

문제

아 과제 하기 싫다. 아무 것도 안 하고 싶다. 더 적극적으로 격렬하게 아무 것도 안 하고 싶다.

있잖아. 내가 야까 책상예다가 n 개의 과제 목록을 적어놨어. 각각의 과제 i 는 d_i 일이 걸리고, 오늘로부터 t_i 일 안에 끝내야 해. 그러니까 오늘이 0일이면, t_i 일이 끝나기 전에 제출이야. 과제는 한번 시작하면 쉬지 않고 계속해야 해. 안 그러면 머리 아파 지거든.

근데 있잖아. 내가 지금 너무, 너무 아무 것도 안 하고 싶어. 그래서 오늘은 아무 것도 안 할 거야. 더 중요한 게 뭔지 알아? 사실 나 내일도, 모레도, 아무 것도 안 하고 싶어. 한 며칠 동안은 계속 아무 것도 안하려고. 아. 과제가 있을 때 내가 내일부터 연속으로 최대 며칠동안 놀 수 있는지 궁금하다. 궁금하긴 한데, 난 아무 것도 안 하고 싶어. 나도...

좋은 생각이 났다. 너희가 이걸 대신 구해주면, 내가 너희의 맞은 문제 수를 하나 올려줄게.

입력

첫째 줄에는 과제의 개수인 정수 n ($1 \leq n \leq 10^6$)이 주어진다.

이후 n 개의 줄에 각각의 과제를 나타내는 두 정수 d_i, t_i ($1 \leq d_i, t_i \leq 10^9$)가 순서대로 주어진다. 오늘은 0일이다.

모든 입력에 대해, 오늘 아무 것도 안 해도 과제를 마무리 할 수 있는 방법이 존재함이 보장된다.

출력

내일(1일)부터 연속으로 최대 며칠 동안 놀 수 있는지를 출력한다. 가령, 답이 0이면, 내일 과제를 해야 하며, 1 이면, 모레에 과제를 해야 한다.

예제 입력 1 복사

```
3
2 8
1 13
3 10
```

예제 출력 1 복사

```
5
```



마감일이 가장 늦은 과제부터 채워나가기 ?

```
struct Task {
    int d;
    int t;
};
```

Task 구조체
(라제하는데 걸리는 시간 d
마감일 t)

```
bool cmp(Task& p1, Task& p2) {
    return p1.t > p2.t;
}
```

마감일을 기준으로 내림차순으로 정렬하기 위한
sort 함수에 쓸 비교 함수 cmp

```
int main() {
    int n;
```

n 입력 받기

```
    cin >> n;
```

```
    Task* tasks;
```

Task 구조체 배열 tasks

```
    tasks = new Task[n];
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &(tasks[i].d));
```

각 d 값은 입력 받기

```
        scanf("%d", &(tasks[i].t));
```

```
    }
```

```
    sort(tasks, tasks + n, cmp);
```

t를 기준으로 내림차순으로 정렬

```
    int result = tasks[0].t;
```

내림차순 했으니까 가장 큰 t 값을
result에 초기화

```
    for (int i = 0; i < n; i++) {
```

```
        if (result > tasks[i].t) result = tasks[i].t;
```

```
        result -= tasks[i].d;
```

```
    }
```

result = 13 - 1 = 12

```
    printf("%d", result);
```

12랑 10 비교, 더 작은 수!

= 10 - 3 = 7

7이랑 8 비교, 더 작은 수!

= 7 - 2 = 5 → 출력값

d	t
1	13
3	10
2	8