

問題 1 椅子の総数

(3点)

選手のみなさん、パソコン甲子園によろこそ。パソコン甲子園の本選は会津大学で行われ、会場内では1つの机に1チームが割り当てられます。パソコン甲子園は1チーム2人なので、チーム数×2脚の椅子が必要です。大学では、他にも様々なイベントの会場設営で机と椅子を準備する機会がありますが、必要な机と椅子の数も様々です。そこで、あるイベントに対して準備する机の数と、机1つあたりに必要な椅子の数が与えられたとき、必要な椅子の総数を計算するプログラムを作成してください。

入力

入力は以下の形式で与えられる。

d c

入力は1行であり、必要な机の数 $d(1 \leq d \leq 100)$ と机1つあたりに必要な椅子の数 $c(1 \leq c \leq 10)$ を表す整数が与えられる。

出力

イベントにおいて必要な椅子の総数を1行に出力する。

入出力例

入力例 1	出力例 1
20 2	40
入力例 2	出力例 2
1 1	1

問題2 お財布メタボ診断

(4点)

4月に消費税が8%になってから、お財布が硬貨でパンパンになっていませんか。同じ金額を持ち歩くなら硬貨の枚数を少なくしたいですね。硬貨の合計が1000円以上なら、硬貨をお札に両替して、お財布のメタボ状態を解消できます。

お財布の中の硬貨の枚数が種類ごとに与えられたとき、硬貨をお札に両替できるかどうかを判定するプログラムを作成してください。

入力

入力は以下の形式で与えられる。

```
c1 c5 c10 c50 c100 c500
```

入力は1行からなる。 $c1, c5, c10, c50, c100, c500$ ($0 \leq c1, c5, c10, c50, c100, c500 \leq 50$) は、それぞれ、1円、5円、10円、50円、100円、500円硬貨の枚数を表す。

出力

硬貨をお札に両替できるなら1を、そうでなければ0を1行に出力する。

入出力例

入力例 1	出力例 1
3 1 4 0 4 1	0
入力例 2	出力例 2
2 1 4 3 2 3	1
入力例 3	出力例 3
21 5 9 3 1 1	0
入力例 4	出力例 4
2 4 3 3 3 1	1
入力例 5	出力例 5
50 50 50 4 0 0	1

問題3 残り物には福がある

(4点)

K個の石から、P人が順番に1つずつ石を取るゲームがあります。P人目が石を取った時点で、まだ石が残っていれば、また1人目から順番に1つずつ石を取っていきます。このゲームでは、最後の石を取った人が勝ちとなります。KとPが与えられたとき、何人目が勝つか判定するプログラムを作成してください。

入力

入力は以下の形式で与えられる。

```
N
K1 P1
K2 P2
:
KN PN
```

1行目にはゲームを行う回数 N ($1 \leq N \leq 100$) が与えられる。続く N 行に、 i 回目のゲームにおける石の個数 K_i ($2 \leq K_i \leq 1000$) と、ゲームに参加する人数 P_i ($2 \leq P_i \leq 1000$) が与えられる。

出力

それぞれのゲームについて、何人目が勝つかを1行に出力する。

入出力例

入力例	出力例
3	1
10 3	2
2 10	2
4 2	

問題 4 路線バスの時刻表

(5点)

バスマニアの健次郎君は、A市内のバスをよく利用しています。ある日ふと、健次郎君の家の前のバス停から出発するすべてのバスを写真に収めることを思い立ちました。このバス停には飯盛山行きと鶴ヶ城行きの2つのバス路線が通ります。各路線の時刻表は手に入れましたが、1つの時刻表としてまとめた方がバス停で写真が撮りやすくなります。

