0_送分題 – Hello World

(30分)

前言

比賽開始了！

趕快驗證一下，

網路是否設定正確?

上傳競賽程式是否順利?

程式解答是否用 STDOUT 輸出？

都沒問題，30分就到手了！ 繼續 ... 衝！衝！衝！

問題敘述

請寫一個程式輸出Hello World!

輸入格式

本題無需輸入值

輸出格式

[A~Z][a~z], 空格, 以及常用英文符號 。

資料範圍

[A~Z][a~z], 空格, 以及驚嘆號 “!”

輸入範例1

(無輸入值)

輸出範例1

Hello World!

範例解釋

輸入範例1, 無輸入值，簡單而快樂的輸出Hello World!

0_Hello World

(30 points)

Introduction

YTP Contest has started!

Let’s verify everything first.

Is the internet setting correct?

Is the source code submission working well?

Do you use STDOUT output for program solutions?

Everything is ready! Go get 30 points now!! Go! Go! Go!

Description

Please write a program to output Hello World!

Input Format

This problem requires no input.

Output Format

[A~Z][a~z], space, and common English punctuation.

Constraints

[A~Z][a~z], space, and exclamation mark "!".

Input Example 1

(no input)

Output Example 1

Hello World!

Example Explanation

Input Example 1 has no input, simply output Hello World!

1_走一起的路 (On Our Way)

(5分)

問題敘述

嘉嘉和阿偉曾經是最好的朋友，他們一起練習程式，一起吃拉麵，一起過著資工的生活。曾經，他們因為爭執而反目成仇，誓言從此之後走自己的路。但在小翊的幫忙之下，他們已重修舊好，並打算一起玩桌遊—「第三次領地危機」。

遊戲盤由 $N \times M$ 的二維表格組成，每行由上到下以 1 到 N 的正整數編號，每列由左到右以 1 到 M 的正整數編號，第 x 行第 y 列的格子以 (x, y) 表示。在遊戲開始時，嘉嘉位於 (x_A, y_A)，阿偉位於 (x_C, y_C)；而他們的目標為移動到各自的終點：嘉嘉的終點為 (x_B, y_B)，阿偉的終點為 (x_D, y_D)。

在每次移動時，他們只能往上、下、左、右，其中一個方向移動一格；也就是說，原本位於 (a, b) 的玩家移動一步後可能的位置 (a', b') 只有 $(a+1, b), (a-1, b), (a, b+1), (a, b-1)$，且移動後的位置在遊戲盤中，也就是必須符合 $1 \leq N \leq a', 1 \leq M \leq b'$。

總是會對未來做好規劃的他們打算在遊戲開始前先規劃好各自的移動路徑。其中嘉嘉的路徑為 $J_1, J_2, \ldots, J_k$，阿偉的路徑為 $W_1, W_2, \ldots, W_l$，並且 J_1 和 W_1 是他們在遊戲開始時的位置，而 J_k 和 W_l 是他們各自的終點。

經過了之前的教訓，他們發現了遊戲盤上佈滿了重重的陷阱，走自己的路是十分危險的，因此他們會讓自己路徑上經過的格子數盡量少。除此之外，他們還打算「走一起的路」：他們希望兩人的路徑中重複的格子盡量多；也就是說，他們希望滿足 $(a, b) \in J$ 且 $(a, b) \in W$ 的相異格子盡量多。

因為嘉嘉跟阿偉的演算法期中考爆了，他們希望請更擅長演算法的你寫一支程式幫他們算算，在各自都走最短路徑的情況下，兩人路徑中重複的格子數量最多可以有幾個，好讓他們可以「走一起的路」。

輸入格式

輸入的第一行為一整數 T，表示測資的數量。每筆測資一行，包含十個正整數 N, M, x_A, y_A, x_B, y_B, x_C, y_C, x_D, y_D，分別表示地圖長寬與 A, B, C, D 的位置。

輸出格式

對於每筆測資輸出一行，為嘉嘉和阿偉各自都走最短路徑的情況下，兩人路徑中重複的格子最大可能數量。

資料範圍

- $1 \leq T \leq 10^5$
- $1 \leq N, M \leq 10^5$
- $1 \leq x_A, x_B, x_C, x_D \leq N$
- $1 \leq y_A, y_B, y_C, y_D \leq M$
- (x_A, y_A), (x_B, y_B), (x_C, y_C), (x_D, y_D) 皆相異

