

5044--怎样更有力气

(liqi)

【问题描述】

OI 大师抖儿在夺得银牌之后，顺利保送 PKU。这一天，抖儿问长者：“我虽然已经保送了，但我的志向是为国家健康工作五十年。请问我应该怎样变得更有力气？”

长者回答：“你啊，Too Young Too Simple,Sometimes Naive!如果你想要我教你，你要先进行艰苦的修行。”

长者的住宅中有一堵长度为 n 的墙。每天抖儿起床修行，会选择一段长度为 x 的区间染成白色。长者的住宅附近有一群香港记者，为了借助抖儿拜访长者，第 i 天香港记者会将区间 $[l_i, r_i]$ 染成白色来讨好抖儿（也就是说，每天墙会被抖儿和香港记者各染一次）。现在抖儿已经预先知道了香港记者的动向，他想知道他最少几天就能把墙全部染白，完成修行。

【输入格式】

第一行三个整数 n, m, x ，分别表示墙的长度，天数和区间的长度。

接下来 m 行，每行两个整数 l_i, r_i ，表示香港记者在第 i 天会将区间 $[l_i, r_i]$ 染成白色。

【输出格式】

输出仅一行为一个整数，表示抖儿最少几天能把墙全部染白。

如果 m 天之后依然无法染白，则输出 “Poor Douer!”

【输入样例 1】

```
10 3 3
2 5
4 8
9 10
```

【输出样例 1】

```
2
```

【样例 1 说明】

第一天抖儿刷墙的区间为 $[1, 3]$ ；第二天抖儿刷墙的区间为 $[8, 10]$

【输入输出样例 2】

见选手目录下的 liqi/liqi2.in 和 liqi/liqi2.out

【限制与约定】

| 测试点编号 | n | m | x |
|-------|------------------|-----------------|-------|
| 1 | $n \leq 10$ | $m \leq 1$ | |
| 2 | $n \leq 10$ | $m \leq 10$ | |
| 3 | $n \leq 100$ | $m \leq 100$ | |
| 4 | $n \leq 1000$ | $m \leq 1000$ | |
| 5 | $n \leq 10000$ | $m \leq 10000$ | |
| 6 | $n \leq 100000$ | $m \leq 100000$ | $x=0$ |
| 7 | $n \leq 10^{18}$ | | |
| 8 | $n \leq 100000$ | | |
| 9 | $n \leq 10^{18}$ | | |
| 10 | | | |

对于所有的数据，保证 $n \leq 10^{18}$, $m \leq 100000$, $x \leq n$ 且数据随机

【题目分析】

$m=1$: 由于此部分分低于普及组难度，不再讲述

$n, m \leq 10^4$:

考虑第 i 天，先处理香港记者染白的区间。

令 $dp[i]$ 表示 $1 \sim i$ 全染白至少操作几次。

则 $dp[i] = dp[i-1]$ (i 为白点) 或 $dp[i-x]+1$ (i 为黑点)

时间复杂度 $O(nm)$

$x=0$:

二分答案，那么只要看香港记者染白的区间是否将墙完全覆盖即可。

将区间按左端点排序，比较区间的左端点与前一个区间的右端点即可。

100 分:

其实已经很明了了。

在 task3 中，如果当前区间的左端点小于上一个区间的右端点，就直接退出。

现在只要从上一个区间的右端点+1 开始涂，一直涂到 \geq 当前区间左端点即可。

注意到一次可能直接把整一段覆盖了，那么下一次就要从当前涂到的地方开始。

思路：先二分答案天数，对于给定的区间按左端点从小到大排序，能覆盖就覆盖，否则计算天数： $Ans += (b[i].l - s + 2) / x + 1$ ；更新覆盖位置 $s = \max\{b[i].r, s + tmp * x\}$

```
#include<cstdio>
#include<algorithm>
#include<cstring>
using namespace std;
typedef long long ll;
const int M=100010;
ll n,x;
int m,i;
struct shit{ll l,r;}a[M],b[M];
bool cmp(shit x,shit y){return x.l<y.l||x.l==y.l&& x.r<y.r;}
int check(int mid)
{
    for(int i=1;i<=mid;i++)b[i]=a[i];
    sort(b+1,b+mid+1,cmp);
    ll ans=0,s=0;
    b[mid+1].l=n+1;
    for(int i=1;i<=mid+1;i++)
    {
        if(b[i].l>s+1)
        {
            if(!x)return 0;//注意此处
            ll tmp=(b[i].l-s-2)/x+1;
            if(s+tmp*x>b[i].r)s=s+tmp*x;
            ans+=tmp;
        }
    }
}
```

```
        if(ans>mid)return 0;
    }
    s=max(s,b[i].r);
}
return 1;
}
int main()
{
    freopen("liqi.in","r",stdin);
    freopen("liqi.out","w",stdout);
    scanf("%lld%d%lld",&n,&m,&x);
    for(i=1;i<=m;i++)scanf("%lld%lld",&a[i].l,&a[i].r);
    int L=1,r=m,Ans;
    while(L<=r)
    {
        int mid=L+r>>1;
        if(check(mid)){Ans=mid;r=mid-1;}
        else L=mid+1;
    }
    if(!check(m))printf("Poor Douer!\n");
    else printf("%d\n",Ans);
    return 0;
}
```