健次郎君を助けるために、2つの路線の時刻表を、0時0分を基準として出発時刻が早い順に1つの時刻表としてまとめるプログラムを作成してください。

入力

入力は以下の形式で与えられる。

```
N h1 m1 h2 m2 ... hN mN
M k1 g1 k2 g2 ... kM gM
```

1行目に、飯盛山行きのバスの数を表す整数 N ($1 \leq N \leq 100$)と、それに続いて飯盛山行きのバス出発時刻が早い順に与えられる。出発時刻は整数 h_i ($0 \leq h_i \leq 23$)と m_i ($0 \leq m_i \leq 59$)からなり、 h_i 時 m_i 分に i 番目のバスが出発することを表す。2行目に、鶴ヶ城行きのバスの数を表す整数 M ($1 \leq M \leq 100$)と、それに続いて鶴ヶ城行きのバス出発時刻が早い順に与えられる。出発時刻は整数 k_i ($0 \leq k_i \leq 23$)と g_i ($0 \leq g_i \leq 59$)からなり、 k_i 時 g_i 分に i 番目のバスが出発することを表す。

同じ路線には、同じ時刻に出発するバスはないものとする。

出力

2つの路線の時刻表を、0時0分を基準として出発時刻が早い順に1つにまとめた時刻表を1行に出力する。ただし、同じ時刻に複数のバスが出発するときは1つだけを出力する。出発時刻の時と分を:で区切る。分が1桁のときは左に0をひとつ加えて2桁で出力する。時刻と時刻の間は空白1つで区切る。

入出力例

入力例 1	出力例 1
2 9 8 15 59 1 8 27	8:27 9:08 15:59

入力例 2	出力例 2
2 0 0 21 0 3 7 30 21 0 23 7	0:00 7:30 21:00 23:07

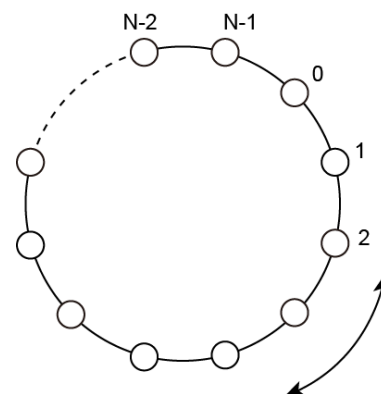
問題5 鉄道路線 II

(8点)

わたしの住む街には、図のようなN個の駅からなる円環状の鉄道路線があります。この路線の各駅には0からN-1までの番号が順番に割り当てられていて、隣の駅まで100円で移動することができます。移動はどちらの方向にでも可能です。

わたしはこの路線上のいくつかの駅で買い物をしたいと思っています。そこで、プログラムを作成して移動費を計算することにしました。

ある駅を始点として、買い物をするすべての駅を訪問する最小の費用(円)を求めるプログラムを作成してください。ただし、同じ駅を何度訪問しても良く、どのような順番で駅を訪問しても構いません。



入力

入力は以下の形式で与えられる。

```
N M p
d1
d2
:
dM
```

1行目に駅の数 N ($2 \leq N \leq 100000$)、買い物をする駅の数 M ($1 \leq M \leq 10000$ かつ $M < N$)、出発する駅の番号 p ($0 \leq p \leq N-1$) が与えられる。続く M 行に買い物をする駅の番号 d_i ($0 \leq d_i \leq N-1$) が与えられる。ただし、 d_i は全て異なり、 p もまたどの d_i とも異なるものとする。