輸入範例 1

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

輸出範例 1

```
4
4
```

輸入範例 2

```
1
4 5 3 1 1 3 4 5 1 1
```

輸出範例 2

```
3
```

輸入範例 3

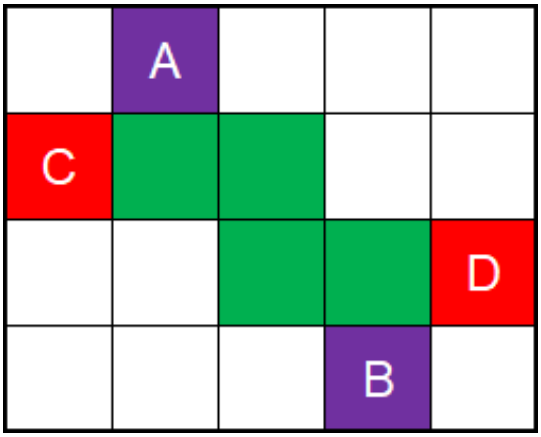
```
1
4 5 1 1 1 2 4 4 4 5
```

輸出範例 3

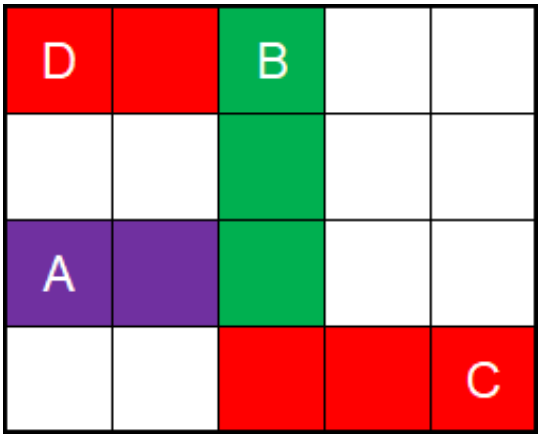
```
0
```

範例說明

下圖為範例測資 1 其中一種兩人可以規劃的路徑，其中紫色為只有嘉嘉經過的格子，紅色為只有阿偉經過的格子，綠色為兩人路徑中重複的格子。



下圖為範例測資 2 其中一種兩人可以規劃的路徑，其中紫色為只有嘉嘉經過的格子，紅色為只有阿偉經過的格子，綠色為兩人路徑中重複的格子。



範例測資 3 兩人皆只需向右一步就會到各自的終點，路徑中不會有重複的格子。

1_On Our Way

(5 points)

Description

Jiajia and Wayne used to be best friends before. They practiced problem solving together, ate ramen together, and enjoyed their lives as a CSIE student. Once, they turned against each other over an argument and vowed to go their own way, but soon patched up their friendship with the support from Yee. Today, they're going to play a table game "The 3rd Territory Crisis" together.

The game board is a table of size $N \times M$. We will consider the table rows numbered from top to bottom from 1 to N, and the columns numbered from left to right from 1 to M. We will denote a cell that is in the x -th row and in the y-th column as (x, y). At the beginning of the game, Jiajia is at the cell (x_A, y_A), Wayne is at the cell (x_C, y_C). The target of the game is to move to their own destinations: Jiajia's is at (x_B, y_B), Wayne's is at (x_D, y_D).

From the cell (a, b), they can move to one of the cells $(a + 1, b)$, $(a - 1, b)$, $(a, b + 1)$, or $(a, b - 1)$, if it exists.

Jiajia and Wayne always plan for their future, and so do they for the game. They're going to decide their own paths together before the game starts. Jiajia's path will be $J_1, J_2, \ldots, J_k$, Wayne's path will be $W_1, W_2, \ldots, W_l$. J_1 and W_1 are the cells where they started at the beginning of the game, J_k and W_l are their destinations.

After their first try of the game, they found that there are many traps on the game board, therefore they decide to make their paths as short as possible this time. Besides, they want to maximize the number of cells which belong to both of their paths. More formally, they want to maximize the number of cells (a, b) that satisfy $(a, b) \in J$ and $(a, b) \in W$.

Jiajia and Wayne both messed up their data structure and algorithm exams, so they would like to ask for your help to calculate the maximal possible number of cells which belong to both of their paths if they make their paths as short as possible.

Input Format

The first line contains a single integer T - the number of test cases. The description of test cases follows. A single line for each test contains ten integers $N, M, x_A, y_A, x_B, y_B, x_C, y_C, x_D, y_D$ - the size of the maps, the cells where they started, and their destinations.

Output Format

For each test case output a single integer - the maximal possible number of cells which belong to both of their paths if they make their paths as short as possible.

Constraints

- $1 \leq T \leq 10^5$

- $1 \leq N, M \leq 10^5$
- $1 \leq x_A, x_B, x_C, x_D \leq N$
- $1 \leq y_A, y_B, y_C, y_D \leq M$
- (x_A, y_A), (x_B, y_B), (x_C, y_C), (x_D, y_D) are pairwise different

