Hints for Sushi Bar

- What is the maximum possible number of seats in the bar? Is it feasible to store the state (occupied/not occupied) of every seat in the bar?
- We are interested in allocating an unoccupied seat to a new customer.
  - We can store **ranges** of contiguous unoccupied seats. This allows us to store the unoccupied seats available efficiently.
  - Recall that when a new customer arrives, we are required to allocate the seat that maximises the minimum distance to the adjacent occupied seat(s).
  - What attributes do we need to store for each range of unoccupied seats?
  - How should we order the ranges of unoccupied seats?
  - What data structure should we use to store the ranges of unoccupied seats?
  - Using the data structure that you have chosen, can you efficiently find the range that contains the required seat?
- After we have found a seat to allocate to the new customer, what happens to the range that contains the allocated seat?
- Customers who have previously entered the bar can leave the bar at any time afterwards.
  - When a customer leaves the bar, what happens to the ranges adjacent to the seat that the customer was occupying?
  - What updates do we need to perform on our data structure?