Problem A
雅先生の地球侵略日誌

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

クァークゴ帝国遠征軍は、地球侵略をもくろむ悪の組織である。彼らは侵略者の伝統にのっとり、日本の東京周辺をターゲットに、毎週１体のペースで怪人を送りこみ続けていた。しかし、そのたびに人類戦隊アースファイブと名乗る５人の戦士があらわれ、街で暴れまわっていた怪人はあっさりと倒されてしまうのであった。

ワルザード・スルー（地球名：源雅）は、そんな現状を本気で憂いている、クァークゴ帝国遠征軍の女幹部である。彼女は、毎週の敗北から何も学ぼうとしない司令官や、どこかズレた発明を繰り返す天才科学者らの下で、頭の痛い日々を過ごしていた。

そんな中、次の作戦は吸血怪人ドラキュラスを日本に送りこむことに決まった。ドラキュラスは、血を吸った人間をドラキュラスに変えてしまう恐るべき怪人である。ドラキュラスに血を吸われた人間もまた、他の人間の血を吸って仲間のドラキュラスを増やしていく。こうして地球全土をドラキュラスで埋めつくしてしまおうという作戦だ。

これを聞いたワルザードは、すぐに悟った。この作戦は、幼稚園バスの乗っ取りのような普段やっているセコい作戦とは訳が違う。あのダメ司令官が立案したものにしては珍しく、本当に地球を征服できる可能性を秘めていると。

倍々ゲームの勢いは恐ろしい。地上に降りたったドラキュラスは、またたく間に仲間の数を増やしていった。このまま行けば地球侵略は目と鼻の先だと思われた。だがその瞬間、ワルザードの脳裏を、強烈にいやな予感が走り抜けた。まさか、この怪人、オリジナルが倒されると仲間も全滅するなんてことはないよな？

慌ててドラキュラスを設計開発した科学者を問いつめたところ、やはりワルザードの危惧していた通りだった。オリジナルのドラキュラスがやられると、血を吸われた人間はすべて元に戻るように設計してあるのだという。ふざけんなよ開発者。どうしてそんな余計な機能をつけた！

ワルザードは、開発担当者に飛び膝蹴りをきめると、すぐさまオリジナルの回収作業にとりかかった。いくらオリジナルと偽物は見た目がまったく同じだといっても、このままでは、駆けつけたアースファイブに何かしらの理由でオリジナルを見抜かれて敗北するのが目に見えている。

開発者の話によると、ドラキュラス化した人間の重さはすべて同じだが、オリジナルのドラキュラスはそれよりも少しだけ重いらしい。それならば、天秤ばかりを使えばオリジナルを見つけることがで

きるはずだ。アースファイブがあらわれる前に、できるだけ早くオリジナルのドラキュラスを見つけて回収しなければ。

► Input

N

入力の 1 行目には、整数 N（$2 \leq N \leq 2,000,000,000$）が書かれている。これは、オリジナルと偽物を合わせたドラキュラスの数をあらわす。

► Output

天秤ばかりを使って N 体のドラキュラスの中から 1 体のオリジナルを見つけるには、最悪の場合でも天秤ばかりを何回使えれば十分か。その最小値を出力せよ。ただし、天秤ばかりの左と右の皿に、それぞれ何体かのドラキュラスを乗せて重さを比較することを、1 回とカウントする。

► Sample Input and Output

Input #1:
```
8
```

Output #1:
```
2
```

Input #2:
```
30
```

Output #2:
```
4
```

Input #3:
```
2000000000
```

Output #3:
```
20
```

Problem B
友だちの誘い方

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

明日から、待ちに待った夏休みが始まります。なのでわたしは、友だちを誘って、海に遊びに行くことに決めました。

けれども、わたしの友だちには、恥ずかしがりやさんが多いです。その人たちは、あまり多くの人がいっしょに来ると知ったら、きっと嫌がるでしょう。

ほかにも、わたしの友だちには、目立ちたがりやさんも多いです。その人たちは、いっしょに来る人があまり多くないと知れば、きっと嫌がるでしょう。

それと、わたしの友だちには、いつもは目立ちたがりやなのに実は恥ずかしがりやさんな人もいます。その人たちは、いっしょに来る人が多すぎても少なすぎても、きっと嫌がるでしょう。

こういうのは、大勢で行った方が楽しいはずです。だからわたしは、できるだけたくさんの友だちを誘いたいと思っています。けれども、嫌がる友だちを無理やり連れていくのはよくありません。

いったい、わたしは最大で何人の友だちを誘うことができるのでしょうか。

わたしは、こういう頭を使いそうな問題が大の苦手です。なので、あなたにお願いがあります。もしよろしければ、わたしの代わりにこの問題を解いていただけないでしょうか。いえ、決して無理にとは言いません。けれど、もし解いていただるのでしたら、わたしはとても嬉しいです。

► Input

N

a_1 b_1

a_2 b_2

$\vdots$

a_N b_N

入力の 1 行目には、整数 N（$1 \le N \le 100,000$）が書かれている。これは、友だちの数をあらわす。

続く N 行には、整数 a_i と整数 b_i（$2 \leq a_i \leq b_i \leq 100,001$）が、空白区切りで書かれている。1 + i 行目に書かれた整数 a_i と b_i は、i 番目の友だちは海に行く人数が a_i 人以上 b_i 人以下でないと嫌がることをあらわす。海に行く人数には「わたし」も含まれることに注意せよ。