Input Example 1

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

Output Example 1

```
4
4
```

Input Example 2

```
1
4 5 3 1 1 3 4 5 1 1
```

Output Example 2

```
3
```

Input Example 3

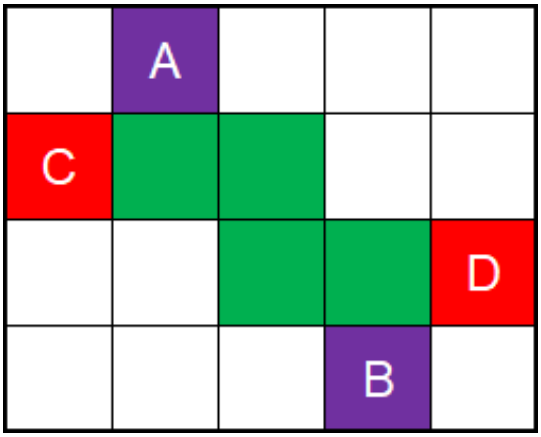
```
1
4 5 1 1 1 2 4 4 4 5
```

Output Example 3

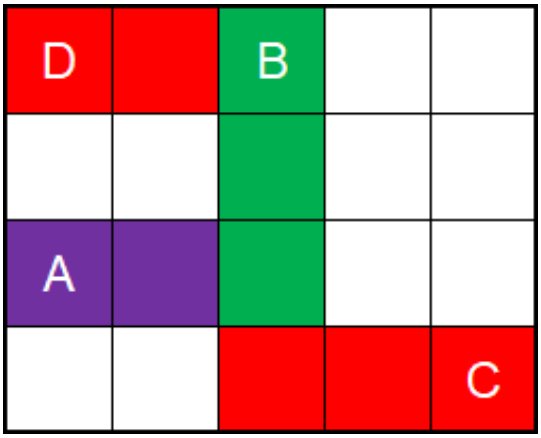
```
0
```

Example Explanation

The figure below shows one of the possible paths in sample test case 1. Jiajia's path consists of purple and green cells, Wayne's path consists of red and green cells.



The figure below shows one of the possible paths in sample test case 2. Jiajia's path consists of purple and green cells, Wayne's path consists of red and green cells.



In sample test case 3, Jiajia and Wayne only need to move one step to their destinations.

2_我鳥光罩 (UR Bubble Shield)

(10分)

問題描述

布丁「一人」下令「R」國侵略「U」國，造成平民死傷並有可能造成核子戰爭與污染。

防患於未然，智者依據時空旅者描繪"西元3000年本地上空有光罩 (bubble shield) 保護"。所以，開始著手研發以臭氧油 (Ozone Oil) 爲基礎的光罩系統 (Halo Protection System) 。

為此，智者要你先建立雙層彎曲結構 (Bi-layer Curved Structure) 分子模型 (Molecular model) 的模擬系統，以分析光罩的化學成分。

因此參考，牛頓拉弗森（Newton Raphson) 法，

- $x_{n+1} = x_n - f(x_n)/f'(x_n)$
  - $f(x)$ 為多項式, 即: $f(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots a_0$
  - $f'(x)$ 為此多項式微分, 即: $f^{'}(x) = n \times a_n x^{n-1} + (n-1) \times a_{n-1}x^{n-2} + \ldots a_1$

假設 $x_0 = 0$, 當 $\frac{f(x_m)}{f'(x_m)} \leq 0.0001,\ (1 < m < 100)$ 時, x_m (小數點後2位四捨五入) 即為 $f(x)$ 的根, 即 $f(x_m) \cong 0$
(不考慮發散、無解、無窮多解, 或是$f'(x_i) = 0$ 的情況)

請你寫一支程式，參考牛頓拉弗森 （Newton Raphson) 法求取多項式等於0的解。

輸入格式

- $\mathbf{a_n\ a_{n-1} \ldots a_0}$ (多項式各項的係數, 以空白隔開)

輸出格式

精確到小數點後2位的浮點數

資料範圍

- $n \leq 10$
- $\mathbf{a_i}, \forall i = 1..n$ 都是整數, $abs(i) \leq 10, abs(a_i) \leq 10$

輸入範例1

```
2 3 -1 -2
```

輸出範例1

```
-1.28
```

輸入範例2

```
1 0 3 -2
```

輸出範例2

```
0.60
```

範例說明

範例3計算如下，

- $x_1 = 0.666667$ and $f(x_1) = 0.296296$
- $x_2 = 0.598291$ and $f(x_2) = 0.009031$
- $x_3 = 0.596074$ and $f(x_3) = 0.000009$
- $f(x_3) < 0.0001$, 所以 0.596... 取小數第 2 位即 0.60.

輸入範例3

```
3 -2 -3 5
```

輸出範例3

```
-1.24
```

2_UR Bubble Shield

(10 points)

Description

Mr. PP launched a full-scale invasion of U Country and made thousands of people dead.

To avoid risk of war, and based on previous time traveler described, there will have a bubble shield in AD. 3000. You are assigned to do some research to invent an Ozone-Oil based Halo Protection System to generate bubble shield in the current complex international environment. As early stage, it is required to build a molecular model simulation system of Bi-Layer Curved Structure to find the chemical composition.

In particular, the program must find the root of a given equation via Newton-Raphson method.

- $x_{n+1} = x_n - f(x_n)/f'(x_n)$
  - $f(x)$ is a polynomials and $f(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots a_0$
  - $f'(x)$ is a differential for $f(x)$, That is, $f^{'}(x) = n \times a_n x^{n-1} + (n-1) \times a_{n-1}x^{n-2} + \ldots a_1$
- Set $x_0 = 0$, loop through till $\frac{f(x_m)}{f^{'}(x_m)} \leq 0.0001,\ (1 < m < 100)$, and then x_m is the root of $f(x)$, that is $f(x_m) \cong 0$
  (assume infinite and no solution, or $f^{'}(x_i) = 0$ situation are excluded).

You are requested to write a program to find the solution of a polynomial equal to zero by referring to Newton Raphson's method.

Input Format

$\mathbf{a_n}\ \mathbf{a_{n-1}} \ldots \mathbf{a_0}$.

Output Format

A float number with precision to 0.01.

