

2015 網際網路程式設計全國大賽

高中組決賽

- 本次比賽共 8 題，含本封面共 26 頁。
- 全部題目的輸入都來自**標準輸入**。輸入中可能包含多組輸入，以題目敘述為主。
- 全部題目的輸出皆輸出到螢幕 (**標準輸出**)。
輸出和裁判的答案必須完全一致，英文字母大小寫不同或有多餘字元皆視為答題錯誤。
- 每一題的執行時間限制，請參考 Kattis 上的題目敘述。
- 比賽中上傳之程式碼，使用 C 語言請用 `.c` 為副檔名；使用 C++ 語言則用 `.cpp` 為副檔名。
- 使用 `cin` 輸入速度遠慢於 `scanf` 輸入，若使用需自行承擔 Time Limit Exceeded 的風險。
- 使用 `scanf` 或 `printf` 處理長整數 (`long long int`) 時，請使用 “`%lld`”。
詳細可參閱下頁之輸入輸出範例。
- 部分題目有浮點數輸出，會採容許部分誤差的方式進行評測。一般來說「相對或絕對誤差小於 ϵ 皆視為正確」， ϵ 值以題目敘述為主。
舉例來說，假設 $\epsilon = 10^{-6}$ 且 a 是正確答案， b 是你的答案，如果符合 $\frac{|a-b|}{\max(a,b,1)} \leq 10^{-6}$ ，就會被評測程式視為正確。

2015 網際網路程式設計全國大賽

輸入輸出範例

C 程式範例：

```
1 #include <stdio.h>
2 int main()
3 {
4     int cases;
5     scanf("%d", &cases);
6     for (int i = 0; i < cases; ++i)
7     {
8         long long a, b;
9         scanf("%lld %lld", &a, &b);
10        printf("%lld\n", a + b);
11    }
12    return 0;
13 }
```

C++ 程式範例：

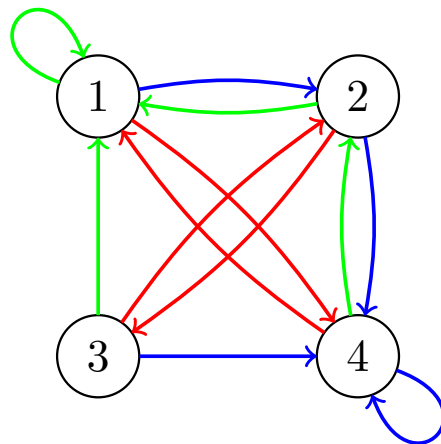
```
1 #include <iostream>
2 int main()
3 {
4     int cases;
5     std::cin >> cases;
6     for (int i = 0; i < cases; ++i)
7     {
8         long long a, b;
9         std::cin >> a >> b;
10        std::cout << a + b << std::endl;
11    }
12    return 0;
13 }
```

A. 天龍地鐵之謎

Problem ID: subway

艾迪是在佛夏村土生土長的好青年，有著過人的智慧，在求學過程中從來沒有輸給任何人過。唯一的遺憾就是一直沒機會親眼見識佛夏村以外的世界。於是在大學畢業的這個暑假，艾迪和他的好夥伴們一行人決定前往從小到大只在政治版頭條看過的大都市——天龍國——好好享受這個暑假。

而要在天龍國玩遍各個角落，最方便的就是坐地鐵了。天龍國的地鐵非常神祕，總共有 N 個地鐵站，而整個鐵路網又可以分為三層。在第 i 層中，每一個地鐵站 j 都恰有一條直接的路徑可以通往地鐵站 $s_{i,j}$ （這個路徑是單向的而且有可能連向自己）。舉例來說，下圖是一張 $N = 4$ 的地鐵路線圖，其中藍綠紅分別代表第一層、第二層以及第三層地鐵。