► Output

嫌がる友だちが出ないように、海に誘うことのできる友だちの最大人数を出力せよ。

► Sample Input and Output

Input #1:
```
4
2 5
4 7
2 4
3 6
```

Output #1:
```
3
```

Input #2:
```
5
8 100001
7 100001
12 100001
8 100001
3 100001
```

Output #2:
```
0
```

Input #3:
```
6
2 9
4 8
6 7
6 6
5 7
2 100001
```

Output #3:
```
5
```

Problem C

時空のスゴロク・ロード

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

全時空統一ディメンション・スゴロク・トーナメント。500 兆人を超える参加者の中から、ただ１人のスゴロク絶対王者を決定するその大会に、あなたは２１世紀の地球代表として参加している。

今あなたが挑戦している課題は、自分自身をコマとした１次元スゴロク。端のスタートマスから出発して、１から６までの目が一つずつ書かれた巨大な６面ダイスを振って出た目の数だけ進むことを繰り返す、あなたもよく知っている形式のスゴロクだ。スタートマスとは反対の端にあるゴールマスに止まるとゴールとなる。もちろん、ゴールするまでにサイコロを振った回数が少なければ少ないほど良い成績となる。

マスの中には特殊な効果を持ったマス「○マス進む」と「○マス戻る」が存在し、そこに止まってしまうと、指定されたマス数だけ進む、もしくは戻らなければならない。マスの効果で動いた結果、ふたたび効果のあるマスに止まった場合は、続けて指示どおりに移動する。

しかし、これは一筋縄では攻略できない時空スゴロクだ。恐るべきことに、たとえば「３マス進む」の３マス先に「３マス戻る」が置かれていることもありうる。このようなマスに止まり、マスの効果で無限ループに陥ってしまった場合は、永遠にマスを往復し続けなければならない。

だが幸いなことに、あなたの身体には、望んだ事象を全事象に変えることのできる異能『確率湾曲』が宿っている。この能力を使えば、サイコロの出目を自由に操ることも可能。このアドバンテージを活かして、無限ループに陥らないようにしながら進んでいくとき、ゴールするまでにサイコロを振る回数の最小値はいくつになるだろうか。

► Input

N

p_1

p_2

$\vdots$

p_N

入力の１行目には、整数 N（$3 \leq N \leq 100,000$）が書かれている。これは、スゴロクのマス数をあらわ

す。マスには 1 番から N 番までの番号がふられている。スタートのマスが 1 番で、それからスタートに近い順に 2 番、3 番、……、$N-1$ 番と続き、ゴールのマスが N 番である。i 番のマスでサイコロを振って j の目が出たら、$i+j$ 番のマスへと移動する。ただし、もしも $i+j$ が N を超えていたら、余った数だけ戻ったりはせず、ゴールしたとみなされる。

続く N 行には、整数 p_i（$-100,000 \le p_i \le 100,000$）が書かれている。1＋i 行目に書かれた整数 p_i は、i 番のマスに書かれている指示をあらわす。$p_i > 0$ ならば「p_i マス進む」、$p_i < 0$ ならば「$-p_i$ マス戻る」であり、$p_i = 0$ ならばそのマスには何の効果もない。p_1 と p_N は必ず 0 である。マスの効果で、スタートより前やゴールより後に移動しろと指示されることはない。

なお、与えられるスゴロクは、ゴールすることが可能であると仮定してよい。

► Output

ゴールするまでにサイコロを振る回数の最小値を出力せよ。

► Sample Input and Output

Input #1:

```
11
0
0
-2
0
-4
1
-1
2
0
0
0
```

Output #1:

```
3
```

Input #2:

```
12
0
0
7
0
2
0
0
3
-6
-2
1
0
```

Output #2:

```
1
```

Input #3:

```
8
0
4
-1
1
-2
-2
0
0
```

Output #3:

```
2
```

Problem D
僕の友達は小さい

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

僕には、たくさんの友達がいる。どの友達も、とても小さい。

僕は、よく友達と一緒に出かける。何人かの友達をリュックに入れて、一緒に出かける。

僕は毎朝、その日一緒に出かける友達を決める。空のリュックに、１人ずつ友達を入れていく。

僕は、あまり力が強くない。だから、同時に運べる友達の重さには限界がある。

僕は、重さの限界を超えないように友達を入れていく。どの順番で入れていくかは気分次第。

僕は、入れられる友達がまだ残っている限り、入れるのを止めない。決して止めない。

……ところで、リュックに入った友達の組み合わせは、全部で何パターンあるんだろう？

► Input

N W

w_1

w_2

$\vdots$

w_n

入力の１行目には、整数 N（$1 \le N \le 200$）と整数 W（$1 \le W \le 10,000$）が、この順に空白区切りで書かれている。整数 N は友達の数を、整数 W は同時に運べる重さの限界をあらわす。重さの合計が W より大きいと運ぶことはできない。

続く N 行には、友達の重さをあらわす整数が書かれている。整数 w_i（$1 \le w_i \le 10,000$）が、i 番目の友達の重さをあらわす。

► Output

最終的にリュックに入っている友達の組み合わせは何通り考えられるか。その総数を $1,000,000,007$ で割った余りを出力せよ。なお $1,000,000,007$ は素数である。