Constraints

- $n \leq 10$
- $\mathbf{a_i}, \forall i = 1..n$ is an integer, $abs(i) \leq 10, abs(a_i) \leq 10$

Example 1 Input

```
2 3 -1 -2
```

Example 1 Output

```
-1.28
```

Example 2 Input

```
1 0 3 -2
```

Example 2 Output

```
0.60
```

Example 3 Input

```
3 -2 -3 5
```

Example 3 Output

```
-1.24
```

Example Explaination

Example 2.

- $x_1 = 0.666667$ and $f(x_1) = 0.296296$
- $x_2 = 0.598291$ and $f(x_2) = 0.009031$
- $x_3 = 0.596074$ and $f(x_3) = 0.000009$,
- $f(x_3) < 0.0001$, so round (0.596) to 0.01 as 0.60.

3_三口羊與他的最愛 （Strange Sheep）

(10分)

問題敘述

給你一個長度為 N （$N \leq 10^6$）的正整數序列 A $(a_1, a_2, \cdots, a_N)$，請你找到最長的一段區間 $[l, r]$，使得存在另一個正整數序列 B 在 $[l, r]$ 內的數值 b_i 皆滿足以下條件：

- $1 \leq b_i \leq a_i$
- $b_i < b_{i+1}$
  最後請輸出最長滿足條件區間的長度。

酷愛 treeeeap 的三口羊宣稱她要用 treeeeap 來做這道題目，但聰明的你知道這樣會因為常數太大而 TLE，請問你能否幫助三口羊用比較好一點的方法做出這道題目呢？

輸入格式

第一行有兩個正整數 N，序列的長度為 N。

第二行包含了 N 個正整數 $a_1, a_2, \cdots, a_N$，代表正整數序列 A。

- $0 < N \leq 10^6$
- $0 < a_i \leq 10^9$

輸出格式

請輸出一個正整數代表最長滿足條件區間的長度。

範例輸入 1

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

範例輸出 1

```
3
```

範例輸入 2

```
10
1 5 3 4 3 5 9 6 3 6
```

範例輸出 2

```
6
```

範例輸入 3

```
20
3 1 4 1 5 9 2 6 5 3 5 8 9 7 9 3 2 3 8 4
```

範例輸出 3

```
8
```

範例說明

- 在第 1 筆範例測資中，$[1, 3]$ 這個區間是最長可以滿足題目條件的區間。
- 在第 2 筆範例測資中，$[2, 7]$ 這個區間是最長可以滿足題目條件的區間。
- 在第 3 筆範例測資中，$[8, 15]$ 這個區間是最長可以滿足題目條件的區間。

3_Strange Sheep

(10 points)

Problem Statement

Given an array A with N positive integer $(a_1, a_2, \cdots, a_N)$, you should find the longest subinterval $[l, r]$ such that there exists another array B while every b_i in $[l, r]$ satisfy the condition below:

- $1 \leq b_i \leq a_i$
- $b_i < b_{i+1}$
  Please output the length of such longest interval.

Three-mouthed Sheep, who has fallen head over heels in love with treeeeap, claims that she can solve this problem by treeeeap, but smart as you know that she will fail on this problem because of the running time. Can you help Three-mouthed Sheep to solve this problem with a better algorithm?

Input Format

The first line of the input contains one integer N — the number of elements in the array A.

The second line of the input contains N integers $a_1, a_2, \cdots, a_N$ — elements of the array A.

- $0 < N \leq 10^6$
- $0 < a_i \leq 10^9$

Output Format

Print the length of the longest interval which satisfy the condition above.

Sample Input 1

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

Sample Output 1

```
3
```

Sample Input 2

```
10
1 5 3 4 3 5 9 6 3 6
```

Sample Output 2

```
6
```

Sample Input 3

```
20
3 1 4 1 5 9 2 6 5 3 5 8 9 7 9 3 2 3 8 4
```

Sample Output 3

```
8
```

Example Explanation

- [1,3] is the longest interval in the first sample test case.
- [2,7] is the longest interval in the second sample test case.
- [8,15] is the longest interval in the third sample test case.

4_舞台設計 (Stage Design)

(10 分)

問題敘述

「公元二三四年，諸葛孔明於五丈原之戰病歿。西元二零二二年萬聖夜，孔明轉生於日本東京都澀谷……」

轉生於現代的諸葛孔明，因緣際會下成為了經紀人兼軍師，決心輔佐創作歌手「EIKO」完成她的演唱夢想。千里之行始於足下！要前往武道館也得從小 pub 開始唱起，今晚的 EIKO 將在 CS 夜店進行演唱。為了讓更多人認識 EIKO 美妙的歌聲，孔明決定從舞台設計下手。他請人調查了 N 位聽眾的習性，發現每個人的活動範圍都可以在平面上用一個矩形表示。其中第 i 位聽眾的活動矩形以左下角 $(x1_i,\ y1_i)$ 至右上角 $(x2_i,\ y2_i)$ 標記。

EIKO 今晚的舞台將會是一個圓形。根據孔明的計算，只要該舞台與某位聽眾的活動矩形有至少一點的交集，即便只有剛剛好的相切，那位幸運的聽眾也會被 EIKO 極具感染力的歌聲所吸引，從此成為忠實粉絲。CS 夜店的店長，寇巴亞希提供了 M 種舞台設計方案。每個方案包含三個參數 $x_j,\ y_j,\ r_j$，其中前兩個數字代表第 j 個舞台方案的圓心座標，最後一個數字則代表半徑長度。

