

训练赛 6

题目名称	点点	矩形	牛走
源文件名	dot.cpp/c/pas	rectangle.cpp/c/pas	walk.cpp/c/pas
输入文件名	dot.in	rectangle.in	walk.in
输出文件名	dot.out	rectangle.out	walk.out
时间限制	1s	2s	1s
空间限制	64MB	256MB	64MB
优化开关	无	无	无

点点(dot.cpp/c/pas)

题目描述

有 N 个位置, 从左至右依次标号为 1-N, 给出了 M 个区间, 你可以在一些位置放上一个点, 但是对于这 M 个区间都要满足区间内恰有一个点, 求最多能放多少个点.

输入格式

第一行包含两个整数 N, M.

接下来 M 行, 每行两个整数 $l[i], r[i]$, 分别代表该区间的左端点和右端点.

输出格式

输出一个数字代表答案, 如果无法满足条件则输出-1.

样例

dot.in	dot.out
5 3 1 4 2 5 3 4	1

数据范围

测试点	分值	N	M
1	10	≤ 10	≤ 10
2	10	≤ 100	≤ 100
3	10	≤ 1000	≤ 200
4	10	≤ 1000	≤ 2000
5	10	≤ 2000	≤ 1000
6	10	≤ 2000	≤ 1000
7	10	≤ 200000	≤ 100000
8	10	≤ 200000	≤ 100000
9	10	≤ 200000	≤ 100000
10	10	≤ 100000	≤ 100000

矩形(rectangle.cpp/c/pas)

题目描述

有一个 $N \times N$ 的棋盘, 棋盘上有一些格子是坏的. 现在你需要在棋盘上框出两个矩形, 要求满足以下条件:

1. 一个矩形在上方, 一个矩形在下方, 上方矩形的底边必须被下方矩形的顶边包含.
2. 两个矩形内部至少有一个格子.
3. 矩形的边界上不能够有坏格子.

两个矩形的权值为两个矩形内点的个数的乘积, 求乘积最大是多少.

如下是一个棋盘和一个框矩形的最优方案

```
.....
.....
...*****...
.*.....*.*
.*.....*.*
....*.....
...*...****...
.....
..**.*.*.*..
...*...**.*...
*..*...*.....
.....
.....*.*.....
.....*.....
.....
```

```
..@#@#@#@#@#@..
..@.....@..
..@*****..@..
.*@...*.....@.*
.*@.....*...@*..
..@.*.....@..
..@*...****..@..
..@#@#@#@#@#@#@..
..@**.*.*.*.*..@..
..@.*...**.*..@..
*@.*...*.....@..
..@.....@..
..@...*.*...@..
..@.....*...@..
..@#@#@#@#@#@#@..
```

有一个 6×9 的矩形和一个 6×12 的矩形. 总的分数为 3888.
注意: 矩形的边界是有宽度的, 矩形内部的点数不包括边界上的点.

输入格式

第一行包含一个数 N , 代表棋盘的边长.

接下来 N 行, 每行一个字符串, 第 i 个字符串描述了棋盘的第 i 行的状态, '*'代表坏格子, '.'代表好格子.

数据保证存在框矩形的方案.

输出格式

输出一个整数代表答案.

样例

rectangle.in	rectangle.out
15 <pre> * * * * * . . . * . . . * . . * . * . . . * . . * . . . * * . . * * * * * * * . * * . . . * . . * * * * . * . . * * . . * * </pre>	3888

样例解释

解释过了.

数据范围

测试点	分值	N
1	10	≤ 15
2	10	≤ 10
3	10	≤ 80
4	10	≤ 200
5	10	≤ 250
6	10	≤ 250
7	10	≤ 300
8	10	≤ 300
9	10	≤ 300
10	10	≤ 300

牛走(walk.cpp/c/pas)

题目描述

有一张 N 个点的图, 图中有 $N-1$ 条有向边, 保证从任何一个点都能够走到 1 号点. 对于每个点 $i > 1$, 都有一条边通向 $P[i]$, 这个点上有 $C[i]$ 头奶牛. 在每个单位时间内, i 到 $P[i]$ 这条道路允许通过不超过 $M[i]$ 只牛.

柠檬想要所有牛都集中在 1 号点. 下面是走奶牛的规则: 1) 我们认为时间是离散的 2) 任何牛都可以在一个单位时间内走任意多条道路. 但是必须满足每条道路的上限为 $M[i]$. 3) 奶牛不会离开 1 号点. 换句话说, 每个单位时间, 每头奶牛可以选择一下行动之一: 1) 留在当前点. 2) 经过一条或多条道路向 1 号点移动. 同样需要满足每条道路的上限 $M[i]$.

柠檬有 K 个 $T[i]$, 他想知道对于每个 $T[i]$, 如果采取最优策略, $T[i]$ 时刻结束时(也就是过了 $T[i]$ 个单位时间)最多有多少头牛能够到 1 号点.

输入格式

第一行包含两个整数 N, K .

接下来 $N-1$ 行, 每行包含三个整数, 分别表示 $P[i], C[i], M[i]$.

接下来 K 行, 每行一个整数 T[i].

输出格式

一共输出 K 行, 第 i 行输出第 i 个询问的答案.

样例

walk.in	walk.out
4 1 1 1 5 2 12 7 3 12 3 5	25

数据范围

$0 \leq W[i], C[i], T[i] \leq 1,000,000,000$

测试点	分值	N	K	特殊条件
1	10	≤ 20	≤ 100	无
2	10	≤ 20	≤ 5	$T[i] \leq 30$
3	10	≤ 100	≤ 50	$T[i] \leq 1500$
4	10	≤ 500	≤ 50	$T[i] \leq 1500$
5	10	≤ 1000	≤ 100	$T[i] \leq 10000$
6	10	≤ 1000	≤ 100	无
7	10	≤ 10000	≤ 1000	
8	10	≤ 50000	≤ 10000	
9	10	≤ 50000	≤ 100	
10	10	≤ 50000	≤ 50000	