

观看文艺汇演问题题目描述：

为庆祝中国共产党成立 **100** 周年，某公园将举行多场文艺汇演，很多演出都是同时进行。一个人只能同时观看一场演出，且不能迟到早退。由于演出分散在不同的演出场地，所以连续观看的演出最少要有 **15** 分钟的时间间隔。

小明是一个狂热的文艺迷，想观看尽可能多的演出。现给出演出时间表，请帮小明计算他最多能观看几场演出。

输入描述：

第一行为一个数  $N$ ，表示演出场数， $1 \leq N \leq 1000$

接下来  $N$  行，每行两个空格分隔的整数，第一个整数  $T$  表示演出开始时间，第二个整数  $L$  表示演出持续时间。 $T$  和  $L$  的单位都是分钟， $0 \leq T \leq 1440$ ， $0 < L \leq 180$

输出描述：

请输出最多能观看的演出场数

补充说明：

示例 1

输入：

```
2
720 120
840 120
```

输出：

```
1
```

说明：

两场演出间隔时间为  $0$ ，不满足最小  $15$  分钟时间间隔的要求，所以最多只能观看一场演出

## 示例 2

输入：

```
2
0 60
90 60
```

输出：

```
2
```

说明：

两场演出间隔大于  $15$  分钟，都能观看到

```
import java.util.Scanner;
import java.util.Arrays;
import java.util.Comparator;
```

// 注意类名必须为 Main，不要有任何 package xxx 信息

```
public class Main {
    public static class Peform{
        int start;
        int duration;
        public Peform(int s,int d){
            start = s;
            duration = d;
        }
    }
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        // 注意 hasNext 和 hasNextLine 的区别
        int N = in.nextInt();
        Peform[] peforms = new Peform[N];
        for(int i = 0; i < N; i++){
            int start = in.nextInt();
            int duration = in.nextInt();
            peforms[i] = new Peform(start, duration);
        }

        Arrays.sort(peforms,Comparator.comparingInt(p -> (p.duration + p.start)));
    }
}
```

```
int count = 0;
int lastMoment = 0;
for(int i = 0; i < N; i++){
    if(peforms[i].start >= lastMoment){
        count++;
        lastMoment = peforms[i].start + peforms[i].duration + 15;
    }
}

System.out.println(count);
}
```