**ASSUMPTION UNIVERSITY**
**Faculty of Science and Technology**
Department of Computer Science

SC3231

Algorithm Design

# Scientific Conference

**by**

**Prateep Dharaan 5416813**

**1/2017**

# 1203. Scientific Conference

Functioning of a scientific conference is usually divided into several simultaneous sections. For example, there may be a section on parallel computing, a section on visualization, a section on data compression, and so on.

Obviously, simultaneous work of several sections is necessary in order to reduce the time for scientific program of the conference and to have more time for the banquet, tea-drinking, and informal discussions. However, it is possible that interesting reports are given simultaneously at different sections.

A participant has written out the time-table of all the reports which are interesting for him. He asks you to determine the maximal number of reports he will be able to attend.

Time limit: 1.0 second
Memory limit: 64 MB

## Input

The first line contains the number $1 \le N \le 100000$ of interesting reports. Each of the next $N$ lines contains two integers $T_s$and $T_e$ separated with a space ($1 \le T_s < T_e \le 30000$). These numbers are the times a corresponding report starts and ends. Time is measured in minutes from the beginning of the conference.

## Output

You should output the maximal number of reports which the participant can attend. The participant can attend no two reports simultaneously and any two reports he attends must be separated by at least one minute. For example, if a report ends at 15, the next report which can be attended must begin at 16 or later.

## Sample

| Input | Output |
|---|---|
| 5<br>3 4<br>1 5<br>6 7<br>4 5<br>1 3 | 3 |

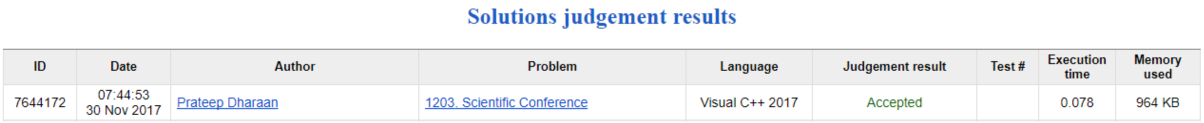
## Solution

```
#include <cstdio>
#include <algorithm>
using namespace std;
struct interval{
   int s,e;
   interval(){}
   bool operator < (interval X)const{
        return e<X.e;
    }
}I[100000];
int main(){
    int N;
    scanf("%d",&N);
    for(int i = 0;i<N;++i) scanf("%d %d",&I[i].s,&I[i].e);
    sort(I,I+N);
    int last = 0,ans = 0;
    for(int i = 0;i<N;++i){
        if(I[i].s>last){
            last = I[i].e;
            ++ans;
        }
    }
     printf("%d\n",ans);
    return 0;
}
```

**Problem Author:** Magaz Asanov
**Problem Source:** USU Internal Contest, March 2002

Difficulty: 99

## Test using Timus Online Judge



Solutions judgement results

| ID | Date | Author | Problem | Language | Judgement result | Test # | Execution time | Memory used |
|---|---|---|---|---|---|---|---|---|
| 7644172 | 07:44:53<br>30 Nov 2017 | Prateep Dharaan | 1203. Scientific Conference | Visual C++ 2017 | Accepted |  | 0.078 | 964 KB |