**CS307 Operating System Exercises Spring 2020**

**Exercise 1: CPU Scheduling**

The following rules apply to the exercises:

\* We assume that when a preempted process and new coming process need to be put in the

ready queue at the same time, we prefer to put the new coming process in front of the

preempted process in the ready queue.

\* When processes are of same priority, use FCFS to determine the order.

1. Consider the following set of processes, with the arrival times and the length of the

CPU‐burst times given in milliseconds, and the priorities given so that smaller priority

number means a higher priority.

Process Arrival time Burst time Priority

P1 0 10 3

P2 2 2 1

P3 3 2 3

P4 4 1 4

P5 6 5 2

a) Draw Gantt charts illustrating the execution of these processes using the following

scheduling algorithm: (1) FCFS, (2) nonpreemptive SJF, (3) preemptive SJF, (4)

nonpreemptive priority, (5) preemptive priority, (6) RR (quantum = 1), and (7) RR

(quantum = 3). We specify that when a preempted process and new coming process

need to be put in the ready queue at the same time, we prefer to put the new coming

process in front of the preempted process in the ready queue. We also specify that

when processes are of the same priority, use FCFS to determine the order.

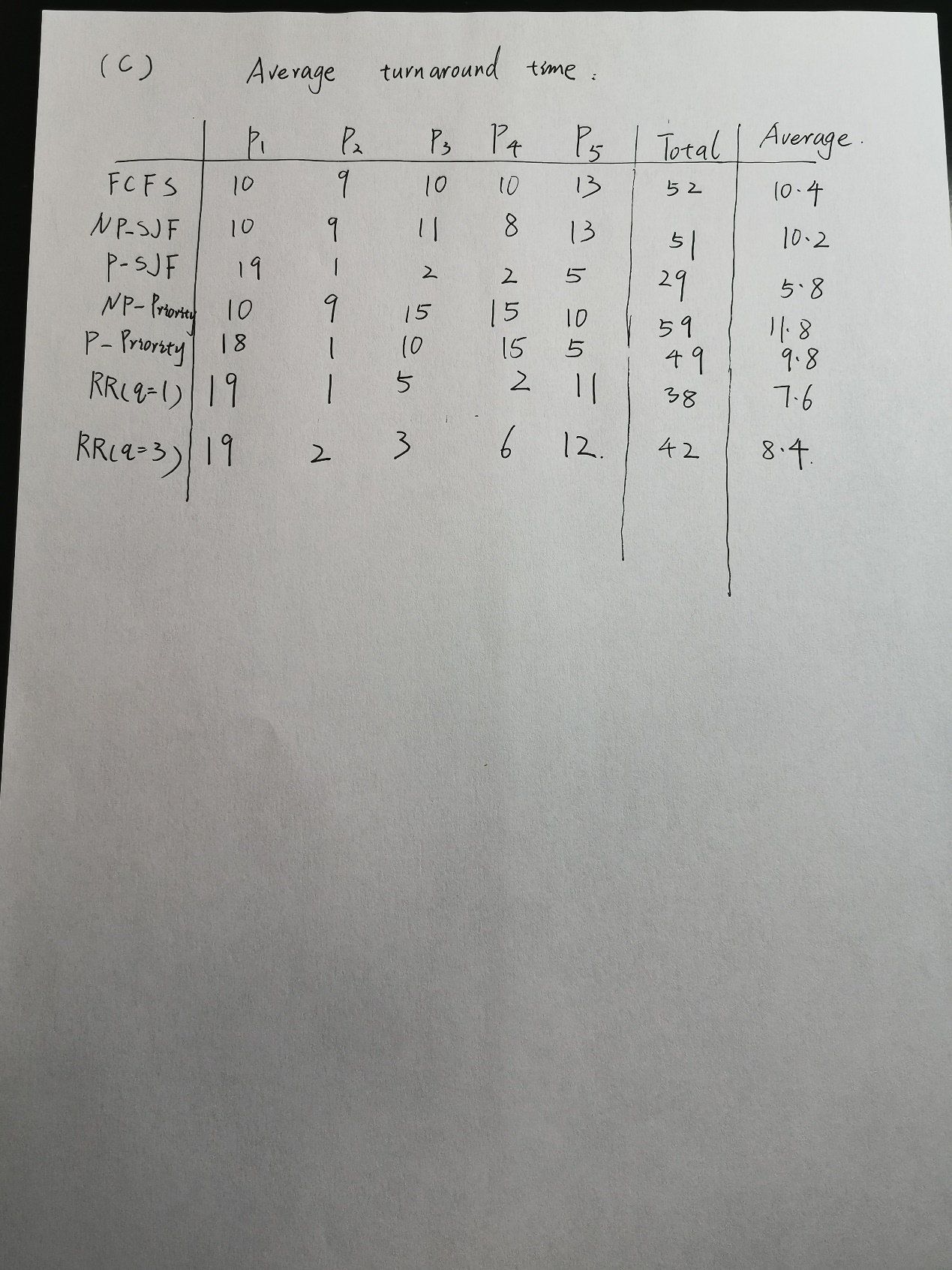
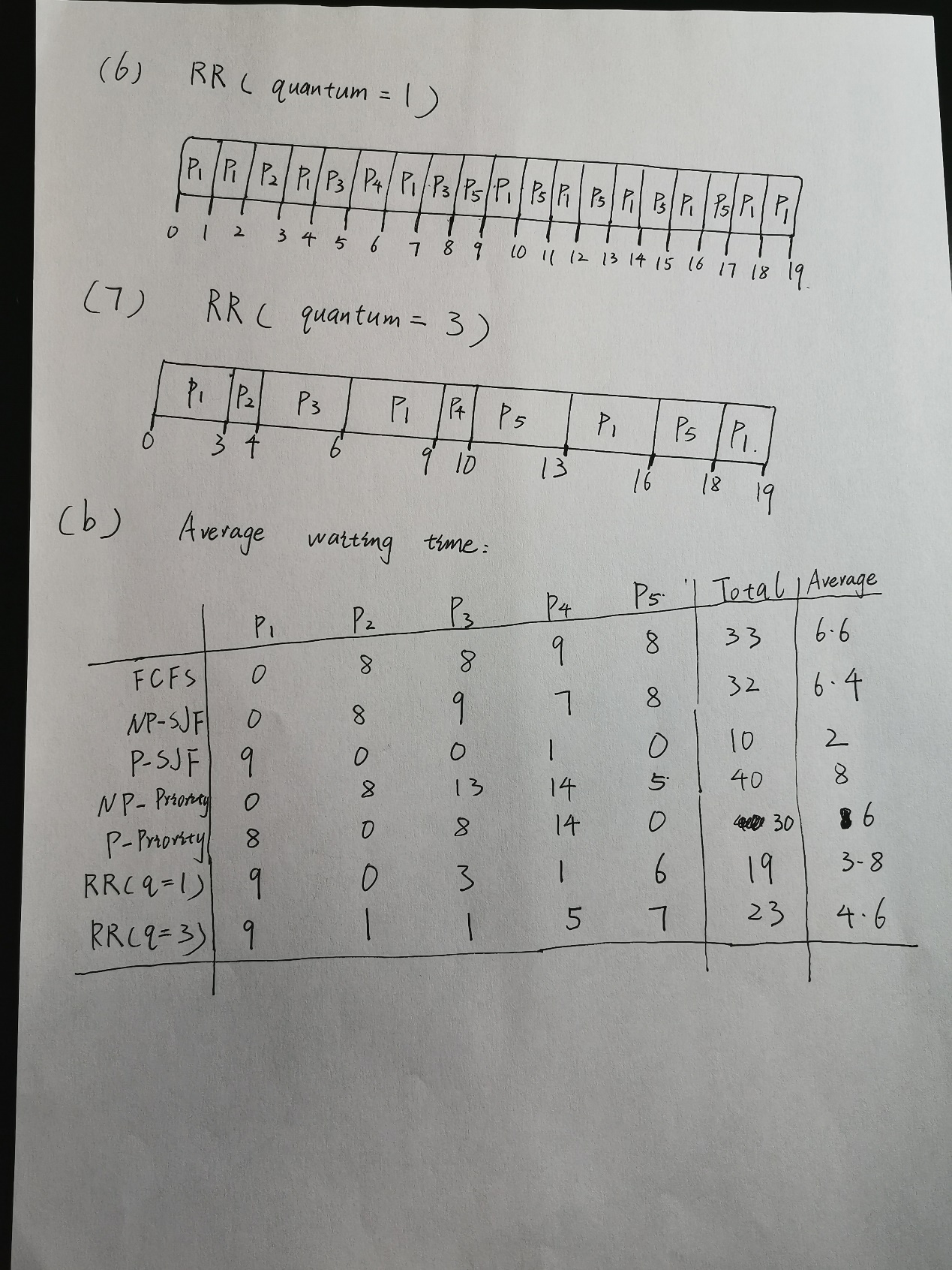
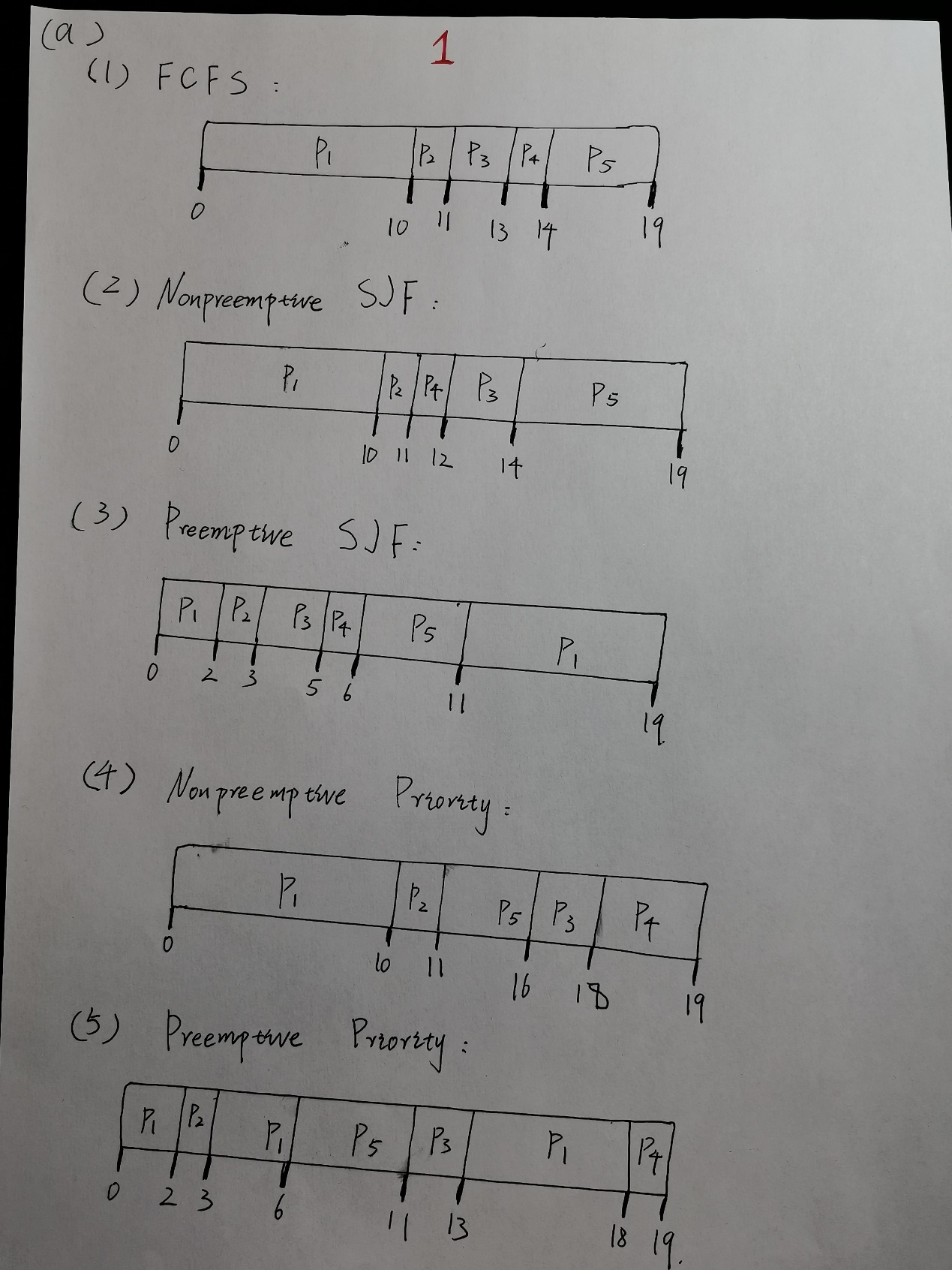
b) Calculate the average waiting time when using each of the above scheduling