此外，天龍國有個都市傳說：「天龍國充滿著無限可能，什麼事都可能發生！」於是在艾迪一行人踏進位於天龍國機場的地鐵站時，眼前突然一片漆黑，緊接著就是一陣昏迷。

醒過來之後，艾迪滿臉困惑，只隱約感受到頭部的疼痛。雖然不知道發生了什麼事，但艾迪發現他身旁的好夥伴們都消失了。不過幸好在天龍國的地鐵站裡都有免費的無線網路熱點，艾迪馬上透過網路聯絡上了他的夥伴們。在一陣混亂地溝通後，艾迪發現他們一行人被分散到天龍國中的各個地鐵站了，但是人生地不熟的，艾迪也無法知道他的其他夥伴們到底是在哪個地鐵站裡。

困惑的艾迪，看著手中的天龍國地鐵路線圖，突然想到了一個好方法：在每一輪，他透過網路通知他的夥伴們到第 k 層地鐵搭乘到下一站（記得在每一站裡的每一層恰好只有一個路徑通往別的地鐵站）。透過這個方法，也許存在一個序列可以保證所有人在最後都一定會聚集在同一個地鐵站裡。以上面的例子而言，艾迪可以讓所有人搭乘第一層地鐵兩次，這樣的話，不管其他夥伴們剛開始的位置在哪，最後都一定會聚集在 4 號地鐵站。

但正當艾迪開始實施他的計畫時，他發現有不明人士在監控他們的網路，而這個人會在艾迪每次送出一個指令後偽造他的身份送出另一個指令。聰明的艾迪發現這個人似乎就是當初拆散他們一行人的壞人，他送出假指令的目的就是要阻止艾迪和他的夥伴們團圓。但艾迪無法告知他的夥伴們哪些指令才是真的，所以在接到假指令時他們也只能照著指令行動。以上面的例子而言，艾迪可以叫他的夥伴們先搭乘第一層地鐵。接下來如果壞人叫他們搭第一層地鐵，那這時候大家一定會聚在同一站；那如果壞人叫他們搭第二層地鐵，艾迪可以接著叫他的夥伴們搭乘第二層地鐵，這樣也能保證大家會聚在同一站；那如果壞人叫他們搭第三層地鐵，艾迪可以接著叫大家搭第二層地鐵，這樣也能讓大家聚在一起。因此在這個情況下，不論壞人如何干擾艾迪，艾迪都有辦法和他的夥伴們團圓。

但實際上天龍國的地鐵圖根本不像上面那個這麼簡單。絕望的艾迪想知道是否存在一個策略能在壞人的干擾下保證他和他的夥伴們能夠再次團圓，聚集到某個地鐵站。但此時頭部的疼痛讓艾迪難以思考，所以他尋求你的幫忙。

Input

測試資料第一行包含一個正整數 N ，代表天龍國地鐵站的個數。

接下來有三行，其中每一行包含 N 個由單一空格隔開的正整數。而其中第 i 行的第 j 個數 $s_{i,j}$ 代表在第 i 層地鐵中，第 j 個地鐵站能夠直接到達第 $s_{i,j}$ 個地鐵站。

- $2 \leq N \leq 1000$
- 對於所有 $1 \leq i \leq 3$ ， $1 \leq j \leq N$ ，都滿足 $1 \leq s_{i,j} \leq N$

Output

請輸出一行，如果存在一個艾迪想要的策略請輸出 “Yes”（不含引號），否則輸出 “No”（不含引號）。

Sample Input 1

```
4
2 4 4 4
1 1 1 2
4 3 2 1
```

Sample Output 1

```
Yes
```

Sample Input 2

```
4
2 3 4 1
4 1 2 3
4 3 2 1
```

Sample Output 2

```
No
```

This page is intentionally left blank.