由於孔明正忙著調整夜店裡的擺設與動線，早已分身乏術，無法再分心處理舞台設計的問題。因此他請你幫忙完成這項任務，告訴他每一種方案的忠實粉絲數量，以便他做出最佳的決策。

輸入格式

第一行包含兩個正整數 $N,\ M$，分別代表聽眾與舞台方案的數量。
接下來 N 行，每行有四個整數 $x1_i,\ y1_i,\ x2_i,\ y2_i$ 代表第 i 個聽眾的活動矩形。
接下來 M 行，每行有三個整數 $x_j,\ y_j,\ r_j$，分別代表第 j 個舞台方案的圓心座標與半徑長度。

輸出格式

對於每筆方案，輸出一個非負整數代表第 j 種方案的忠實粉絲數量。

資料範圍

- $1 \leq N,\ M \leq 1000$
- $-10^9 \leq x1_i < x2_i \leq 10^9$
- $-10^9 \leq y1_i < y2_i \leq 10^9$
- $-10^9 \leq x_j,\ y_j \leq 10^9$
- $1 \leq r_j \leq 10^9$

輸入範例 1

```
2 3
1 0 3 3
4 3 5 4
3 1 1
3 5 3
6 1 1
```

輸出範例 1

```
1
2
0
```

輸入範例 2

```
3 2
1 1 4 3
3 0 8 2
2 4 3 5
6 5 3
3 1 1
```

輸出範例 2

```
3
2
```

輸入範例 3

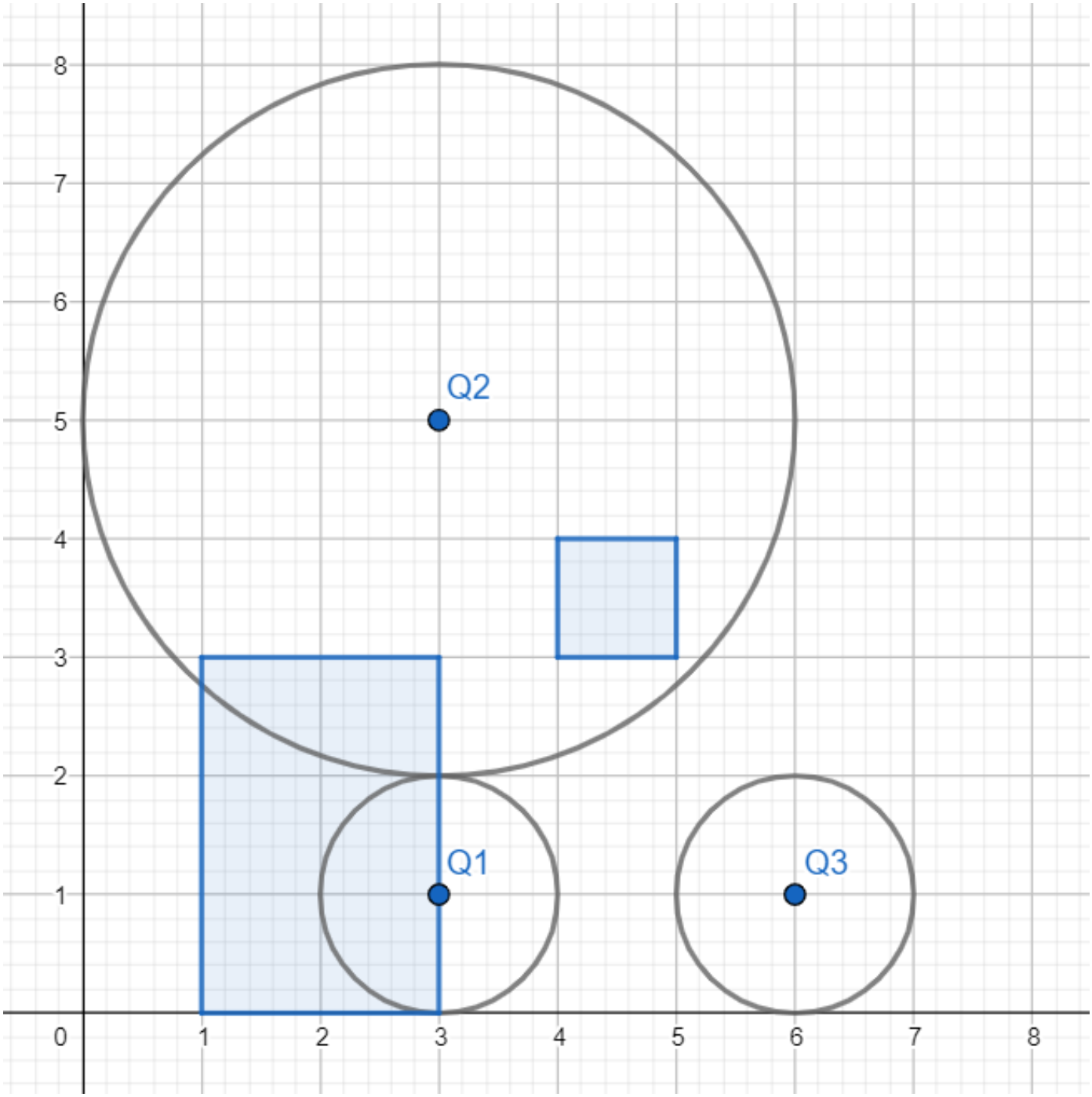
```
3 2
1 1 8 7
5 2 9 5
6 3 7 4
3 5 1
5 2 2
```