時間制限

入力に対して、実行時間が5秒を超えてはならない。

出力

最小の費用を1行に出力する。

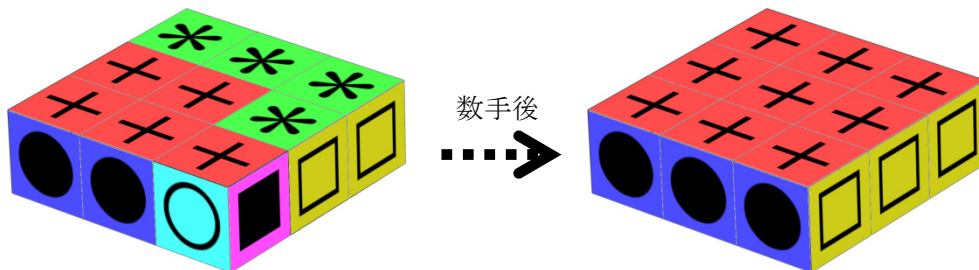
入出力例

入力例 1	出力例 1
5 4 0 1 2 3 4	400
入力例 2	出力例 2
7 2 1 6 2	400

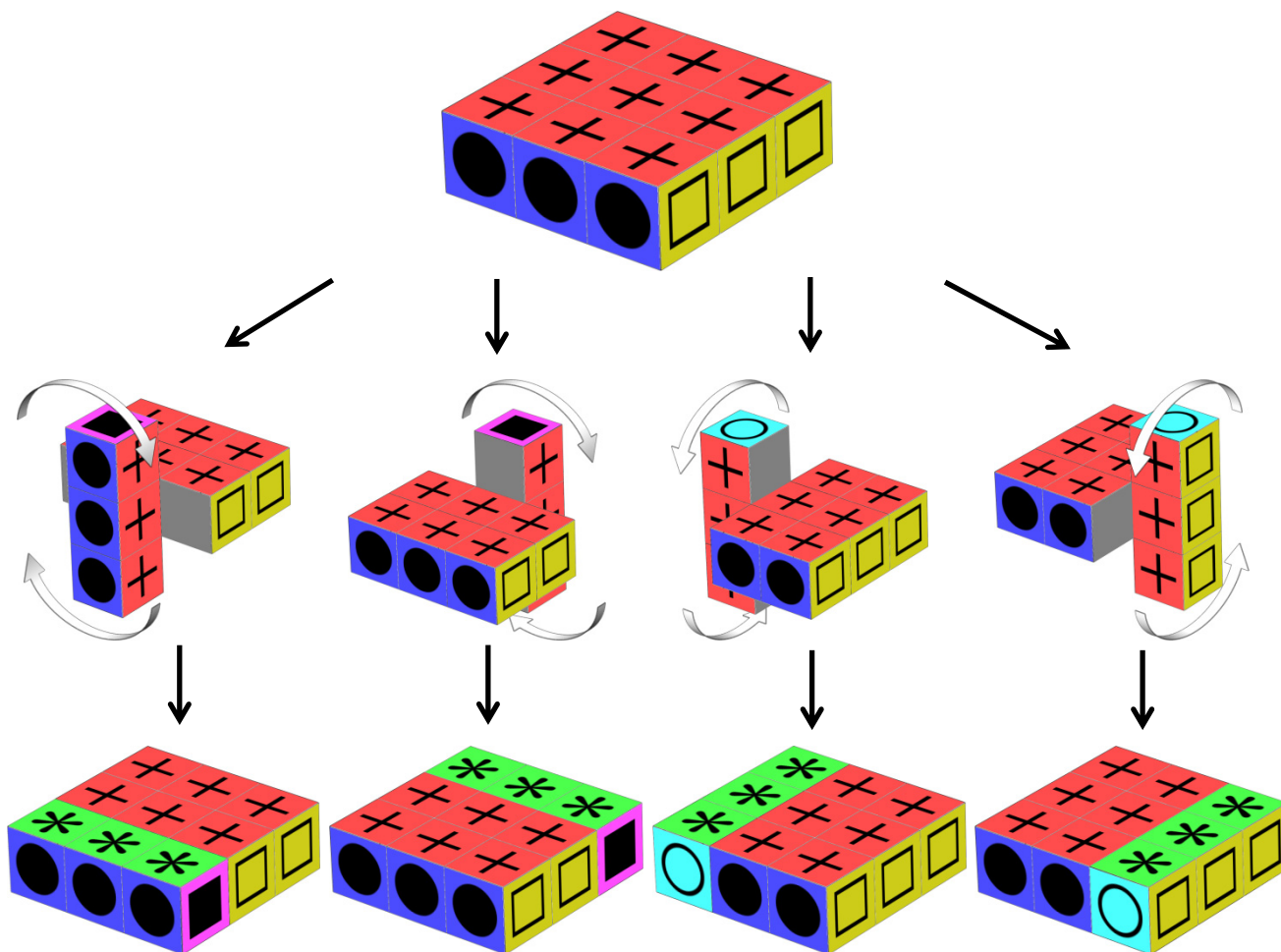
問題6 フロッピーキューブ

(10点)