B. 上司的薪水

Problem ID: boss

諛迪是國際級大財團南投寵物服務公司（Nantou Pet Service Company，簡稱 NPSC）的董事長。最近諛迪在辦公室閒晃的時候，發現每個員工都看起來悶悶不樂的。於是身為親切好老闆的諛迪，就趁午休時間跟那些不快樂的員工聊了一下。發現不快樂最主要的原因是薪水太少，所以諛迪決定要來幫大家加薪，讓大家心情快樂，工作更有效率。

由於 NPSC 是一個龐大的財團，公司內部位階結構嚴謹，每個員工——除了董事長諛迪——都有一位直屬長官，萬事皆向他負責。每個員工剛進公司的時候都會被賦予一個唯一的編號，編號值愈小的代表他在公司的年資愈老。像董事長諛迪的編號就是尊爵不凡的 1 號。而董事長諛迪認為，當人長官的一定要比下屬更有經驗才行，所以每個員工的直屬長官的年資都比該員工老。

而這間公司加薪的方式也相當程度地反映出了 NPSC 的上司所具有的權威性：只要一個員工加薪了 x 元，則他的直屬長官、直屬長官的直屬長官、直屬長官的直屬長官的直屬長官……也都會加薪 x 元。

董事長諛迪的調查還發現，每個人只要總共加薪了 K 元以上（包含 K 元），他就會對自己薪水很滿意，心情變得快樂。

經過一番規劃之後，諛迪開展了他的 NPSC 大加薪計劃：每個禮拜選一位員工（可能是諛迪自己）加薪 x 元，持續 Q 個禮拜不中斷。諛迪想知道，每一次加薪完之後，總共有多少員工對自己的薪水感到滿意。

Input

輸入的第一行有三個正整數 N, Q, K ，分別代表 NPSC 公司的員工人數（包含董事長諛迪）、計劃持續多少個禮拜以及令人滿意的薪資水準。

第二行則包含了 $N - 1$ 個正整數 P_i ，其中第 i 個正整數代表編號 $i + 1$ 的員工他的直屬長官的編號。

接下來包含 Q 行，每行包含兩個正整數 A_i, X_i ，代表第 i 個禮拜將對編號 A_i 的員工及其長官們加薪 X_i 元。

- $3 \leq N \leq 3 \times 10^5$
- $1 \leq Q \leq 3 \times 10^5$
- $1 \leq K \leq 10^{12}$
- $1 \leq P_i \leq i$
- $1 \leq A_i \leq N$
- $1 \leq X_i \leq 10^9$

Output

總共輸出 Q 行，第 i 行包含一個正整數，代表第 i 次加薪後，有多少人對自己當前的薪水感到滿意。

Sample Input 1

```
5 5 5
1 2 3 4
5 1
3 2
2 7
1 2
1 1
```

Sample Output 1

```
0
0
2
2
2
```

Sample Input 2

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

Sample Output 2

```
2
2
3
4
```

This page is intentionally left blank.

C. 魯石戰記

Problem ID: mission

艾迪最近迷上了著名的卡牌遊戲「魯石戰記」。身為不課金的優良玩家，要獲得的遊戲裡金幣的唯一方法就是完成每日任務。而每日任務的規則如下：

- 每天的一開始可以隨機獲得一個新任務。但若是之前累積了三個未完成的任務，就不會得到新任務。
- 完成一項任務可以獲得相對應的獎勵。獎勵分成三種：40 金幣、60 金幣、100 金幣，依照特定的機率分佈隨機出現。
- 每天可以最多替換一個任務。替換任務要銷毀其中一項任務（不會獲得其獎勵），然後會再根據機率分佈隨機獲得一個新任務。
- 完成任務的時間沒有限制，可以在任何時候完成任務，一天也可以完成多個任務，且不需要考慮任務解太久，造成一天解不完的問題。



玩了一個月以後，艾迪發現要解什麼任務對他而言都輕而易舉，但艾迪的夢想是要得到很多很多的金幣。所以艾迪決定要在接下來的 N 天進行一項計畫，打算要用最佳的解任務策略來得到最多的金幣獎勵。在開始這項計畫前，艾迪想先請你幫他算出在使用最佳策略的情況下，在這 N 天的計畫中能獲得的金幣的期望值是多少。

