Week 6 Programming Assignment

Due on 2019-03-18, 23:59 IST

- Select your language (C/C++/Java/Python2/Python3)
- Paste your code into the submission window.
- There are some public test cases and some (hidden) private test cases.
- "Compile and run" will evaluate your submission against the public test cases.
- "Submit" will evaluate your submission against the hidden private test cases and report a score on 100. There are 10 private testcases in all, each with equal weightage. You will only get a score on 100. You will not get feedback on which private testcases passed or failed.
- Ignore warnings about "Presentation errors".

The Siruseri Sports Stadium

(IARCS OPC Archive, K Narayan Kumar, CMI)

The bustling town of Siruseri has just one sports stadium. There are a number of schools, colleges, sports associations, etc. that use this stadium as the venue for their sports events.

Anyone interested in using the stadium has to apply to the Manager of the stadium indicating both the starting date (a positive integer S) and the length of the sporting event in days (a positive integer D) they plan to organise. Since these requests could overlap it may not be possible to satisfy everyone.

It is the job of the Manager to decide who gets to use the stadium and who does not. The Manager, being a genial man, would like to keep as many organisations happy as possible and hence would like to allocate the stadium so that maximum number of events are held.

Suppose, for example, the Manager receives the following 4 requests:

Event No.	Starting Date	Length
1	2	5
2	9	7
3	15	6
4	9	3

He would allot the stadium to events 1, 4 and 3. Event 1 begins on day 2 and ends on day 6, event 4 begins on day 9 and ends on day 11 and event 3 begins on day 15 and ends on day 20. You can verify that it is not possible to schedule all the 4 events (since events 2 and 3 overlap and only

one of them can get to use the stadium).

Your task is to help the manager find the best possible allotment (i.e., the maximum number of events that can use the stadium).

Solution hint

Use the greedy algorithm for interval scheduling.

Input format

The first line of the input will contain a single integer N (N ≤ 100000) indicating the number of events for which the Manager has received a request. Lines 2,3,...,N+1 describe the requirements of the N events. Line i+1 contains two integer Si and Di indicating the starting date and the duration of event i. You may assume that 1 ≤ Si ≤ 1000000 and 1 ≤ Di ≤ 1000.

Output format

Your output must consist of a single line containing a single integer M, indicating the maximum possible number of events that can use the stadium.

Constraints

The range of values over which your program is to be tested is mentioned above. In addition, 50% of the test cases will also satisfy N ≤ 10000.

Example:

We now illustrate the input and output formats using the example described above.

Sample input:

```
4
2 5
9 7
15 6
9 3
```

Sample output:

```
3
```

Sample Test Cases

	Input	Output
Test Case 1	4 2 5 9 7 15 6 9 3	3
Test Case 2	20 51292 123 82060 781	18

```
19998 1405
19999 1403
20000 1401
```

Due Date Exceeded.
10 out of 10 tests passed.
You scored 100.0/100.

Your last recorded submission was :

```cpp
#include<bits/stdc++.h>
using namespace std;
bool f(const pair<long int ,long int>&a,const pair<long int ,long int>&b)
    return(a.second<b.second);
}
int main(){
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    long int n;
    cin>>n;
    vector<pair<long int,long int> >s;
    for(long int i=0;i<n;++i){
        long long int a,b;
        cin>>a>>b;
        s.push_back(make_pair(a,a+b-1));
    }
   sort(s.begin(),s.end(),f);
   long int ans=1,i=0;
   for(long int j=1;j<n;++j){
    if(s[j].first>s[i].second){
        ans++;
        i=j;
       }
   }
   cout<<ans;
}


```


End