フロッピーキューブをプログラミングで解いてみましょう。フロッピーキューブは図のように表面に色のついた9個の立方体から構成されている立体パズルで、キューブの列を回転させることによって、6つの各面の色をそろえます。



フロッピーキューブに対しては下図のような4種類の操作を行うことができ、一回の操作で、端にある3つの隣接したキューブを180度回転することができます。わかりやすいように、図では、上面に+ (赤色)、下面に* (緑色)、右前面に□ (黄色)、左前面に● (青色)、右奥面に○ (水色)、左奥面に■ (紫色)の記号が付いている状態を初期状態としています。



フロッピーキューブの初期状態が与えられるので、パズルを解くために必要な最小の操作回数を求めるプログラムを作成してください。

入力

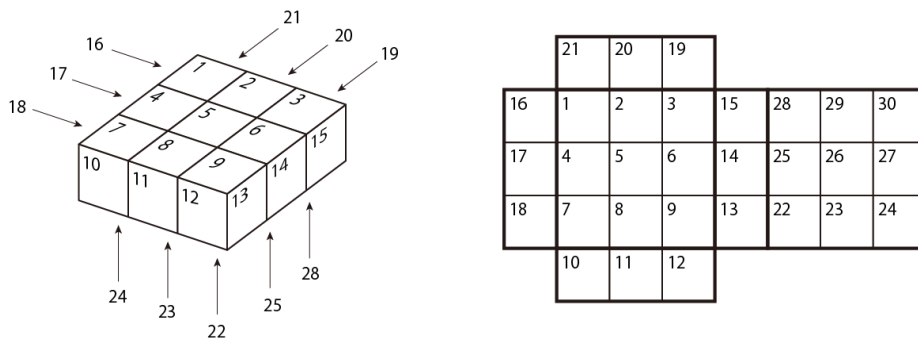
入力は以下の形式で与えられる。

```
N
puzzle1
puzzle2
:
puzzleN
```

1 行目の N ($1 \leq N \leq 30$) は操作回数を計算したいパズルの数である。続く N 行に各フロッピーキューブの初期状態 $puzzle_i$ が与えられる。 $puzzle_i$ は以下の形式で与えられる。

```
p1 p2 p3 p4 p5 p6 p7 p8 p9 p10 p11 p12 p13 p14 p15 p16 p17 p18 p19 p20 p21 p22 p23 p24 p25 p26 p27 p28 p29 p30
```

各フロッピーキューブの情報は30個の整数 p_i ($1 \leq p_i \leq 6$) からなる。 p_i は、下図のようにフロッピーキューブの各面に番号 i を振ったときの、そのキューブの面の色を表す。