「誰もリュックに入っていない」も１通りとして数えることに注意せよ。

► Sample Input and Output

Input #1:

```
4 8
1
2
7
9
```

Output #1:

```
2
```

Input #2:

```
4 25
20
15
20
15
```

Output #2:

```
4
```

Input #3:

```
6 37
5
9
13
18
26
33
```

Output #3:

```
6
```

Problem E
街を駆ける道

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

ネーヴァ王国には、トタタ族とツテテ族、２種類の民族が暮らしている。トタタ族の最大の特徴は、酢豚にパイナップルを入れて食べることである。だがツテテ族は、パイナップルに酢豚を入れて食べる。こんな２つの民族がお互いに仲良くやっていけるはずもなく、トタタ族とツテテ族は、何百年もの昔からずっといがみ合いを続けてきた。

そんなある日、ネーヴァ王のもとに、２つの民族から嘆願書が届いた。それによるとトタタ族は、自分たちが暮らす街Ａと街Ｂのあいだを結ぶ道を建設してほしいらしい。一方でツテテ族も、自分たちが暮らす街Ｃと街Ｄのあいだを結ぶ道を建設してほしいらしい。

２つの民族が衝突するのを防ぐため、トタタ族が通る道とツテテ族が通る道を交差させることはできない。また、技術的な制約により、２つの街のあいだを一直線に結ぶような道しか建設することはできない。つまり、必要ならば街Ａと街Ｂを直接道で結ばずに、いくつかのトタタ族の街を経由して街Ａと街Ｂを間接的に結ぶことになるだろう（もちろん、ツテテ族の街を経由してはならない）。その際、トタタ族の街を結ぶ道どうしは交差していてもよい。街Ｃと街Ｄについても同様である。

道を建設するには、その長さに比例したコストがかかる。なので、条件をみたしつつ、できるだけ建設する道の長さの合計が短くなるようにしたい。さて、長さの最小値はいくつになるだろうか。

► Input

N_A N_B

$x_{A,1}$ $y_{A,1}$

$x_{A,2}$ $y_{A,2}$

$\vdots$

x_{A,N_A} y_{A,N_A}

$x_{B,1}$ $y_{B,1}$

$x_{B,2}$ $y_{B,2}$

$\vdots$

x_{B,N_B} y_{B,N_B}

入力の 1 行目には、整数 N_A（$2 \le N_A \le 1,000$）と整数 N_B（$2 \le N_B \le 1,000$）が、空白区切りで書かれている。これは、ネーヴァ王国にはトタタ族が暮らす街が N_A 個、ツテテ族が暮らす街が N_B 個あることをあらわす。初期状態では、どこにも道は建設されていない。

続く N_A 行には、整数 $x_{A,i}$（$-10,000 \le x_{A,i} \le 10,000$）と整数 $y_{A,i}$（$-10,000 \le y_{A,i} \le 10,000$）が、空白区切りで書かれている。ネーヴァ王国の地理は 2 次元直交座標平面であらわされ、1 + i 行目に書かれた整数 $x_{A,i}$ と $y_{A,i}$ は、トタタ族が暮らす i 番目の街の位置座標が $(x_{A,i}, y_{A,i})$ であることをあらわす。$(x_{A,1}, y_{A,1})$ と $(x_{A,2}, y_{A,2})$ が、結ぶべき 2 つの街の座標である。

続く N_B 行には、整数 $x_{B,i}$（$-10,000 \le x_{B,i} \le 10,000$）と整数 $y_{B,i}$（$-10,000 \le y_{B,i} \le 10,000$）が、空白区切りで書かれている。1 + N_A + i 行目に書かれた整数 $x_{B,i}$ と $y_{B,i}$ は、ツテテ族が暮らす i 番目の街の位置座標が $(x_{B,i}, y_{B,i})$ であることをあらわす。$(x_{B,1}, y_{B,1})$ と $(x_{B,2}, y_{B,2})$ が、結ぶべき 2 つの街の座標である。

どの 2 つの街の座標も異なっており、どの 3 つの街も同一直線上にないと仮定してよい。

▶ Output

問題文の条件をみたすように道を建設したとき、道の長さの合計の最小値を出力せよ。ここで言う「長さ」とは、ユークリッド距離のことである。ただし、どうやっても条件をみたす道を建設できないときは、代わりに -1 を出力せよ。出力は誤差を含んでいてもよいが、真の値との相対誤差が 10^{-9} 未満でなければならない。

▶ Sample Input and Output

Input #1:

```
2 2
0 0
1 1
2 0
2 -1
```

Output #1:

```
2.414213562373
```

Input #2:

```
2 3
4 0
0 0
2 3
2 -2
3 1
```

Output #2:

```
-1
```

Input #3:

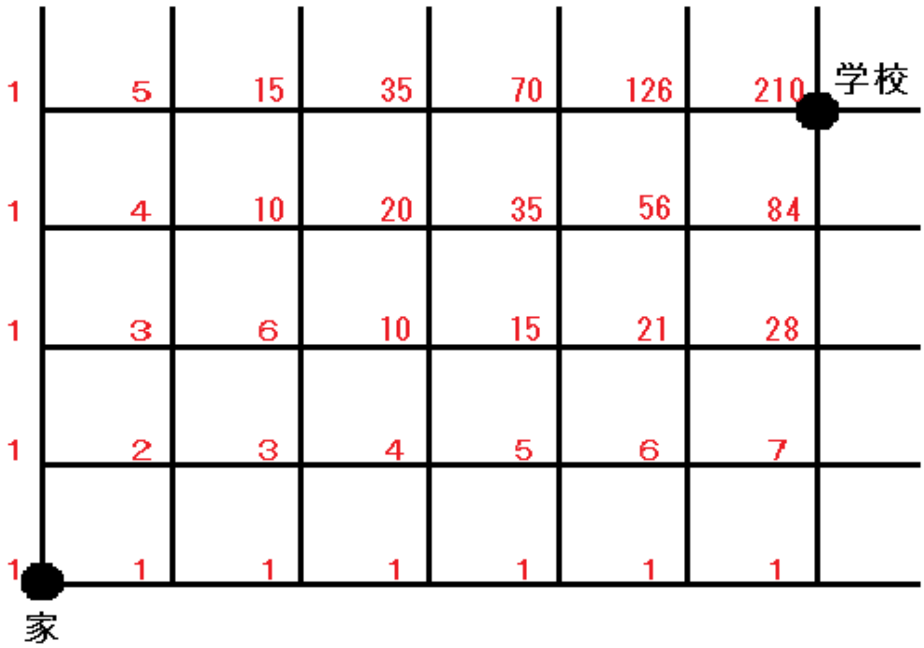
```
5 2
-2 1
1 2
-1 3
-1 5
1 4
0 4
0 0
```

Output #3:

```
12.359173603117
```

Problem F
１０歳の動的計画

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010



ある日の夕方。いつものようにあなたが居間でテレビを見ていると、小学５年生の妹から相談を持ちかけられた。話を聞いてみると、今日学校で出された算数の問題が難しくて理解できなかったので、あなたに解き方を教えてほしいのだという。

妹を悩ませている問題は、「道が碁盤目状に通っている街において、家 $(0,0)$ から学校 (N,M) まで最短距離で進む方法は何通りあるか？」というものだった。もちろん、あなたにとっては赤子の手をひねるよりも簡単な問題である。すぐに上のような図を書いて「家 $(0,0)$ から順番に足し算していけば解けるよ」と教えてあげた。

ところが、それを聞いてもまだ、妹はなにやら下を向いて考えこんでいる。あなたは、自分の説明の仕方が悪かったのだろうかと思ったが、どうやらそうではないらしい。

たっぷり３分ほど悩んでいただろうか。妹は、おもむろに顔を上げると、あなたに向かってこう言った。

「でも、わたしだったら、きっと学校に行く途中で K 回くらい寄り道しちゃうだろうな……。ねぇお兄ちゃん、そうすると答えは何通りになるの？」

さあ大変だ。兄の威厳を保つため、なんとしてもこの問題に答えねば。

この問題を定式化すると、次のようになる。

・ 点 $(0, 0)$ から点 (N, M) まで、碁盤目状の道を通って進む方法は何通りあるか答えよ。

・ 基本的には右か上に 1 マス進むが、途中で、ちょうど K 回だけ寄り道をする。

・ 「寄り道をする」とは、左か下に 1 マス進むことである。

・ K 回の寄り道をすませていない場合、いったん点 (N, M) に着いた後でも歩き続ける。途中で点 $(0, 0)$ に戻ってくる場合もある。

・ 家はこの街の隅に建っているので、X 座標または Y 座標が負になるような点に入ることはできない。

・ しかし、X 座標が N より大きな点、または、Y 座標が M より大きな点には入ることができる。

► Input

N M K

入力の 1 行目には、整数 N（$1 \le N \le 100,000$）と整数 M（$1 \le M \le 100,000$）と整数 K（$0 \le K \le 10,000$）が、この順に空白区切りで書かれている。整数 N は学校の X 座標を、整数 M は学校の Y 座標を、整数 K は寄り道の回数をあらわす。

► Output

点 $(0, 0)$ から点 (N, M) まで、K 回の寄り道をして進む方法。その総数を $1,000,000,007$ で割った余りを出力せよ。なお $1,000,000,007$ は素数である。

► Sample Input and Output

Input #1:
```
6 4 0
```
Output #1:
```
210
```

Input #2:
```
3 3 1
```
Output #2:
```
448
```

Input #3:
```
124 218 367
```
Output #3:
```
817857665
```

Problem G
スプリング・タイル

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

ある朝、あなたが目覚めると、そこはバネだらけの迷宮の中だった。

なぜ自分がこんな見知らぬ場所にいるのかはわからない。この場でじっと助けを待つという選択肢もあるにはあるが、こういった迷宮の中に長居していると、いつかは必ず突風が吹いてきて吹き飛ばされてしまうことを、あなたは経験上知っていた。しかし、突風が吹いてくるまでの時間は判断のしようがない。そこであなたは、迷宮から脱出するまでに必要な移動回数の期待値が最小になるように行動するのが最善の戦略だと考えた。

