1. Consider the set of processes with arrival time (in milliseconds), CPU burst time (in milliseconds), and priority shown below: (Higher number represents lower priority)

Process ID	Arrival Time	Burst Time	Priority
P1	0	4	4
P2	1	3	3
P3	2	1	5
P4	3	5	1
P5	4	2	2

By giving the gantt chart in each case, calculate the throughput, average response time, average waiting time, average turn around time, in the following CPU sheduling policies

      i) Preemptive Shortest Job First

      ii) Preemptive Priority

      iii) Round Robin with quantum time 2ms

      iv) First Come First Serve

2. Imagine a hypothetical barbershop with one barber, one barber chair, and a waiting room with n chairs (n may be 0) for waiting customers. The following rules apply:

- If there are no customers, the barber falls asleep in the chair

- A customer must wake the barber if he is asleep

- If a customer arrives while the barber is working, the customer leaves if all chairs are occupied and sits in an empty chair if it's available

- When the barber finishes a haircut, he inspects the waiting room to see if there are any waiting customers and falls asleep if there is none.

There are two main complications. First, there is a risk that a race condition, where the barber sleeps while a customer waits for the barber to get him/her for a haircut arises because all of the actions—checking the waiting room, entering the shop, taking a waiting room chair—take a certain amount of time. Specifically, a customer may arrive to find the barber cutting another customer hair so he/she returns to the waiting room to take a seat but while walking back to the waiting room the barber finishes the haircut and goes to the waiting room, which he finds empty (because the customer walks slowly or went to the restroom) and thus goes to sleep in the barber chair. Second, another problem may occur when two customers arrive at the same time when there is only one empty seat in the waiting room and both try to sit in the single chair; only the first person to get to the chair will be able to sit.

Propose a solution to this problem that coordinates the actions of the customers and the barber in a way that avoids

3. Consider a system with five processes (P1 to P5) and three resource types (A, B, and C). The maximum resource requirements and current allocation for each process are provided in the table below:

Process	Max(A, B, C)	Allocation(A, B, C)
P1	(7, 5, 3)	(0, 1, 0)
P2	(3, 2, 2)	(2, 0, 0)
P3	(9, 0, 2)	(3, 0, 2)
P4	(2, 2, 2)	(2, 1, 1)
P5	(4, 3, 3)	(0, 0, 2)

The system currently has available resources: (3, 3, 2).

i)   Apply the Banker's algorithm to determine if the system is in a safe state. Show the sequence of resource allocation and deallocation that satisfies all processes' resource requirements without leading to deadlock.

ii)   If the system is in a safe state, determine the safe sequence of processes.

iii)   If the system is not in a safe state, explain why it is unsafe and identify the processes causing the deadlock. How will you efficiently recover from the deadlock

4. Linux Fundamental

1. Create a directory named "exam" in your home directory. Inside this directory, create three subdirectories named "task1", "task2", and "task3". Create an empty text file named "notes.txt" inside "task1" directory.

   mkdir ~/exam
   cd ~/exam
   mkdir task1 task2 task3
   touch task1/notes.txt

2. In the "task2" directory, create a file named "numbers.txt" containing numbers from 1 to 10, each on a new line. Append the numbers from 11 to 20 to the same file without overwriting the existing content.

   cd ~/exam/task2
   seq 1 10 > numbers.txt
   seq 11 20 >> numbers.txt

3. Set the permissions of "notes.txt" in "task1" directory to be readable, writable, and executable by the owner only, and readable by the group and others.

   chmod 744 ~/exam/task1/notes.txt

4. Inside the "task3" directory, create a file named "data.txt" containing the text **"This is a sample text containing Linux.".** Search for the word "Linux" in the file and replace it with "Open Source". Save the changes.

cd ~/exam/task3

echo "This is a sample text containing Linux." > data.txt

sed -i 's/Linux/Open Source/g' data.txt