パズルは、多くとも 8 回の操作で解くことができると仮定してよい。

時間制限

入力に対して、実行時間が 5 秒を超えてはならない。

出力

パズルごとに、最小の操作回数を 1 行に出力する。

入出力例

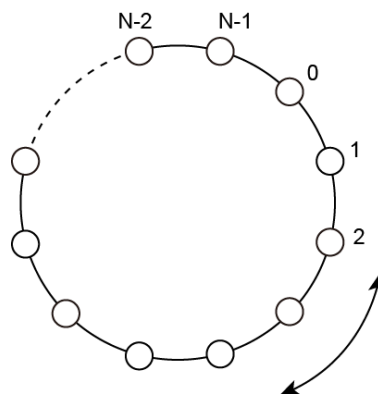
入力例	出力例
4	0
1 1 1 1 1 1 1 1 1 2 2 2 4 4 4 6 6 6 5 5 5 3	1
3 3 3 3 3 3 3 3	2
3 3 3 1 1 1 1 1 1 2 2 2 4 4 6 4 6 6 5 5 5 3	7
3 3 3 3 3 1 1 1	
3 3 3 1 1 3 1 1 1 2 2 5 6 4 4 4 6 6 2 5 5 3	
3 3 1 3 3 1 1 1	
1 3 1 3 1 3 3 1 3 2 2 2 6 4 4 6 6 4 5 5 5 1	
3 1 1 3 1 3 1 3	

※ 入力例で、各フロッピーキューブの初期状態におけるキューブの面の色を表す数列は、表示の都合上 2 行に見えるが、実際は 1 行である。

問題7 バトンリレーゲーム

(12点)

アカベ高校では、毎年全校生徒が参加するゲームを行っています。まず、校庭に N 人の全校生徒が円形に並びます。図のように、各生徒は 0 から $N-1$ までの番号が書かれたゼッケンを付けています。ゲームではバトンを 1 本使い、最初はゼッケン 0 番の生徒がバトンを持っています。



そこから、以下の手順を M 回繰り返します。まず、現時点でバトンを持っている生徒が適当な正の整数 a を宣言します。 a が偶数のときは時計回り、奇数のときは反時計回りに隣の生徒にバトンを渡していき、 a 番目にバトンを受け取った生徒が脱落します。脱落した生徒は、時計回りで隣の生徒にバトンを渡し、円から抜けます。

ゲームが終わった後に円に残った生徒は、放課後の掃除が 1 年間免除されます。しかし、ここ数年は生徒数が増えたため、全校生徒を集めるのが難しくなっています。そこで、競技プログラミング部のあなたは、シミュレーションで掃除が免除される生徒を求めるプログラムを作成するよう頼まれました。

指定した生徒が掃除を免除されているかどうかを質問したとき、それに答えるプログラムを作成してください。

入力

入力は以下の形式で与えられる。

```
N M Q
a1 a2 ... aM
q1 q2 ... qQ
```

入力は 3 行であり、 1 行目に生徒の人数 N ($10 \leq N \leq 200000$)、繰り返し回数 M ($5 \leq M < N$)、生徒が掃除を免除されるかどうかを問い合わせる質問の個数 Q ($1 \leq Q \leq 1000$) が与えられる。続く 1 行に、 i 回目の繰り返して最初にバトンを持っている生徒が宣言する整数 a_i ($1 \leq a_i \leq 100$) が与えられる。続く 1 行に、質問としてゼッケン番号 q_i ($0 \leq q_i < N$) が与えられる。

時間制限

入力に対して、実行時間が 5 秒を超えてはならない。

出力

質問ごとに、ゼッケン番号 q_i の生徒が掃除を免除されるかどうかを i 行目に出力する。掃除が免除されるなら 1 を、されないなら 0 を出力する。

入出力例

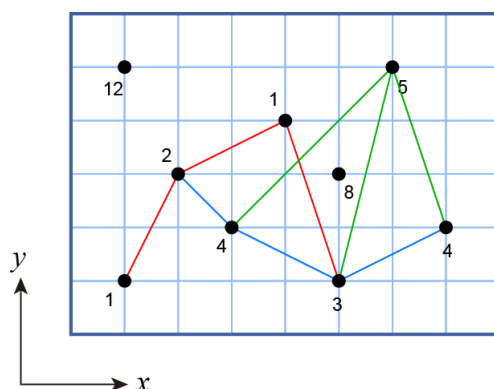
入力例	出力例
10 5 3	1
2 6 5 18 3	0
3 0 5	1

問題 8 天体観測

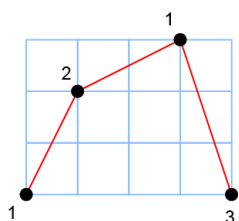
(16点)

