

Committee: 委員会

input ファイル “committee.in”

output 標準出力

ソースファイル committee.c/committee.cpp

時間制限 1 秒 / データ

情報オリンピック日本委員会は上下関係がとても厳格な組織である。委員長は一人で、委員長以外の全ての人間はただ一人の上司を持つ。また、機密性を守るため、組織の人間は自分と直接に関係を持つ人間、つまり自分の直接の上司と直接の部下の顔しか知らない。電子的な手段や公共の手段を利用してやりとりをすることは許されておらず、顔を知らない者同士がやりとりをしたい際は、顔を知っている同士である人間をつたってやりとりをしなければいけない。そして、委員会に属する人間には、一人一人やる気の数値なるものが定まっている。やる気の数値が負である人間が居ることもある。

今、情報オリンピック日本委員会の中で、ある極秘のプロジェクトを立ち上げることとなり、一人以上の人間を選ばなければならなくなった。そのプロジェクトが上手くいくかどうかは、選ばれた人間の人数には関係なく、それらの人間のやる気の数値の合計によると考えられる。ただし、プロジェクトは極秘なので、プロジェクト内の任意の人間同士がやり取りをする際、プロジェクト外の人間を通さずにやり取りができるようにしなければいけない。

入力としてそれぞれの人間の上司とやる気の数値が与えられた時、条件を満たす選び方の、やる気の合計の最大値を答えるプログラムを作成せよ。

Input. 入力ファイル committee.in の 1 行目は 1 つの整数 n ($n \leq 100,000$) が書かれている。これは情報オリンピック日本委員会の人数が n 人であることを表す。

次の n 行には、それぞれの人間の上司とやる気の数値が書かれている。 $i+1$ 行目 ($1 \leq i \leq n$) には 2 つの整数 s_i, a_i ($0 \leq s_i < i, -100 \leq a_i \leq 100$) が空白を区切りとして書かれている。これは、人間 i の上司が人間 s_i であり、人間 i のやる気の数値が a_i であることを表す。 s_i が 0 の時、人間 i は委員長であることを表す。 $s_i < i$ であることから、ある人間の上司は必ずその人間の番号より若い番号を持つ。

Output. 出力は、標準出力に行うこと。やる気の合計の最大値を表す 1 つの整数を出力せよ。

例

committee.in 標準出力

5 15

0 10

1 5

2 -8

1 -15

4 3

Sheet: 色紙

input ファイル “sheet.in”

output 標準出力

ソースファイル sheet.c/sheet.cpp

時間制限 1 秒 / データ

紙 1, 紙 2, ..., 紙 N の N 枚の長方形の紙があり, 色はすべて異なっている. これらの紙のすべてと異なる色の長方形のテーブルの上に, N 枚の紙を一枚ずつ順に次のように置く.

- 紙を置くときは, その辺がテーブルの辺と平行になるように置く.
- すでに置いてある紙の下に, 新しく置く紙をもぐりこませるようなことはしない (つまり, 後に置いた紙の方が常に上にある).
- テーブルや紙の辺の長さは整数であり, また, 紙は必ずテーブルの各辺から整数の長さだけ離して置く.

このようにして N 枚の紙を置いた後, 上から写真を撮った. この写真が与えられたとき, 写真と矛盾しない紙の置かれた順番を 1 つ出力するプログラムを書け.

Input. 入力ファイル sheet.in から入力を読み込め. 入力の 1 行目は, 3 つの整数 N, W, H ($1 \leq N \leq 1000, 1 \leq W, H \leq 100$) からなる. これらは順に, 紙の枚数, 机の横幅, 机の縦幅をあらわす. 入力の $i+1$ 行目 ($1 \leq i \leq H$) には, W 個の整数 $C_{i1}, C_{i2}, \dots, C_{iW}$ が書かれている. これらは, 机を 1×1 のマス目 WH 個に区切って考えたとき, 上から i 番目, 左から j 番目のマス目の色が紙 C_{ij} の色と同じであることをあらわす. ただし, $C_{ij} = 0$ のときは, そのマス目の色が机の色であることを表す.

Output. 出力は, 標準出力に行うこと. 出力は N 個の整数からなる. 紙が置かれた順番として考えられるものを 1 つ, 置かれた順に空白区切りで 1 行に出力せよ. 複数の順番が考えられる場合は, そのうちのどれを出力してもよい.

例

sheet.in	標準出力
4 4 4	3 4 1 2
0 1 3 0	
2 2 2 2	
4 1 4 0	
0 3 3 0	

Flu: インフルエンザ

input ファイル “flu.in”
output 標準出力
ソースファイル flu.c/flu.cpp
時間制限 1 秒 / データ

今年 JOI 国でインフルエンザの大流行の兆しがある。JOI 国には n 個の都市 P_1, \dots, P_n があり、都市 P_i の場所は座標を用いて $P_i(x_i, y_i)$ と表される。異なる都市が同じ座標を持つことはない。すなわち、 $i \neq j$ であれば、 $(x_i, y_i) \neq (x_j, y_j)$ である。また、都市 P_1 は首都の JOI 市である。

