

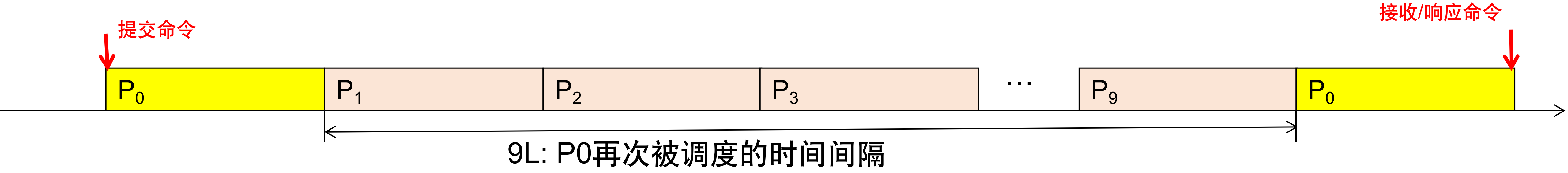


- 情况2：t0=L， 响应时间= t0+9L， 10L≤100ms, L ≤100/10ms=10ms



08 分析

- 情况3：$t_0=2L$，响应时间$=t_0+9L=11L$，
  $11L \le 100ms$, $L \le 100/11ms$



结论：

本题不严谨，前提条件不全，

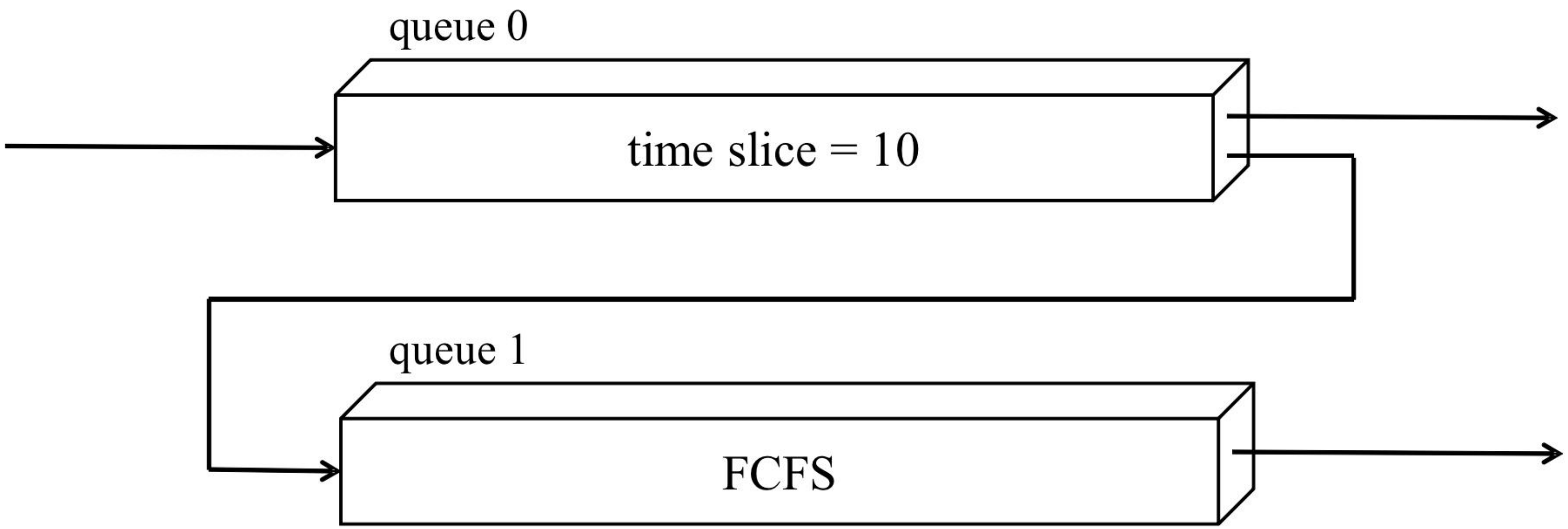
应给出当P_0独占CPU运行时，P_0的响应时间t_0

作业 1 2 CPU、3进程调度

- 在1个在双CPU系统中（不支持超线程HT）， 3个并发进程的执行序列（CPU burst, I/O burst）如下
  - P1: computing, 80ms→I/O operation, 100ms→computing, 40ms
  - P2: computing, 180ms→I/O operation, 70ms→computing, 20ms
  - P3: computing, 130ms→I/O operation, 50ms→computing, 50ms
- 假设
  - 1. 在这3个进程中，P1、P2的I/O操作均为打印机输出操作，且只有1台打印机
  - 2. P3的I/O操作为磁盘访问操作
  - 3个进程的CPU burst可以任意分配到2个CPU上执行
- 若不考虑调度和切换时间，合理地安排这3个进程的执行步骤，使得系统总吞吐量（throughput）最大，要求：
  - 1. 利用甘特图描述这3个进程在2个CPU上执行轨迹；
  - 2. 计算系统最大吞吐量和3个进程的平均周转时间

作业 2 两级队列反馈调度

- As shown below, OS takes a two-level feedback-queue scheme to allocate CPU for concurrent processes. A process entering the system is at first put in queue 0, and sequentially given a CPU time slice of 10 milliseconds. If it does not finish within this time, it is moved to the tail of queue 1. Processes in queue 1 run on FCFS scheduling, but are permitted to run only when there is no process in queue 0. When a process P_i in queue 1 is running on CPU and a new process P_j enters the system, P_j will preempt the CPU occupied by P_i.



- Consider the processes P0, P1, P2, P3. For 0⩽ i ⩽3, the arrival time, the length of the CPU burst time, and the priority of each Pi are given as below

Process	Arrival time	Burst Time	Priority
P_0	0.0	6.0	10
P_1	4.0	15.0	8
P_2	8.0	4.0	6
P_3	12.0	13.0	4

- For the snapshot shown above, suppose that two-level feedback-queue scheduling is employed
  - (1) Draw the Gantt chart that illustrates the execution of these processes.
  - (2) What are the turnaround times for the four processes?