ある晴れた夜の帰り道、空を見上げるとそこには無数の星。強く輝く星々、かすかに見える星々、いろいろな明るさの星々が夜空を彩っています。

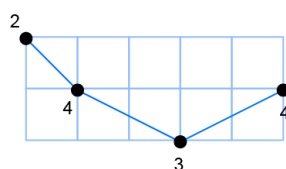
あなたはふと思いました。この星空に自分の星座があつたらどんなに素敵だろうと。そこであなたはひとつの基準を決め、部屋の窓から見える星々を適当につなげて自分の星座を作ることにしました。その基準とは、「どの2つの星を選んでも、それらの明るさの差がある定数 d 以下になるような星のグループを作り、その中で一番大きいものを自分の星座にしよう！」というものです。例えば、図のような窓から見える夜空を考えてみましょう（外側の長方形は窓枠です）。



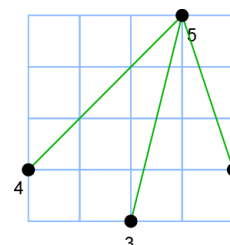
この夜空には、明るさがそれぞれ1, 12, 2, 4, 1, 8, 3, 5, 4の9つの星がありますが、 $d=2$ とすると例えば以下のような3つの星座A, B, Cができます。



大きさが12の星座A



大きさが10の星座B



大きさが16の星座C

星座の大きさを次のように決めることにしました。ある星座の星をすべて含むような、窓枠に平行な辺からなる長方形のうち、面積が最も小さいものを選びます。この長方形の面積をその星座の大きさとしています。例えば、上の夜空では星座A, B, Cの大きさはそれぞれ12, 10, 16になるので、星座Cが最も大きい星座となります。

N 個の星の位置と明るさ、および整数 d が与えられたとき、一番大きい星座の面積を求めるプログラムを作成してください。星の位置は窓枠の左下隅を原点とした座標で与えられ、軸は図のような向きとします。星座を構成する星が1つの場合や、星々が軸に平行な直線上にある場合は、その星座の面積は0となることに注意してください。

入力

入力は以下の形式で与えられる。

```
N d
x1 y1 b1
x2 y2 b2
:
xN yN bN
```

1行目に星の数 N ($1 \leq N \leq 200000$) と整数 d ($0 \leq d \leq 10^9$) が与えられる。続く N 行に、 i 番目の星の座標を表す整数 x_i ($0 \leq x_i \leq 2000000$) と y_i ($0 \leq y_i \leq 2000000$)、明るさを表す整数 b_i ($0 \leq b_i \leq 10^9$) が与えられる。入力される星の座標はすべて異なる。

時間制限

入力に対して、実行時間が5秒を超えてはならない。

出力

一番大きい星座の面積を1行に出力する。

入出力例

入力例	出力例
9 2 1 1 1 1 5 12 2 3 2 3 2 4 4 4 1 5 1 3 5 3 8 6 5 5 7 2 4	16

問題9 力持ち

(18点)

力持ちたちが集う力持ち学園がありました。力持ち学園の運動会では、力持ちたちが隊列を組んで行進します。

力持ちたちは常に自分たちの力を誇示したい一方で、彼らの大半は自分で歩きたくありません。そこで彼らの一部が一番下になり、その上に大勢の人を縦一列に持ちあげて歩くことで、実際に歩く人数を減らそうと考えました。

はじめに、 N 人の力持ちは地面の上に横一列に並んでいて、それぞれ左側から $1, 2, 3, \dots, N$ と通し番号で呼ばれています。通し番号 i の力持ちの体重は w_i で最大 c_i の重量まで持つことができます。