足元に落ちていた正体不明の巻物を拾ってためしに読んでみたところ、偶然にも迷宮の全体マップを知覚することができた。さらに、たまたま持っていた草を飲むことで、すべてのワナの位置も知ることができた。どうやらこの迷宮には、危険なモンスターやバネ以外のワナは存在しないらしい。

この迷宮は、いくつかの種類のタイルが長方形に並んだ形をしており、たとえば以下のようにあらわされる。

```
########
#..##g.#
#*#.*#.#
#......#
#*s#*.*#
########
```

「.」：床。この上なら自由に歩き回ることができる。

「#」：壁。あなたは壁の中に入ることはできない。

「s」：あなたの最初の位置。このタイルの下は床になっている。

「g」：階段。このタイルの上に乗れば、迷宮を脱出したことになる。

「*」：バネ。このタイルの上に乗ると、いずれかの床の上（階段、バネのタイルは除く）にランダムに飛ばされる。それぞれの床に飛ばされる確率はすべて等しい。

タイルの種類は、この５つのいずれかである。

あなたは、現在乗っているタイルから隣り合ったタイルへ、上下左右いずれかの方向に１マスずつ移動することができる。ただし、ナナメに移動することはできない。

迷宮のマップが与えられたとき、最善の戦略をとった場合に迷宮から脱出するまでに必要な移動回数の期待値を求めよ。バネで飛ばされるのは移動回数には含めない。

▶ Input

$W\ H$

$$
\begin{matrix}
c_{11} & c_{12} & \ldots & c_{1w} \\
c_{21} & c_{22} & \ldots & c_{2w} \\
\vdots & \vdots & \ddots & \vdots \\
c_{h1} & c_{h2} & \ldots & c_{hw}
\end{matrix}
$$

入力の１行目には、整数 W（$3 \le W \le 500$）と整数 H（$3 \le H \le 500$）が、この順に空白区切りで書かれている。整数 W は迷宮の横幅を、整数 H は迷宮の縦幅をあらわす。

続く H 行には、迷宮のマップをあらわす W 個の文字が書かれている（空白区切りではない）。この部分の形式は、先に挙げた例のとおりである。なお、データ中に「s」と「g」は、それぞれちょうど１回ずつ出現する。また、迷宮の周囲１マスは、必ず壁で囲まれている。

なお、与えられる迷宮では、どのように移動し、バネでどのように飛ばされたとしても、「バネで飛ばされる床を任意に設定できたとしても脱出不可能な状態」に陥ることはないと仮定してよい。

▶ Output

最善の戦略をとったときに、脱出までにかかる移動回数の期待値を出力せよ。出力は誤差を含んでいてもよいが、真の値との相対誤差が 10^{-9} 未満でなければならない。

▶ Sample Input and Output

Input #1:

```
8 6
########
#..##g.#
#*#.*#.#
#......#
#*s#*.*#
########
```

Output #1:

```
5.857142857143
```

Input #2:

```
21 11
#####################
#*#*.*.*.*.*.*.*.*#*#
#*#...............#*#
#*#.#.#.#.#.#.#.#.#*#
##.#.#.#.#.#.#.#.#.##
#...................#
##.#.#.#.#.#.#.#.#.##
#s#.#.#.#.#.#.#.#.#g#
#...................#
#...................#
#####################
```

Output #2:

```
20.000000000000
```

Input #3:

```
9 8
#########
#...#*.*#
#.*.#...#
#...#*.*#
#########
#.s....g#
#.*#.*..#
#########
```

Output #3:

```
5.000000000000
```

Problem H
ねこ鍋改造計画（仮）

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

あなたは、自宅にたくさんのねこを飼っている。そんなあなたの日課は、飼っているねこが土鍋の中に入って昼寝している様子（通称：ねこ鍋）を撮影することだ。何匹かのねこが、鍋の中でいっしょに丸くなって眠る姿は、実に愛くるしい。

それぞれのねこには、「重さ」m_i と「Cute」c_i が定義されている。Cute とは、そのねこ 1 匹だけを見たときの、愛らしさ・あどけなさ・しなやかさ・やわらかさ・ういういしさ、などを総合的に評価した数値であり、大きければ大きいほどそのねこは魅力的である。

また、ねこ鍋に対しては「UnCute」という数値が定義される。これは、ねこ鍋全体としてのアンバランスさをあらわす値であり、Cute とは逆に、小さければ小さいほど、そのねこ鍋はよいねこ鍋である。ねこ鍋の UnCute は、中にいるねこの Cute の最大値 C_{max} と最小値 C_{min} を用いて、$C_{max} - C_{min}$ とあらわされる。つまり、中にいるねこがみな同じくらいの Cute を持っているねこ鍋が、よいねこ鍋である。

ねこ鍋を持ち運ぶときのことを考えると、ねこ鍋の重さ（＝中にいるねこの重さの総和）は、限界値 W 以下でなければならない。この条件をみたすようにねこ鍋に入ってもらうねこを選んで、ねこ鍋の UnCute を小さくしたいのだが、いったいどこまで小さくできるのだろうか。

……という問題を解いてすっかり満足した気になっていたのは、過去の話だ。

この前、あなたが家の倉庫を整理していたら、倉庫の中から古ぼけた土鍋が見つかった。これであなたの家にある土鍋は２つ。そう、これからは、デュアルねこ鍋の時代なのだ！　ただのねこ鍋などもう古い！

