Scheduling

DiPS CodeJam 23

Prompt

Having finished their new mathematical proof, Pranav and Prithvi now have to decide on the maximum number of conferences that they can attend in a year, assuming that they can only attend one conference at a time. Each conference has a set start and end time.
Can you help them figure out how many conferences they can attend?

- The first line of the input contains an integer n, denoting the number of conferences.
- The next n lines of the input each contain the start and end dates of a conference, in the format $(\text{start}, \text{end})$ in a space-separated list.

Output Format

The first and only line of your output must contain a single integer m, denoting the maximum number of conferences they can attend.

Constraints

- $30 \leq n \leq 365$
- Assume that the conferences are already sorted based on end dates.
- Assume 365 days in a year.
- Assume that dates are given in the form of a single integer n denoting the n^{th} day of the year.

Sample Input/Output

Input	Output

15
0 4
2 4
1 5
3 6
3 6
8 9
10 11
12 15
14 18
13 18
16 21
19 21
16 21
17 22
19 22
	5

Solution

This is an example of the *Activity Selection Problem*.

Simplifying the Problem

Assume there exist n conferences with each of them being represented by a start time s_i and finish time f_i. Two conferences i and j are said to be non-conflicting if $s_i \geq f_j$ or $s_j \geq f_i$. The activity selection problem consists in finding the maximal solution set (S) of non-conflicting conferences. Here, using a greedy algorithm to find the solution will always result in an optimal solution.

Solving the Problem

- Let us create an empty array $a[]$.
- Now we can start adding conferences to this array.
- Since this is a greedy algorithm, the first conference is always selected.
- Now we loop through the rest of the conferences. For each conference:
  - If this conference has a start date that is greater than or equal to the finish date of the previously selected conference, then append it to $a[]$.
- Finally, we print the length of $a[]$, denoting the number of conferences.

Sample Program

```
# n --> Total number of conferences
# s[]--> An array that contains start time of all conferences
# f[] --> An array that contains finish time of all conferences

n = int(input())

s = []
f = []

for i in range(n):
     inputArr = list(map(int, input().split()))
     s.append(inputArr[0])
     f.append(inputArr[1])

conferences = []

# The first activity is always selected
i = 0
conferences.append(i)

# Consider rest of the conferences
for j in range(n):

     # If this activity has start time greater than
     # or equal to the finish time of previously
     # selected activity, then select it
     if s[j] >= f[i]:
          conferences.append(j)
          i = j

print(len(conferences))
```