而為了減輕你計算的負擔，艾迪在計畫開始的前一天把之前還沒完成的任務都完成了，所以在計畫開始的第一天，艾迪只會有當天拿到的一個任務。

Input

測試資料第一行包含一個正整數 N ，代表計畫進行的天數。

第二行有三個以單一空格分開的整數 a, b, c ，分別代表 40 金幣、60 金幣和 100 金幣的任務個別出現機率的百分比。

- $1 \leq N \leq 10000$
- $0 \leq a, b, c \leq 100$
- $a + b + c = 100$

Output

請輸出一行，包含一個浮點數，表示艾迪在最佳策略下，在 N 天的計畫中期望獲得的金幣量。

輸出的答案與正確答案誤差於 10^{-6} 內皆視為正確。

Sample Input 1

```
1
20 40 40
```

Sample Output 1

```
83.2
```

Sample Input 2

```
514
0 0 100
```

Sample Output 2

```
51400
```

Sample Input 3

```
7
17 65 18
```

Sample Output 3

```
514.51711373707985330839
```

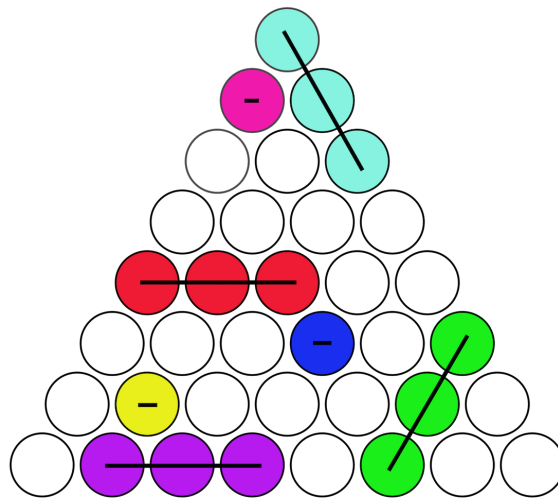
D. 三角尼姆

Problem ID: triangle

有一天鮑伯和艾麗絲在整理家裡的時候，發現了一個寫著「三角尼姆」的紙盒，打開一看發現裡頭有一個三角形的棋盤和一些棋子，像是某種桌遊。但是左看右看都找不到規則說明書，所以鮑伯決定求助谷歌大神。他們在危機百科上查到以下規則：

- 棋盤是一個由 N 層圓圈組成的正三角形。
- 兩方必須輪流在棋盤上擺放棋子，每次可以擺放一枚或三枚棋子，但所擺放的棋子必須相連一直線，且一格中最多只能放一枚棋子。
- 擺放最後一枚棋子的就輸了。

下圖給出了在 $N = 8$ 時的一些可行動作。



這時候鮑伯想知道在他先手的情況下，他有沒有一個必勝的策略。

Input

測試資料只有一行包含一個正整數 N ，代表三角形棋盤的邊長。

- $1 \leq N \leq 100$

Output

請輸出一行，如果鮑伯有一個必勝策略請輸出“Bob”（不含引號），否則輸出“Alice”（不含引號）。

Sample Input 1

1	Alice
---	-------

Sample Output 1

Sample Input 2

2	Alice
---	-------

Sample Output 2

E. 水晶塔傳輸問題

Problem ID: tower

無線通訊隨著科技的進步，已經逐漸成為人們不可或缺的一項技術。然而，由於有空中的訊號衝突以及障礙物屏障等等的因素，要長時間地穩定高速傳輸仍然不是一件容易的事。

為了再把傳輸技術更一步地提升，有一群科學家開始研究最近被發現的一種新的神秘水晶 Ξ 。藉由該種水晶，我們可以架設如同電波塔功能的「水晶塔」。而根據某些複雜的原理，水晶塔可以某種程度地達到穩定的傳輸效能。