輸出範例 3

```
1
3
```

範例說明

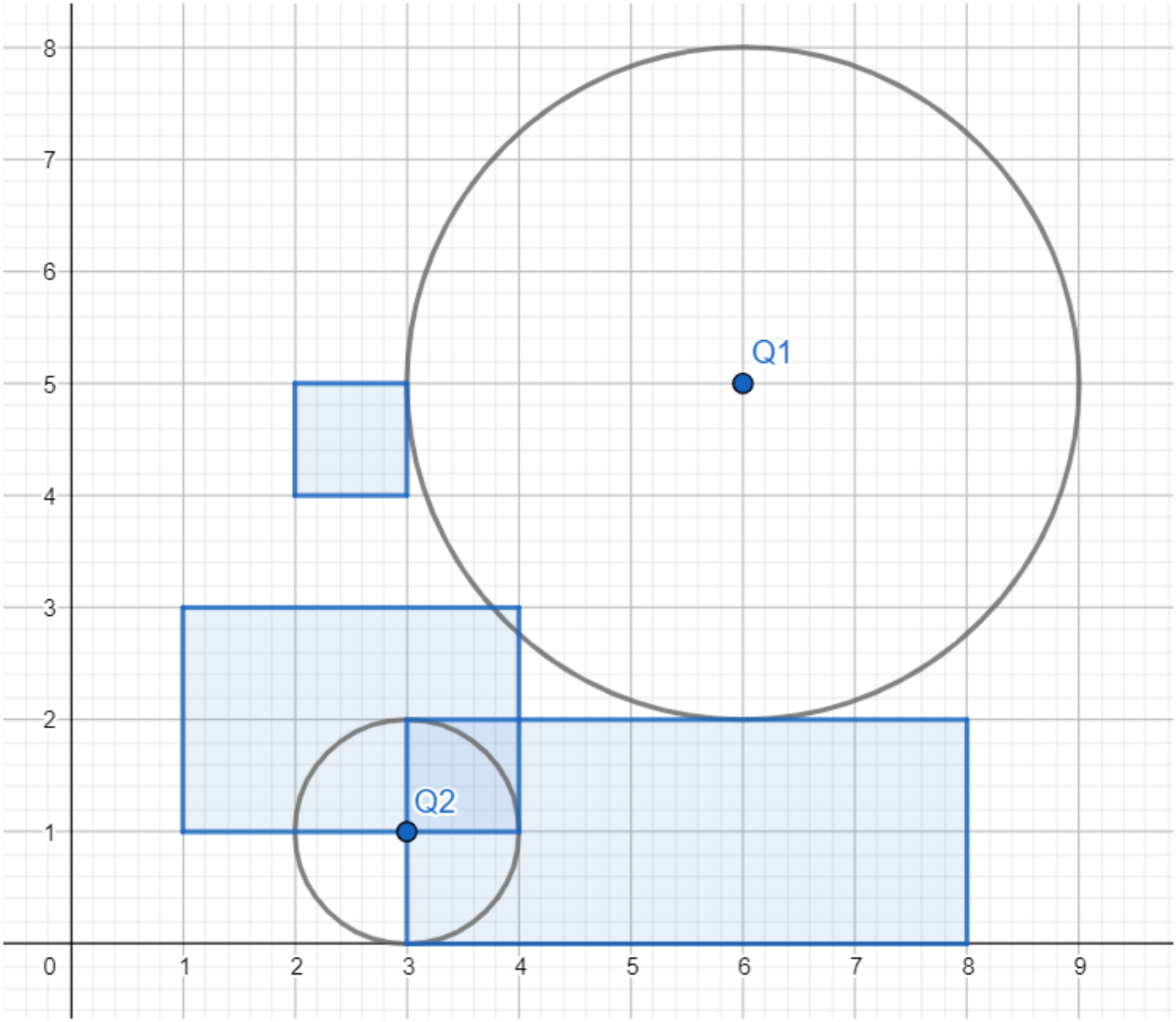
範例測資 1



圖中每個長方形為某個人的活動範圍，第 j 個圓形代表該次方案的舞台。

- 方案一只能吸引到左方的人：共 1 個
- 方案二可以同時吸引所有人：共 2 個
- 方案三完全無法吸引到人：共 0 個

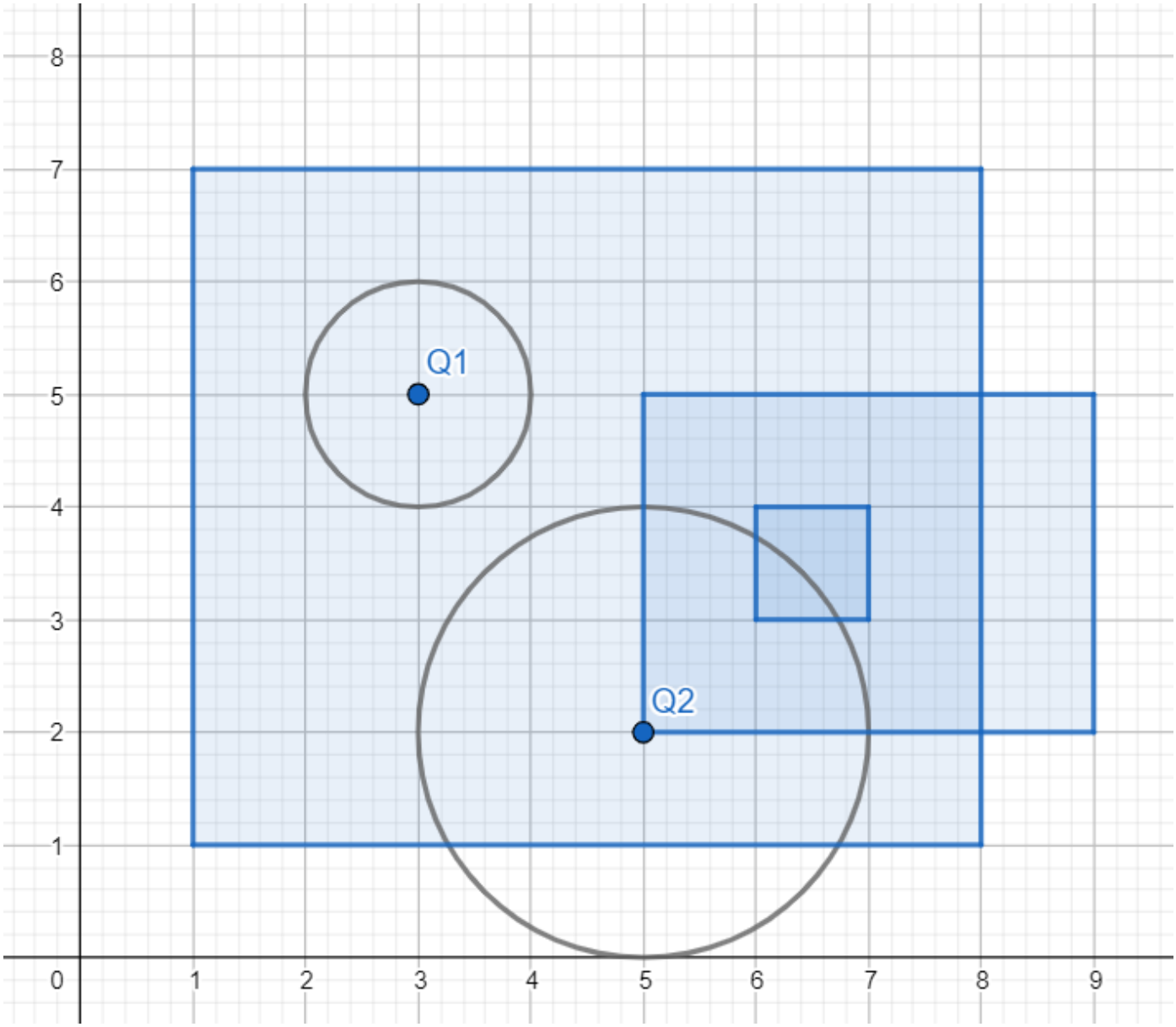
範例測資 2



圖中每個長方形為某個人的活動範圍，第 j 個圓形代表該次方案的舞台。

- 方案一可以同時吸引所有人：共 3 個
- 方案二可以吸引到下方兩個人：共 2 個

範例測資 3



圖中每個長方形為某個人的活動範圍，第 j 個圓形代表該次方案的舞台。

- 方案一只能吸引到完全覆蓋的那個人：共 1 個
- 方案一可以同時吸引所有人：共 3 個

4_Stage Design

(10 points)

Description

"In A.D. 234, Zhuge Kongming died in the Battle of Wuzhang Plains. On Halloween 2022, Kongming reincarnated in Shibuya, Tokyo..."

Reincarnated in the present, Kongming became a manager through serendipity and was determined to assist EIKO, a singer-songwriter, with her concert dream. A journey of a thousand miles begins with a single step! Aiming to step on Budokan should also start from a little pub. Thus EIKO is going to perform in CS pub tonight. In order to introduce the beauty of EIKO's voice to more people, Kongming decided to put his hands on stage design. He
asked someone to investigate N listeners' habits, finding out that everyone's activity range can be described as a rectangle on a plain. The i-th listener's activity rectangle is notated as $(x1_i,\ y1_i)$ being the left-bottom corner, and $(x2_i,\ y2_i)$ being the right-top corner.

The stage for EIKO tonight is a circle. According to Kongming's precise calculation, as long as the stage has any intersection with a listener's activity rectangle, even if there is only one point of tangency, the lucky listener will be charmed by EIKO's infectious singings and become a devoted fan. The shopkeeper of CS pub, Kobayashi, provided M stage designing plans. Each plan consists of three parameters, $x_j,\ y_j,\ r_j$, with the first two numbers representing the center coordinate of the j-th stage plan and the last number representing the radius of the stage.

Since Kongming is busy adjusting the arrangement and traffic flow of the pub, he has no time to deal with the stage designing problem. Therefore he asks your help to finish this task, by telling him how many listeners can be charmed for each plan, and then he can make the best decision.

