Allocating Hotel Rooms

Task

Allocate customers to hotel rooms based on their arrival and departure days. Each customer wants their own room, so two customers can stay in the same room only if the departure day of the first customer is earlier than the arrival day of the second customer. The number of rooms used should be minimized.

Input

A list or array of **n** customers, 1 ≤ n ≤ 1000. Each customer is represented by (arrival_day, departure_day), which are positive integers satisfying arrival_day ≤ departure_day.

Output

A list or array of size **n**, where element i indicates the room that customer i was allocated. Rooms are numbered **1,2, …, k** for some **1 ≤ k ≤ n**. Any allocation that minimizes the number of rooms **k** is a valid solution.

Example:

Suppose customers is [(1,5), (2,4), (6,8), (7,7)].
Clearly customers 1 and 2 cannot get the same room. Customer 3 can use the same room as either of them, because they both leave before customer 3 arrives. Then customer 4 can be given the other room.
So any of [1,2,1,2], [1,2,2,1], [2,1,2,1], [2,1,1,2] is a valid solution.

NOTE: The list of customers is not necessarily ordered by arrival_time.

Also note that your code should not modify the list of customers.

```
    test_cases = [

               [(1,2),(2,4),(4,4)],          ## [1,2,1] or [2,1,2]

               [(1,5),(2,4),(6,8),(7,7)],   ## any of [1,2,1,2], [1,2,2,1], [2,1,2,1], or [2,1,1,2]

               [(15,22),(2,4),(6,9),(3,33),(12,21)], ## [1,2,2,3,2], [2,1,1,3,1], [3,1,3,2,1], etc

               [(1,10),(2,5),(6,6),(3,7),(6,6),(11,13),(9,15),(8,14)],

                   ## Solutions include:  [1,2,2,3,4,1,3,2], [1,2,2,3,4,1,2,3],

                   ##                     [1,2,4,3,2,1,3,2], [2,3,3,1,4,2,1,3], and others

               [(8,8),(5,8),(8,9),(1,4),(1,3),(5,7),(4,8),(2,2),(4,5),(6,8)]

                   ## Solutions include:  [4, 1, 5, 1, 2, 4, 2, 3, 3, 3]

                   ##                     [5, 4, 2, 2, 1, 2, 3, 3, 1, 1], and others
```