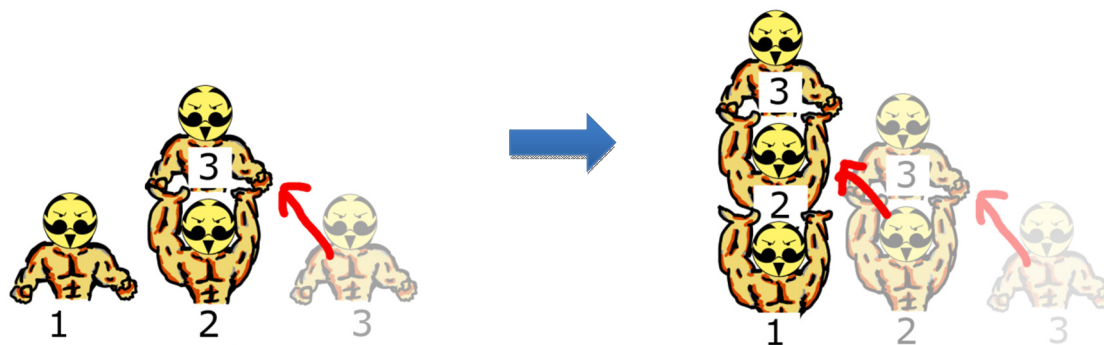
力持ちは、以下の条件をすべて満たすときだけ、隣に立っている左右どちらかの力持ちを持ちあげることができます。

- 自分の上下には力持ちはいない。つまり、誰かに持ち上げられてもいないし、誰かを持ちあげてもいない。
- 隣の力持ちの体重が、自分が持つことのできる最大の重量以下である。ただし、隣の力持ちが既に誰かを持ち上げているなら、縦に積み上がった力持ちたちの体重の合計が、自分が持つことのできる最大の重量以下でなければならない。

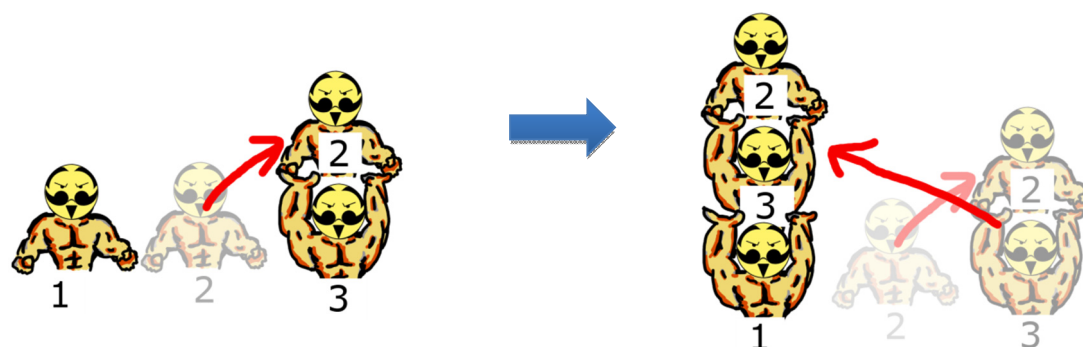
例えば、右のような3人の力持ちの隊列を考えてみましょう。



下図のように、2の力持ちが持つことのできる重さが w_3 以上のとき、2の力持ちは3の力持ちを持ちあげることができます。続いて、1の力持ちが持つことのできる重さが $w_2 + w_3$ 以上のとき、1の力持ちは2の力持ちを持ちあげることができます。



また、下図のように3の力持ちが2の力持ちを持ちあげた場合は、1の力持ちの隣が、2の力持ちを持った3の力持ちになるので、1の力持ちは3の力持ちを持ちあげることができます。



2の力持ちが、下の図のように1と3の力持ちを両方持つことはできません。



力持ち学園の専属プログラマーとして、一番下で歩くことになる最小の人数を求めてください。

入力

入力は以下の形式で与えられる。

N
c_1 w_1
c_2 w_2
:
c_N w_N

1行目に力持ちの人数 N ($1 \leq N \leq 1000$) が与えられる。続く N 行に i 番目の力持ちが持てる重量の最大値 c_i ($1 \leq c_i \leq 100000$) と体重 w_i ($1 \leq w_i \leq 100000$) が与えられる。