デュアルねこ鍋では、土鍋 A はオスのねこ専用、もう片方の土鍋 B はメスのねこ専用である。また、デュアルねこ鍋の UnCute を定義するにあたっては、Cute のアンバランスさだけではなく、２つのねこ鍋の重さのアンバランスさも考慮に入れる必要がある。具体的には、重いほうのねこ鍋の重さを M_{max}、軽いほうのねこ鍋の重さを M_{min} とおくと（２つのねこ鍋の重さが同じときは $M_{max} = M_{min}$）、デュアルねこ鍋の UnCute は、

$$\max\{M_{max} - M_{min}, C_{max} - C_{min}\}$$

とあらわされる。なお、ここでの C_{max} と C_{min} は、ねこ鍋 A の中またはねこ鍋 B の中にいるねこの Cute の最大値・最小値である。

ねこ鍋を持ち運ぶときのことを考えると、M_{max} は、限界値 W 以下でなければならない。この条件をみたすようにデュアルねこ鍋に入ってもらうねこを選んで、デュアルねこ鍋の UnCute を小さくしたいのだが、いったいどこまで小さくできるのだろうか。

► Input

N_A N_B W

$m_{A,1}$ $c_{A,1}$

$m_{A,2}$ $c_{A,2}$

$\vdots$

m_{A,N_A} c_{A,N_A}

$m_{B,1}$ $c_{B,1}$

$m_{B,2}$ $c_{B,2}$

$\vdots$

m_{B,N_B} c_{B,N_B}

入力の 1 行目には、整数 N_A（$1 \leq N_A \leq 500$）と整数 N_B（$1 \leq N_B \leq 500$）と整数 W（$1 \leq W \leq 10,000$）が、空白区切りで書かれている。これは、あなたが飼っているオスのねこが全部で N_A 匹、メスのねこが全部で N_B 匹いることをあらわす。W はねこ鍋の重さの限界値である。

続く N_A 行には、整数 $m_{A,i}$（$1 \leq m_{A,i} \leq 10,000$）と整数 $c_{A,i}$（$1 \leq c_{A,i} \leq 1,000,000,000$）が、空白区切りで書かれている。1 + i 行目に書かれた整数 $m_{A,i}$ と $c_{A,i}$ は、i 番目のオスのねこの重さが $m_{A,i}$、Cute が $c_{A,i}$ であることをあらわす。

続く N_B 行には、整数 $m_{B,i}$（$1 \leq m_{B,i} \leq 10,000$）と整数 $c_{B,i}$（$1 \leq c_{B,i} \leq 1,000,000,000$）が、空白区切りで書かれている。1 + N_A + i 行目に書かれた整数 $m_{B,i}$ と $c_{B,i}$ は、i 番目のメスのねこの重さが $m_{B,i}$、Cute が $c_{B,i}$ であることをあらわす。

オスのねこにもメスのねこにも、重さが W 以下であるようなねこが、それぞれ 1 匹以上は存在すると仮定してよい。

► Output

M_{max} が W を超えないという条件のもとで、デュアルねこ鍋の UnCute の最小値を出力せよ。ただし、どちらのねこ鍋にも 1 匹以上のねこが入っていなければならない。

► Sample Input and Output

Input #1:

```
4 3 12
3 6
2 4
7 9
10 1
6 5
8 4
15 19
```

Output #1:

```
2
```

Input #2:

```
1 3 10
1 15
6 8
5 9
8 7
```

Output #2:

```
6
```

Input #3:

```
8 6 65
30 98
27 51
4 74
65 87
49 19
27 48
43 7
35 28
43 69
8 47
64 75
18 23
54 29
40 43
```

Output #3:

```
8
```

Problem I
よくわかる二重魔法

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

紀慧暦 2119 年 15 の月。宮廷魔術士 Sengemon Lugene の手によって、一冊の魔術書が書き記された。この魔術書「In Magiray」は、この世界の魔術法則の根幹に深く関わる「二重魔法」を、誰にでも習得可能な技術として体系化した、という点で革新的な書物であった。まずは、この書物の内容を要約したものを見ていこう。

【エレメント】

世界中のあらゆる物体は、「泉」「風」「羽」「花」「砂」「灯」……といった、極小要素「エレメント」から構成されている。エレメントには多くの種類があり、人間はエレメントと契約を交わすことで、契約を交わした種類のエレメントを用いた二重魔法を発動させることができる。

【二重魔法】

二重魔法とは、その名の通り、２つのエレメントを重ね合わせて解き放つ魔法である。主となるエレメントを「主属性」、もう一方を「副属性」と呼ぶ。たとえば、主属性が「灯」で副属性が「花」なら、その二重魔法は（灯，花）と表記される。（灯，花）と（花，灯）は別の二重魔法である。また、主属性と副属性は同じであってもよい。

【キーワード】

「泉→ huda」「風→ loar」「羽→ sheza」「花→ leide」といったように、各々のエレメントに、それぞれ短い「キーワード」を１対１対応させることができる。キーワードは、エレメントの持つ性質を短い言葉に象徴させることで、いくつかの連続するキーワード列を唱えるだけで簡単に二重魔法を発動させることを可能にした、画期的な概念である。

