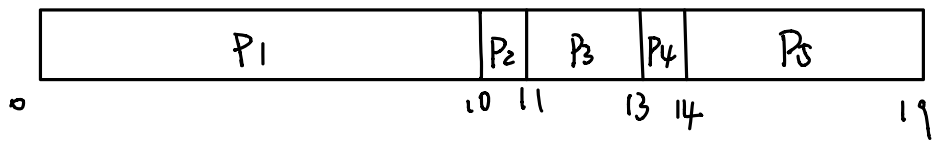
5.4 Consider the following set of process, with the length of the CPU-burst time given in milliseconds:

Process	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

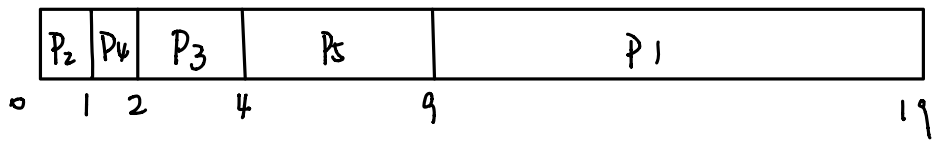
The processes are assumed to have arrived in the order P1,P2,P3,P4,P5, all at time 0.

a. Draw four Giant charts illustrating the execution of these process using FCFS, SJF, a nonpreemptive priority(a smaller priority number implies a higher priority), and RR (quantum=1) scheduling,

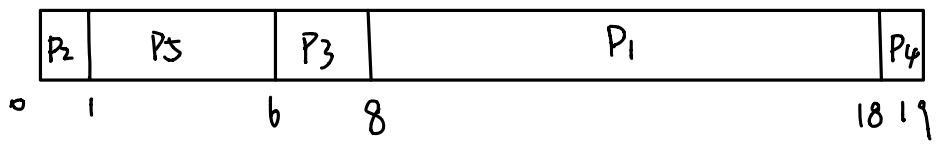
FCFS



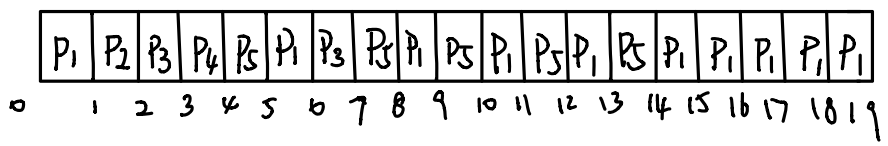
SJF



A nonpreemptive priority



RR



b. What is the turnaround time of each process for each of the scheduling algorithms in part a?

FCFS : $P_1 = 10$ $P_2 = 11$ $P_3 = 13$ $P_4 = 14$ $P_5 = 19$

SJF : $P_1 = 19$ $P_2 = 1$ $P_3 = 4$ $P_4 = 2$ $P_5 = 9$

A nonpreemptive priority : $P_1 = 18$ $P_2 = 1$ $P_3 = 8$ $P_4 = 19$ $P_5 = 6$

RR : $P_1 = 19$ $P_2 = 2$ $P_3 = 7$ $P_4 = 4$ $P_5 = 14$

c. What is the waiting time of each process for each of the scheduling algorithms in part a?

FCFS: $P_1 = 0$ $P_2 = 10$ $P_3 = 11$ $P_4 = 13$ $P_5 = 14$

SJF: $P_1 = 9$ $P_2 = 0$ $P_3 = 2$ $P_4 = 1$ $P_5 = 4$

A nonpreemptive priority: $P_1 = 8$ $P_2 = 0$ $P_3 = 6$ $P_4 = 18$ $P_5 = 1$

RR: $P_1 = 9$ $P_2 = 1$ $P_3 = 5$ $P_4 = 3$ $P_5 = 9$

d. Which of the schedules in part a results in the minimal average waiting time (over all processes) ?

FCFS: $(0+10+11+13+14) \div 5 = 38 \div 5 = 9.6$

SJF: $(9+0+2+1+4) \div 5 = 16 \div 5 = 3.2$

A nonpreemptive priority: $(8+0+6+18+1) \div 5 = 33 \div 5 = 6.6$

RR: $(9+1+5+3+9) \div 5 = 27 \div 5 = 5.4$

So. SJF has the minimal average waiting time.

5.6 Consider a variant of the RR scheduling algorithm where the entries in the ready queue are pointers to the PCBs.

a. What would be the effect of putting two pointers to the same process in the ready queue?

The process will increase its priority by getting time more often it is receiving preferential treatment.

b. What would be the major advantages and disadvantages of this scheme?

Advantage: jobs which are more important will get more time in treatment.
Disadvantage: shorter jobs will suffer.

c. How would you modify the basic RR algorithm to achieve the same effect without the duplicate pointers?

We can have more quantums possible in the RR scheme.