specs Schedule

RoomSchedule

```
Given an array of intervals (start, end) for events (possibly
overlapping), find the minimum number of rooms required for all
events.

The time intervals are represented by pairs of numbers, x, y, where
y is larger than x.  x represents the starting time and y represents
the ending time of the lecture. For example:

[(1,4), (5,8), (7,9)]

Represents 3 events where the second and the third event intersect.
That is, the third event (7,9) starts before the second event (5,8)
ends.

1. Use the starter code provided.  This code reads integers (such as
1 4 5 8 7 9) and creates an array of events (such as [(1,4), (5,8),
(7,9)]).  The starter code provides code to read, create and display
the array of events.  The only thing you need to do is to find the
minimum number of rooms required for all the events.

2. the schedule is entered by the user as a sequence of integers

3. assume the user is always going to enter an even number of ints

4. read the ints from the user and display the set of intervals as
an array as well as graphically (see examples).  The code required
to do this is provided in the starter code.

5. compute what is the minimum number of classrooms required for the
schedule

6. the starter code provides a method that displays a schedule such
that the events are displayed one per line as a sequence of periods.
For example, the schedule

[(1,4), (5,8), (7,9)]

is represented as follows:

....
    ....
      ...
```

Examples:

```
% java RoomSchedule
1 3 5 7 9 11 15 17
[(1,3), (5,7), (9,11), (15,17)]
 ...
     ...
         ...
             ...
minimum nr of rooms: 1
% java RoomSchedule
1 10 2 9 3 8 4 7 5 6
[(1,10), (2,9), (3,8), (4,7), (5,6)]
```

```
 ..........
  ........
   ......
    ....
     ..
minimum nr of rooms: 5
% java RoomSchedule
1 10
[(1,10)]
 ..........
minimum nr of rooms: 1
% java RoomSchedule
1 3 3 5 6 9 10 20
[(1,3), (3,5), (6,9), (10,20)]
 ...
   ...
      ....
         ...........
minimum nr of rooms: 2
%
% java RoomSchedule
0 10 20 30 40 50
[(0,10), (20,30), (40,50)]
...........
                   ...........
                                        ...........
minimum nr of rooms: 1
% java RoomSchedule
1 10 2 9 3 8 4 7 5 6
[(1,10), (2,9), (3,8), (4,7), (5,6)]
 ..........
  ........
   ......
    ....
     ..
minimum nr of rooms: 5
% java RoomSchedule
1 10 4 14 8 18 12 22 16 26
[(1,10), (4,14), (8,18), (12,22), (16,26)]
 ..........
    ...........
        ...........
            ...........
                ...........
minimum nr of rooms: 3
% java RoomSchedule
1 6 4 10 8 14 12 18 16 22 20 26 24 30 28 34
[(1,6), (4,10), (8,14), (12,18), (16,22), (20,26), (24,30), (28,34)]
 ......
    .......
        .......
            .......
                .......
                    .......
                        .......
                            .......
minimum nr of rooms: 2
% java RoomSchedule
1 5 2 6 3 7 4 8 5 9 6 10 7 11 8 12
[(1,5), (2,6), (3,7), (4,8), (5,9), (6,10), (7,11), (8,12)]
 .....
  .....
   .....
    .....
     .....
      .....
       .....
        .....
```

```
minimum nr of rooms: 5
%
```