【呪文】

「huda・leide・loar・sheza」のように、１つ以上のキーワードを連続して並べたものを「呪文」という。術者が意図した呪文を唱えることで、対応する二重魔法を発動させることができる。具体的には、呪文の一番最初のキーワードに対応するエレメントを主属性、一番最後のキーワードに対応するエレメントを副属性とする二重魔法が発動する。先ほどの呪文の例だと、発動する二重魔法は（泉，羽）である。なお、呪文が１つのキーワードだけから成る場合には、主属性も副属性も、そのキーワードに対応するエレメントになる。

【エレメント同士の相性】

「huda・leide・loar・sheza」の呪文で（泉，羽）の二重魔法が発動することはすでに述べた。しかし、

一見すると、（泉，羽）を使いたいのならば、わざわざ長い呪文を唱えずとも「huda・sheza」だけで良いようにも思える。だが実際には、それは不可能である。なぜなら、「泉（huda）」と「羽（sheza）」は相性が良くないため、これら２つが隣り合った呪文を唱えると、反発作用が起こってエレメント同士が相殺してしまうからである。一方で、「泉（huda）」と「花（leide）」、「花（leide）」と「風（loar）」、「風（loar）」と「羽（sheza）」は、どのペアも好相性なので、「huda・leide・loar・sheza」や「loar・leide・huda」といった呪文では、エレメントの相殺は起こらない。

【呪文論理】

二重魔法を使おうとする者は、あらかじめ、すべての好相性なエレメントのペアそれぞれに対して、どちらのキーワードが「上位ワード」で、どちらのキーワードが「下位ワード」なのかを定義しておかなければならない。好相性な２つのエレメントのキーワード「A」と「B」に対して、A を上位ワード、B を下位ワードと定義したとき、$A > B$ と表記する。$A > B$ であるとき、一つの呪文内で、B の直後に A を並べることはできない。つまり、「A_1・A_2・A_3・……・A_k」が正しい呪文であるならば、$A_1 > A_2$，$A_2 > A_3$，……，$A_{k-1} > A_k$ でなくてはならない。ちなみに、$A > B$ かつ $B > C$ であっても、$A > C$ である必要はない。

このように、各々の術者が、上位・下位ワードを自分好みに定義することを「呪文論理の構築」と呼ぶ。これは、あえて呪文の構文に制約をもたせることで、エレメントの流れを明確化し、目的の二重魔法の成功確率を上げるための工夫である。だが、その代償として、呪文論理を構築すると、どのように呪文を唱えても発動させることのできない二重魔法が生じてしまう可能性がある（ただし、いかなる呪文論理を構築しても使うことのできない二重魔法は存在しないことが証明されている）。このような事情があるため、どのような呪文論理を構築するかは、今も昔も多くの魔術士を悩ませている厄介な問題なのである。

とある小さな町で暮らす見習い魔術士の少年ユートは、すべてのエレメントとの契約を終え、いよいよ明日、自分の呪文論理を構築するための儀式を行おうとしている。ユートは単純に、できるだけたくさんの種類の二重魔法を発動させることができるような呪文論理が良い呪文論理である、と考えていた。

超一級電術技師（現代日本で言うところのスーパープログラマー）であるあなたは、友達のユートから、うまく呪文論理を構築することができれば、最大で何種類の二重魔法を使えるようになるのか知りたいと相談を受けた。これは、今まで世界中の人間が挑戦してきたが、誰も解くことができなかったほどの難問である。もしも解くことができれば、あなたの名前は未来永劫、世界中で語り継がれることになるであろう。

▶ Input

N M

s_1

s_2

$\vdots$

s_N

p_1 q_1

p_2 q_2

$\vdots$

p_M q_M

入力の 1 行目には、整数 N と整数 M が、空白区切りで書かれている。これは、エレメントが全部で N 種類、好相性なエレメントのペアが全部で M 組存在することをあらわす。

続く N 行には、1 つの文字列が書かれている。1 + i 行目に書かれた文字列 s_i は、i 番目のエレメントに対応するキーワードが s_i であることをあらわす。

続く M 行には、2 つの文字列が空白区切りで書かれている。1 + N + i 行目に書かれた文字列 p_i と q_i は、キーワード p_i に対応するエレメントと、キーワード q_i に対応するエレメントのペアが、好相性であることをあらわす。逆に、ここにあらわれなかったエレメントのペアは、好相性ではない。

これらのデータは、以下の条件をみたす。

- $2 \leq N \leq 100$
- $1 \leq M \leq 4,950$
- s_i は、アルファベットの大文字または小文字のみからなる、1 文字以上 15 文字以下の文字列
- $i \neq j \quad \Rightarrow \quad s_i \neq s_j$
- $p_i \neq q_i$
- $i \neq j \quad \Rightarrow \quad (p_i, q_i) \neq (p_j, q_j)$ かつ $(p_i, q_i) \neq (q_j, p_j)$
- いかなる呪文論理を構築しても発動することのできない二重魔法は存在しない

► Output

呪文論理を構築したとき、使用可能になる二重魔法の種類の最大数を出力せよ。

► Sample Input and Output

Input #1:

```
4 3
huda
leide
loar
sheza
huda leide
leide loar
loar sheza
```

Output #1:

```
10
```

Input #2:

```
3 3
torn
siole
dimi
torn siole
torn dimi
siole dimi
```

Output #2:

```
9
```

Input #3:

```
9 10
riris
soa
elah
clar
arma
memori
neoles
qhaon
clue
clar riris
riris memori
riris neoles
soa clar
soa memori
neoles elah
elah qhaon
neoles qhaon
clue riris
arma elah
```

