

# Finals Spring 2015 - Conference Room Scheduler (250 points)

## Introduction

Bloomberg needs a new system to schedule conference rooms! To keep things simple, the system operates in 15 minutes blocks, and only during an 8 hour workday, so that there are 32 available time slots per day for each room.

Users will submit one of two different commands: Create **Booking**, and **Query** Available.

When a user attempts to create a booking, they will submit a Room Id, a starting Timeslot, and a length (in timeslots). Each conference room can only be occupied by one user, and is booked in increments of timeslots (a minimum booking is length 1, maximum is length 32). Any user can book any room for as many slots as possible, so long as their booking does not interfere with an already occupied room. If the booking overlaps with any other bookings for that conference room (even if it's only for one slot of many), the entire booking command is rejected (i.e., the room schedule remains unchanged).

A user can also **query availability** to ask which rooms are available during a certain time block (a starting timeslot + a length, in timeslots). The system should report to the user which rooms are available for the entire length of their requested time. If a room is unavailable for any amount of time during the requested window, it is not returned from the Query.

## Input Specifications

Input will be the number of rooms **N** on the first line ( $1 \leq N \leq 100$ ), followed by any number of the following request types:

**Booking:** RoomId-TimeSlot-#OfSlots

**Query:** TimeSlot-#OfSlots

You can assume that **no more** than *100 requests* will be made. Also, RoomIds and TimeSlots are **indexed** starting at 1, not 0.

## Output Specifications

Output as many lines as necessary to answer each request in the order they were received.

**Booking:** You will output Y if the booking is possible and N otherwise.

**Query:** You will output a list of space-delimited room ids in order. There should be no trailing or preceding spaces. If there are no rooms available that match the Query, print None

## Sample Input/Output

### Input

```
3
1-1-4
1-2
```

## Output

Y  
2 3

## Explanation

There are 3 rooms.

Room 1 is booked starting at timeslot 1 and booked for a length of 4 timeslots (1,2,3,4). There are no prior bookings so the output is Y.

User is requesting availability of rooms starting at timeslot 1 for a length of 2 timeslots (1,2). Only room 1 is occupied so 2 3 are printed.