今年のインフルエンザは特別な型であり、次のような特徴がある。

1. 各都市において、インフルエンザの流行はちょうど m 日間続く。すなわち、もし都市 P_i でインフルエンザの流行が始まったとすると、 P_i におけるインフルエンザの流行はちょうど m 日間続き、 m 日後にはインフルエンザの流行は収まる。
2. ある都市で一度インフルエンザが流行すると、その都市の市民の体内に抗体ができる。そのため、インフルエンザが 1 つの都市で 2 回以上流行することはない。
3. インフルエンザは日を追うごとに都市から都市へと流行を拡大する。その感染可能距離は d である。もし、都市 P_i でインフルエンザの流行が始まったとする。都市 P_j と P_i の距離が d 以下であり、まだ P_j ではインフルエンザは流行していない場合、その翌日には、都市 P_j でインフルエンザの流行が始まる（都市 $P_i(x_i, y_i)$ と $P_j(x_j, y_j)$ の間の距離は $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ である。）
4. 上に述べた以外の経路でインフルエンザの流行が拡大することはない。

また、各 $i = 1, \dots, n$ に対し、都市 P_i からの距離が d 以下であるような P_i 以外の都市の数は 10 個以下である。

ある日、インフルエンザの流行が、JOI 市 P_1 で始まった。JOI 市以外の都市では、今年はまだインフルエンザは流行していない。JOI 国では、治療に必要なワクチンを効率よく手配するため、インフルエンザの流行状況を予測することにした。

入力として、都市の数 n 、1 つの都市でインフルエンザの流行が続く日数 m 、インフルエンザの感染可能距離 d 、整数 $k \geq 1$ 、各都市の座標が与えられたとき、JOI 市で流行が始まってから k 日後において、JOI 国内でインフルエンザが流行している都市の数を求めるプログラムを作成せよ。

Input. 入力ファイル `flu.in` の 1 行目には, JOI 国の都市の数 n ($1 \leq n \leq 100,000$) が書かれている. 2 行目には, 各都市で流行が続く日数 m ($1 \leq m \leq 100$) が書かれている. 3 行目には感染可能距離 d ($1 \leq d \leq 25$) が書かれている. 4 行目には整数 k ($1 \leq k \leq 100$) が書かれている. 続く n 行 (5 行目 ~ $n+4$ 行目) は都市 P_1, \dots, P_n の位置を表す. $i+4$ 行目 ($1 \leq i \leq n$) には, 2 つの整数 x_i, y_i ($0 \leq x_i, y_i < 1,000$) が空白を区切りとして書かれており, 都市 P_i の座標が $P_i(x_i, y_i)$ であることを意味する.

Output. 出力は, 標準出力に行うこと. JOI 市で流行が始まってから k 日後において, JOI 国内でインフルエンザが流行している都市の数を表す整数を 1 行で出力せよ.

採点基準 採点に用いる入力データのうち, 配点の 20% 分は $n \leq 1000, k \leq 50, 0 \leq x_i, y_i < 100$ をみたとす. また, これとは別の配点の 30% 分は $n \leq 10000, 0 \leq x_i, y_i < 1000$ をみたとす.

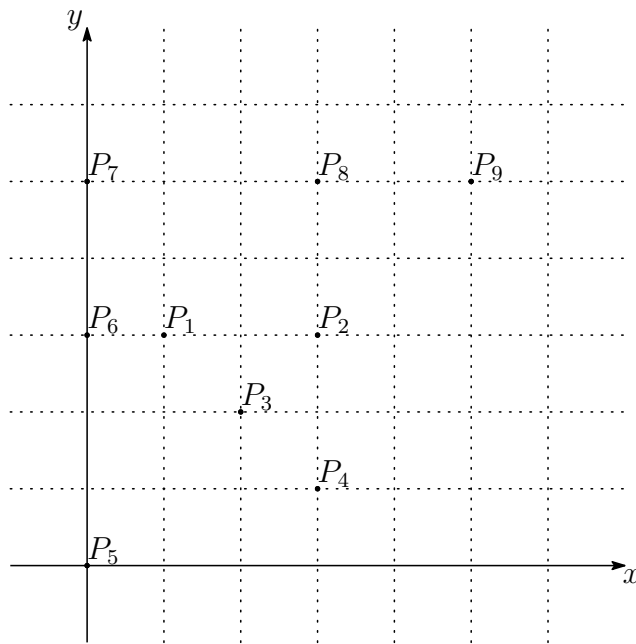
例

<code>flu.in</code>	標準出力
9	4
2	
2	
3	
1 3	
3 3	
2 2	
3 1	
0 0	
0 3	
0 5	
3 5	
5 5	

この入力例において, 都市の数は $n = 9$ である. また, 9 つの都市 P_1, \dots, P_9 の座標は順に

$P_1(1, 3), P_2(3, 3), P_3(2, 2), P_4(3, 1), P_5(0, 0), P_6(0, 3), P_7(0, 5), P_8(3, 5), P_9(5, 5)$

で与えられる. インフルエンザの感染可能距離は $d = 2$ であり, 各都市で流行が続く日数は $m = 2$ である. また, $k = 3$ である.



最初、インフルエンザが流行しているのはJOI市 P_1 のみである。1日後には、インフルエンザの流行は、 P_1 からの距離が2以下の都市 P_2, P_3, P_6 に拡大する。また、JOI市 P_1 における流行はまだ収まらない。2日後には、流行は P_4, P_7, P_8 に拡大し、JOI市 P_1 における流行は収まる。3日後には、流行は P_9 に拡大し、 P_2, P_3, P_6 における流行は収まる。

したがって、JOI市で流行が始まってから3日後において、インフルエンザが流行している都市は P_4, P_7, P_8, P_9 の4つであるから、4を出力すればよい。

この入力例では、都市 P_5 においてインフルエンザが流行することはない。