Output #3:

```
54
```

Problem J
問題文担当者は働かない！

Summer Camp 2010 Day 2, Tokyo
18 Sep 2010

N 個の点からなる有向グラフがある。このグラフに閉路はない。それぞれの点には、いくつかの石が置かれている。このグラフを使って、２人のプレイヤーがゲームをする。おのおのの手番は、交互に訪れる。各プレイヤーは、自分の手番において、頂点 v を選び、そこに置かれた石を１つ以上取り除く。その後、v から w に辺が張られているすべての頂点 w について、w に置かれている石の個数を自由に変化させることができる（いくつ増やしても、いくつ減らしても、変化させなくてもよい）。これを繰り返して、最初に取る石がなくなった方の負けである。両者が最善をつくしたとき、先手と後手、どちらが勝つか。もしくは、いつまでもゲームが終わらないか。

「……いや、無理でしょう、これ」
「何が？」
「何がじゃなくて！　これにどうやって自然な問題設定をつけろって言うんですか」
「ははは。それを考えるのが君の仕事じゃないか。文章担当だろう？」
「丸投げにもほどがありますよ……。まずゲームのルールが不自然すぎるでしょう」
「そうかなぁ？　僕はよくある石取りゲームだと思うけどね」
「プレイヤーの意思で石を無尽蔵に増やせる石取りゲームがよくあってたまりますか！」
「お。君、今うまいこと言ったね」
「ええ、意思で石を……ってどうでもいいわ！　そっちも少しは考えてくださいよ」
「そうだねぇ。縊死した医師の遺志を継ぐ遺子の意思で石を取る、なんてどうだい？」
「考えるのはそこじゃねぇよ！　石取りゲームの話だよ！」
「違うのかい？　でも『取り』は鳥くらいしか同音異義語がなくて難しいんだよね」
「いったん言葉遊びから離れろや！　自然な問題設定を考えてくれって言ってるんですよ！」
「だからそれは君の仕事だろうに。僕はただの仲介役。原案担当と文章担当の間をとりもつだけさ」
「くそ……。原案担当め、どうしてこんな厄介な問題を……」
「原案担当者から君へのメッセージ。『フヒヒｗ　存分に苦しめ文章担当ｗｗ』だそうだよ」
「畜生あの野郎！　いつか一発殴る！」
「まあ、僕の『石が増える石取りゲームなんて実に設定がつけにくいと思わないかい？』ってアドバイスのおかげもあるだろうけどね」
「畜生お前もか！　いつかと言わず今殴る！」
「はっはっは。君が締め切りまでに文章担当としての役割を果たせないようなら、僕が勝手に問題文を書いちゃうよ？」
「殴……って、え？　お前が書いてくれんの？」
「タイトルは『問題文担当者は働かない！』。本文には、僕たちのこの会話をそのまま掲載」
「頼むからやめて！　そんなふざけた問題が自分名義で世に出たら、これから皆になんて言われるか！」
「はっはっは。締め切りも守れないような奴は、社会的に抹殺されて当然だよね」
「畜生！　こうなったら、意地でも自然な問題設定を思いついてやらあああああ！」

（※　結局そのまま掲載されました）

► Input

N M

v_1

v_2

$\vdots$

v_N

a_1 b_1

a_2 b_2

$\vdots$

a_M b_M

入力の 1 行目には、整数 N（$2 \leq N \leq 1,000$）と整数 M（$1 \leq M \leq 10,000$）が、空白区切りで書かれている。これは、有向グラフが N 個の点と M 本の辺からなることをあらわす。頂点には、1 番から N 番までの番号がふられている。

続く N 行には、整数 v_i（$1 \leq v_i \leq 10,000$）が書かれている。１＋ i 行目に書かれた整数 v_i は、i 番の点に最初は v_i 個の石が置かれていることをあらわす。

続く M 行には、整数 a_i（$1 \leq a_i \leq N$）と整数 b_i（$1 \leq b_i \leq N$）が、空白区切りで書かれている。１＋ N ＋ i 行目に書かれた整数 a_i と b_i は、a_i 番の点から b_i 番の点へ張られている辺が存在することをあらわす。ある点から自分自身へと張られる辺は存在せず、ある点 A からある点 B へ張られる辺は高々 1 本である。

与えられた有向グラフには閉路がないと仮定してよい。

► Output

両プレイヤーが自身の勝利を目指して最善をつくしたとき、先手が勝つならば 1 を、後手が勝つならば 2 を、いつまでもゲームが終わらないならば 0 を出力せよ。

► Sample Input and Output

Input #1:

```
6 5
7
14
5
11
2
5
1 2
2 3
3 4
4 5
5 6
```

Output #1:

```
2
```

Input #2:

```
5 7
295
127
350
982
426
1 5
3 5
4 2
3 1
3 4
5 4
3 2
```

Output #2:

```
1
```

Input #3:

```
8 7
6
1
7
2
5
2
6
3
1 3
6 3
7 4
4 2
1 6
5 4
7 5
```

Output #3:

```
2
```