時間制限

入力に対して、実行時間が5秒を超えてはならない。

出力

最小の人数を1行に出力する。

入出力例

入力例 1	出力例 1
3	1
150 120	
100 50	
80 100	

入力例 2	出力例 2
8	3
50 100	
20 20	
30 20	
90 50	
140 30	
30 60	
59 120	
10 10	

問題 10 学食

(20点)

今日はZ大学のオープンキャンパスです。毎年この日の昼休みには、大勢の高校生たちが学食に列をつくりまわります。そこでZ大学の事務局は、行列の長さが最大でどのくらいの距離になるかを予測することにしました。

事前調査の結果で、以下のことが分かっています。

- 行列にはそれぞれ1からNまでの番号が振られたN人が並びます。
- C個の高校生のペア (a_i, b_i) それぞれについて、以下の2種類の制約があります：
 - 1つ目の制約は順序に関するもので以下のいずれかです：
 - ◇ a_i は b_i よりも先、または同じ位置に並ばなくてはならない
 - ◇ a_i は b_i よりも後、または同じ位置に並ばなくてはならない
 - ◇ a_i は b_i より先でも、同じ位置でも、後でもよい
 - 2つ目の制約は距離に関するもので以下のいずれかです：
 - ◇ a_i と b_i は d_i メートル以上離れなければならない
 - ◇ a_i と b_i は d_i メートル以内に並ばなければならない

また、先頭から同じ距離の場所に複数の人が並ぶことができ、行列の先頭には常に番号1の人が並ぶことが分かっています。

与えられたC個の制約をすべて満たす行列について、先頭から最後尾までの距離が最大となるような並び方をした場合の距離を求めるプログラムを作成してください。ただし、どこまでも離れることができる場合は“inf”と、制約を満たす並び方が不可能な場合は“-1”と出力してください。

入力

入力は以下の形式で与えられる。

```
N C
constraint1
constraint2
:
constraintC
```

1行目に行列に並ぶ高校生の人数 N ($2 \leq N \leq 100$) と制約の数 C ($0 \leq C \leq 200$) が与えられる。続く C 行に各制約 constraint_i が次の形式で与えられる。

```
 $a_i o_i b_i s_i d_i$ 
```

制約には空白は含まれない。 a_i, o_i, b_i, s_i, d_i の意味を以下に示す。

- a_i と b_i ($1 \leq a_i, b_i \leq N$ かつ $a_i \neq b_i$) は高校生の番号、 d_i は距離 ($0 \leq d_i \leq 10000$) を表す整数である。
- o_i は順序の制約を指定する $<=$ 、 $>=$ 、 $*$ のいずれかの文字列であり、 $<=$ の場合「 a_i は b_i よりも先、または同じ位置に並ばなくてはならない」、 $>=$ の場合「 a_i は b_i よりも後、または同じ位置に並ばなくてはならない」、 $*$ の場合「 a_i は b_i より先でも、同じ位置でも、後でもよい」ことを意味する。ただし o_i が $*$ となる制約は7個以上与えられることはない。
- s_i は距離の制約を指定する $+$ または $-$ の文字であり、 $+$ の場合「 a_i と b_i は d_i メートル以上離れなければならない」、 $-$ の場合「 a_i と b_i は d_i メートル以内に並ばなければならない」ことを意味する。

ただし、あるペアに対して複数の制約が与えられることはないものとする。

時間制限

入力に対して、実行時間が5秒を超えてはならない。

出力

先頭から最後尾までの距離を1行に出力する。

入出力例

入力例 1	出力例 1
3 2 1<=2-1 2<=3-2	3

入力例 2	出力例 2
3 3 1<=2-1 2<=3-2 1<=3+4	-1

入力例 3	出力例 3
3 2 1<=2-2 2*3+1	inf