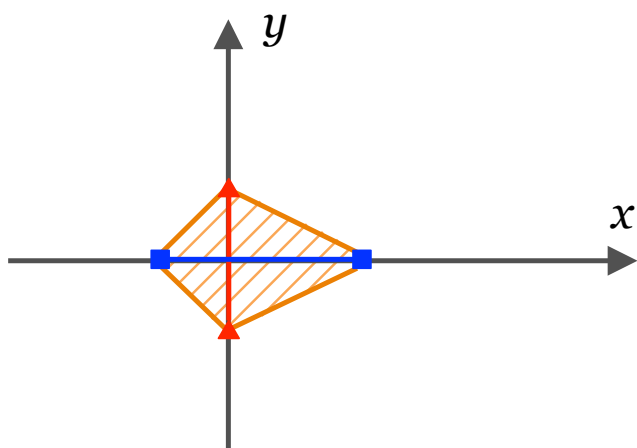
但是，要使用水晶塔做到訊息傳輸的功能必須滿足下列要求：

- 至少要有兩組不同類型（每組至少兩座）的水晶塔。
- 同類型的兩座水晶塔之間會形成能量的連線。若兩種不同類型的水晶塔分別產生的能量連線有所相交 (intersect)，則便可以張開「水晶傳輸空間」。反之，如果兩種能量連線沒有相交，則便無法展開任何傳輸空間。
- 「水晶傳輸空間」的範圍即四座水晶塔所形成的四邊形的面積。在該空間內可以根據某些複雜的原理達到訊息傳輸的目的。

範例請見下圖，其中兩個正方形的點代表第一種類的水晶塔，而兩個三角形的點則表示另一種類的水晶塔。兩種類型各自分別的連線有相交，故可以張開以斜線表示的「水晶傳輸空間」。(此例圖即為第一筆範例輸入的表示圖。)

艾莉莎是一位很好學的水晶工程科學生，她想要研究不同的擺法之間所造成水晶塔張開的總面積的不同。但是，由於一次考慮太多座水晶塔實在是太複雜了，於是她決定先從考慮「四座（且兩個種類各兩座）水晶塔」這個最簡單的情況開始。

為了快速地算出「水晶傳輸空間」的面積，請你寫一支程式，在給定四座水晶塔的座標之情況下，計算其產生的「水晶傳輸空間」的面積。



Input

輸入共四行，分別為要考慮的四座水晶塔之座標。其中前兩行為第一種類的水晶塔的座標；後兩行為另一種類的水晶塔的座標。

每行座標都包含了兩個實數 x, y ，分別代表該水晶塔在 X 方向及 Y 方向於二維平面上距離基準點的偏移單位量。

- 所有的水晶塔座標皆滿足 $-10^4 < x, y < 10^4$ ，數值固定精準到小數點下第三位。
- 四座水晶塔的座標一定兩兩相異。
- 不會有任何一座水晶塔在另兩座水晶塔的連線上。

Output

請先輸出一行恰包含一個非負實數，即輸入中的四座水晶塔所產生的「水晶傳輸空間」之面積(以平方單位的數值表示)。

輸出的答案與正確答案誤差於 10^{-6} 內皆視為正確。

Sample Input 1

```
2.000 0.000
-1.000 0.000
0.000 1.000
0.000 -1.000
```

Sample Output 1

```
3.0000000000
```

Sample Input 2

```
24.152 31.428
13.505 17.779
14.240 32.285
22.130 12.326
```

Sample Output 2

```
160.0970415000
```

Sample Input 3

```
1.500 1.500
1.500 -1.500
-1.500 -1.500
-1.500 1.500
```

Sample Output 3

```
0.0000000000
```

F. 非對稱之美

Problem ID: palindrome

對稱的事物往往令人感到美麗。

回文就是一個對稱的例子。當一個字串反過來寫依然不變的時候我們就說它是一個回文。例如「上海自來水來自海上」和“noon”都是回文；而“XDD”則不是回文。在一個字串中找出最長的回文字子串也是資訊競賽中的經典題型，藉由精巧的 Manacher’s algorithm 可以在線性時間內解決這個問題。

但太多對稱的東西也讓人感到審美疲勞，競賽中只考察經典題也有點無聊。字串界之王蚯蚓決定要另闢蹊徑，探索非對稱之美。因此他想要考考你，如何在字串 s 中，找出最長的「非回文」子字串呢？

蚯蚓很好心的提醒你有關子字串的定義。如果一個字串 t 與 s 中的一個連續區段相等的話，我們就說 t 是 s 的一個子字串。例如 $s = \text{“meow”}$ 時，“meow”跟“eo”都是 s 的子字串；而“mw”則不是。