algorithms.

c) Calculate the average turnaround time when using each of the above scheduling

algorithms.

**Answer:**

****

2. Consider the following set of processes, with the arrival times and the length of the

CPU‐burst times given in milliseconds.

Process Arrival time Burst time

P1 0 17

P2 12 25

P3 28 8

P4 36 32

P5 46 18

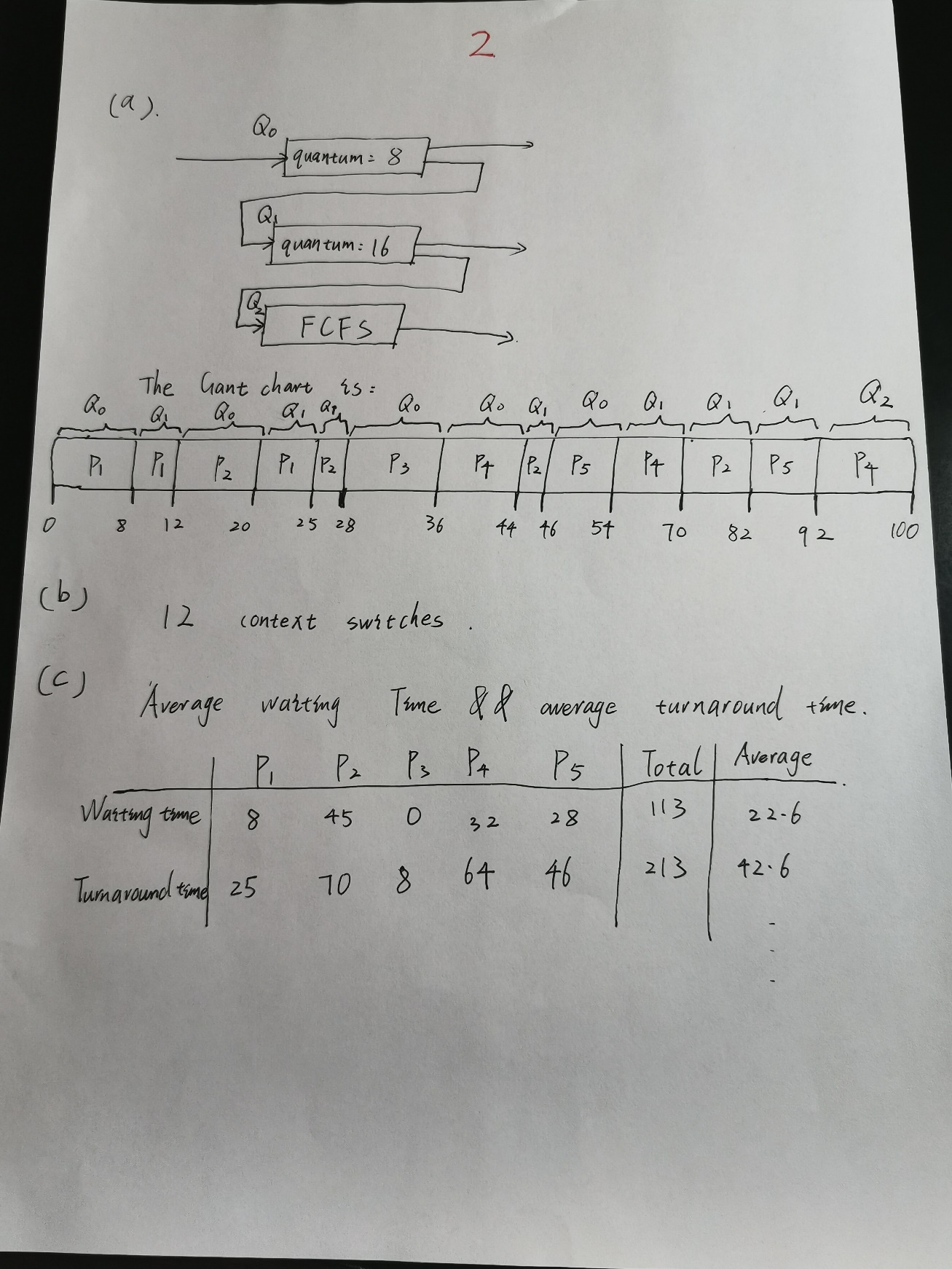
a) Draw a Gantt chart illustrating the execution of these processes using a Multilevel

Feedback Queue Scheduling. Use the same structure of the model as in the lectures:

(queue 0: quantum 8), (queue 1: quantum 16), (queue 2: FCFS).

b) Calculate the number of context switches for the processes.

c) Calculate the average waiting time and the average turnaround time for the scheduling.



3. Which of the following scheduling algorithms could result in starvation, and why?

a. First‐come, first‐served

b. Shortest job first

c. Round robin

d. Priority

**Answer:** b d

Reason: In shortest job first case, if a process has a long CPU burst time, it’s priority is very low and it might never execute, so this case can cause starvation. In priority scheduling, for the same reason, low priority processes may never execute if every time before it execute and there is a process holds a higher priority lining up in front of it. So this case can cause starvation, too.