Project 2 The Sleeping Barber Problem

These are the instructions for your first Operating Systems programming project. The project can be done by group of up to 2 students. You could submit only one copy with two names.

Problem: A barbershop consists of a single barber, a single barber chair, and a waiting room with a limited number of chairs for waiting customers.

When there are no customers, the barber sleeps.
When a customer arrives:

- If the barber is sleeping, the customer wakes the barber.
- If the barber is cutting hair and there is an available chair, the customer sits and waits.
- If all chairs are full, the customer leaves the shop.

Your task is to write a program to synchronize the barber and customers using C (POSIX threads and semaphores) or Java (thread synchronization with wait()/notify()).

Requirements:

- Create:
  - A Barber class/thread (one thread).
  - A Customer class/thread (multiple threads).
  - Shared variables or a monitor to represent the waiting room and the barber chair.
- Use appropriate synchronization to ensure:
  - No race conditions
  - Proper sleeping/waking behavior
  - No customers get served at the same time
- Simulate multiple customer arrivals at random intervals.
- Print messages showing each event, e.g.:
  - "Customer 3 enters the shop."
  - "Customer 3 is waiting."
  - "Barber is cutting Customer 2's hair."
  - "Customer 4 leaves (no chairs available)."
  - "Barber goes to sleep."
- Each customer should visit the barber at least once, and the barber should handle several customers before the simulation ends.
- **Finally you need to analysis to see if deadlock occurs? If yes, under what condition?**

What to hand in?

- A report includes:
  - problem statement
  - Analysis
  - algorithm design
  - class prototype (class contract)

    - program Input/Output
    - tested results (analysis your result to see if it is correct).
- The source program (name your file “project2.c” or “project2.java”)
- You could submit them on Canvas.

Each program should have the following in the beginning:

// Program Name: <The name of your program file>
// Programmer: <Your name here>, <ID>
// Assignment Number: <put project number here, e.g. Project #2>
// Purpose: <A short problem description>