Input

輸入恰有一行，包含一個字串 s 。

- 字串 s 的長度至少為 1 且不超過 10^6
- 字串 s 僅由小寫英文字母構成

Output

請輸出一個正整數於一行，代表 s 的最長非回文字子串的長度。如果 s 中的所有子字串都是回文的話，請輸出 0。

Sample Input 1

meow

Sample Output 1

4

Sample Input 2

aaaa

Sample Output 2

0

Sample Input 3

noon

Sample Output 3

3

G. 排座位

Problem ID: permu

又到了一年一度的全國榮譽相撲大會 (National Prestigious Sumo Conference, 簡稱 NPSC)。一群家喻戶曉且聲名遠播的頂尖相撲選手將從全國各地前來共襄盛舉。在 NPSC 的主會場中，有一張又大又圓的大圓桌，屆時，所有前來的相撲選手便會在歡迎晚會時圍繞在圓桌邊坐成一圈。而身為今年 NPSC 主辦人的你，最傷腦筋的就是為這些相撲選手安排適當的座位。

經過統計，你得知今年將會有 N 位相撲選手前來。為了避免尷尬，你不希望有空的座位，於是你在圓桌周圍擺放 N 個座位圍成一圈。按照習慣，你將這 N 位相撲選手編號為 $1 \sim N$ 以利大會作業，同樣地，你也將這 N 個座位以順時針順序編號為 $1 \sim N$ 。

有趣的是，在相撲界有個迷信，只要是來參加 NPSC 的相撲選手，如果被編號為第 i 位相撲選手，而又恰好坐在編號第 i 或第 $i+1$ 個位置上，他接下來的一年都將會非常幸運。而且就算編號為第 N 位的相撲選手，如果坐在第 N 或第 1 個位置上，也會有同樣的效果。

不過，你並不知道有哪些相撲選手相信這個迷信，因此，你為每個相撲選手都安排適當的座位使該迷信的條件都能發生。如此一來，只要有相撲選手相信這個迷信，都會覺得自己未來一年會很幸運而感到開心。

然而，在大會當天，你睡過頭了！

現場已經有 K 位相撲選手到場並且入座圓桌，而且並不是所有選手都按照你預期的座位入座。你無法幫已經入座的選手更換座位，因為相撲界最忌諱的就是坐到一半臨時換座位！

著急的你，急著尋找替代方案。只是就算無法更換已入座的選手，你還是希望至少有 M 位相撲選手能坐在與該迷信相符的座位上。因此，你想找出不移動目前已入座的 K 位相撲選手的情況下，有多少安排座位的方式。一個特例是，當 $K=0$ 時，代表所有相撲選手也都還沒到！你其實只是記錯時間提早到了，但是沒有睡過頭。

最後，你好奇有多少安排座位的方法，可以使得至少有 M 位相撲選手能坐在與該迷信相符的座位上。

舉例而言（即第一筆範例測資），所有滿足 $N = 4, M = 2, K = 0$ 的安排座位方法共 14 種如下：（座位由左到右分別為編號 1 ~ 4，實際上為環狀）

1. $\langle 1, 2, 3, 4 \rangle$ （編號 1, 2, 3, 4 滿足）。
2. $\langle 1, 2, 4, 3 \rangle$ （編號 1, 2, 3 滿足）。
3. $\langle 1, 3, 2, 4 \rangle$ （編號 1, 2, 4 滿足）。
4. $\langle 1, 4, 2, 3 \rangle$ （編號 1, 2, 3 滿足）。
5. $\langle 1, 4, 3, 2 \rangle$ （編號 1, 3 滿足）。
6. $\langle 2, 1, 3, 4 \rangle$ （編號 1, 3, 4 滿足）。
7. $\langle 2, 1, 4, 3 \rangle$ （編號 1, 3 滿足）。
8. $\langle 3, 1, 2, 4 \rangle$ （編號 1, 2, 4 滿足）。
9. $\langle 3, 2, 1, 4 \rangle$ （編號 2, 4 滿足）。
10. $\langle 4, 1, 2, 3 \rangle$ （編號 1, 2, 3, 4 滿足）。
11. $\langle 4, 1, 3, 2 \rangle$ （編號 1, 3, 4 滿足）。
12. $\langle 4, 2, 1, 3 \rangle$ （編號 2, 3, 4 滿足）。
13. $\langle 4, 2, 3, 1 \rangle$ （編號 2, 3, 4 滿足）。
14. $\langle 4, 3, 2, 1 \rangle$ （編號 2, 4 滿足）。

Input

輸入的第一行有三個正整數 N, M, K ，分別代表前來參加 NPSC 的相撲選手數、你希望至少滿足多少位相撲選手坐在與該迷信相符的座位上以及目前已有多少位相撲選手入座。

接下來包含 K 行。每行包含兩個正整數 A_i, B_i ，代表編號 A_i 的相撲選手已經入座編號 B_i 的位置。

- $1 \leq N \leq 10^3$
- $0 \leq M \leq N$
- $0 \leq K \leq N$
- $1 \leq A_i \leq N$
- $1 \leq B_i \leq N$
- 保證所有 A_i 皆不相同
- 保證所有 B_i 皆不相同

Output

請輸出一行包含一個正整數，代表有多少種安排座位的方式，可以在不移動已入座的 K 位相撲選手下，使得至少 M ($\geq M$) 位相撲選手坐在與迷信相符的座位上。

由於，可行的方法數可能很多，請輸出該方法數除以 $1000000007(10^9 + 7)$ 的餘數。

Sample Input 1

4 2 0

Sample Output 1

14

Sample Input 24 2 1
1 2**Sample Output 2**

5

Sample Input 34 2 2
1 3
3 4**Sample Output 3**

1

H. 曉涵與方陣

Problem ID: square

數學上有許多種類的特殊方陣（square matrix，即行與列數量相同的矩陣）具有各種神奇的性質。身為一個熱愛數字的學生，曉涵最近在思考怎麼樣的方陣才是真正美妙的方陣。

在眾多種方陣中，曉涵把目標放在一種神奇的方陣（稱為 Ψ 樣式）上，這種特別的方陣有個特點：「若大小為 $N \times N$ ，則恰包含 1 到 N 共 N 種正整數，且每一種正整數恰好出現 N 次」。

對於這種特殊的方陣，曉涵基於她自己對於美觀的認定，認為當某些正整數 i 旁有某些正整數 j 相鄰時較美觀。對此，曉涵把這種美觀的定義數值化成「美妙程度」 $a_{ij} = 0$ 或 1 。若值為 1 代表這個相鄰關係美觀，而若為 0 則否。在此我們定義「相鄰」為該元素於方陣中四方向相接的元素，若某一方向為邊界，則該方向沒有相鄰的元素。

現在問題來了，基於曉涵喜歡的樣式，她很好奇：「給定任一 Ψ 樣式方陣，要如何將方陣內的數排列會使得整個方陣的美妙程度總和**最高**呢？」

一個方陣的美妙程度總和定義為該方陣中所有相鄰數對的「美妙程度」總和。請你寫一支程式解決這個問題。

Input

輸入的第一行恰有一個正整數 N ，代表所要考慮的方陣的列數。接下來有 N 行，其中的第 i 行恰有 N 個整數 a_{ij} ，代表 i 和 j 於方陣中相鄰的「美妙程度」。 $1 \leq i, j \leq N$ 。

- $1 \leq N \leq 4$
- $a_{ij} \in \{0, 1\}$ ，並且保證 $a_{ij} = a_{ji}$ ，對於所有 $1 \leq i, j \leq N$ 。

Output

請先輸出一行恰包含一個整數，即最高能夠達成的美妙程度總和。注意，請不要輸出整個方陣的內容。

Sample Input 1

```
3
0 1 0
1 0 1
0 1 0
```

Sample Output 1

```
9
```

Sample Input 2

```
4
1 1 1 0
1 0 1 0
1 1 0 0
0 0 0 1
```

Sample Output 2

```
20
```

Sample Input 3

```
4
1 0 0 0
0 1 0 1
0 0 0 0
0 1 0 1
```

Sample Output 3

```
14
```