Input Format

The first line contains two positive integers $N,\ M$ — the number of listeners and stage plans.
The next N lines contain four integers each $x1_i,\ y1_i,\ x2_i,\ y2_i$ — the activity rectangle of the i-th listener.
The next M lines contain three integers each $x_j,\ y_j,\ r_j$ — the center coordinate and radius of the j-th stage plan.

Output Format

For each stage plan, please print one nonnegative integer — the number of listeners whom can be charmed.

Constraints

- $1 \leq N,\ M \leq 1000$
- $-10^9 \leq x1_i < x2_i \leq 10^9$
- $-10^9 \leq y1_i < y2_i \leq 10^9$
- $-10^9 \leq x_j,\ y_j \leq 10^9$

- $1 \leq r_j \leq 10^9$

Input Example 1

```
2 3
1 0 3 3
4 3 5 4
3 1 1
3 5 3
6 1 1
```

Output Example 1

```
1
2
0
```

Input Example 2

```
3 2
1 1 4 3
3 0 8 2
2 4 3 5
6 5 3
3 1 1
```

Output Example 2

```
3
2
```

Input Example 3

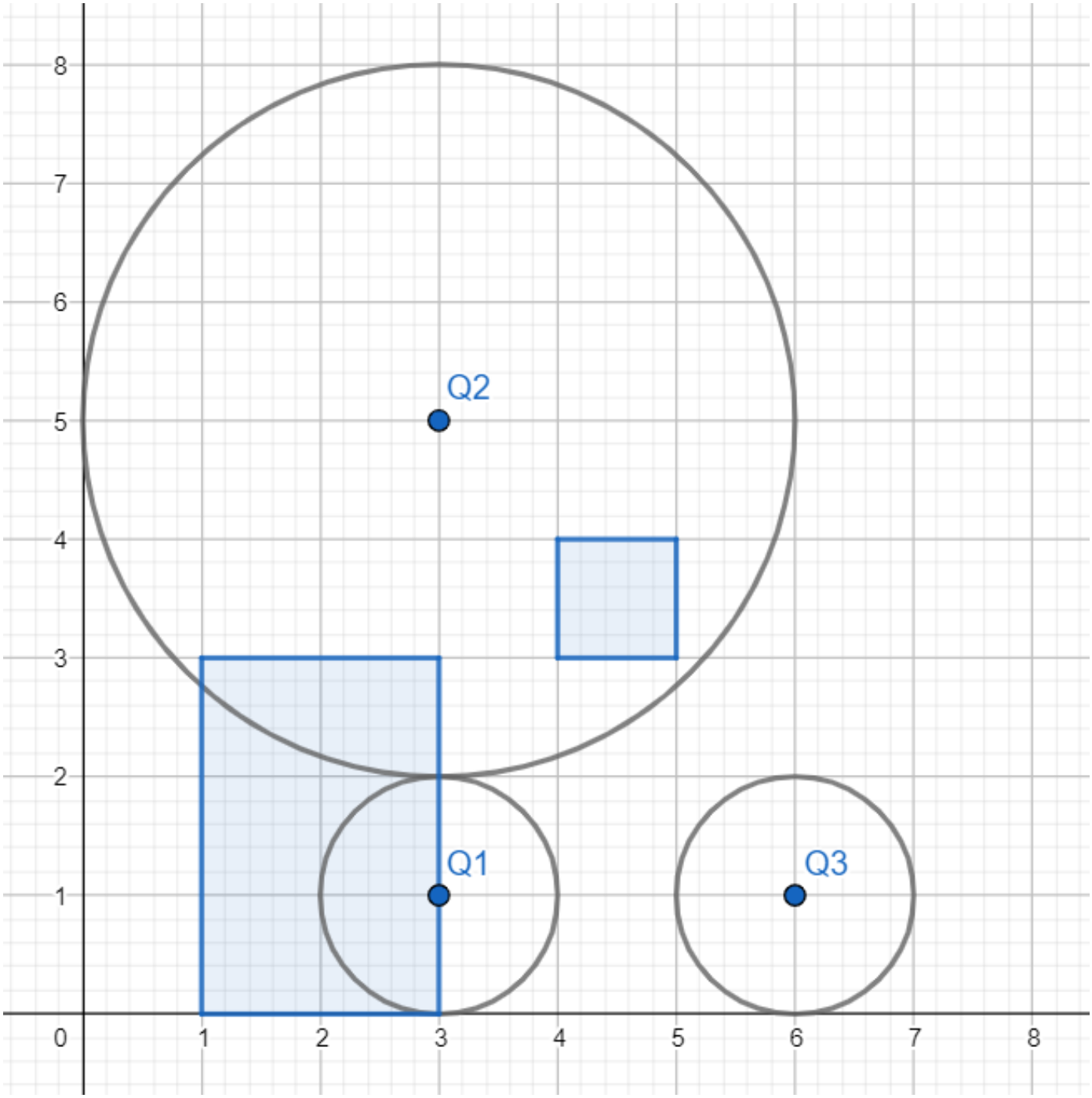
```
3 2
1 1 8 7
5 2 9 5
6 3 7 4
3 5 1
5 2 2
```

Output Example 3

```
1
3
```

Example Explanation

Example 1

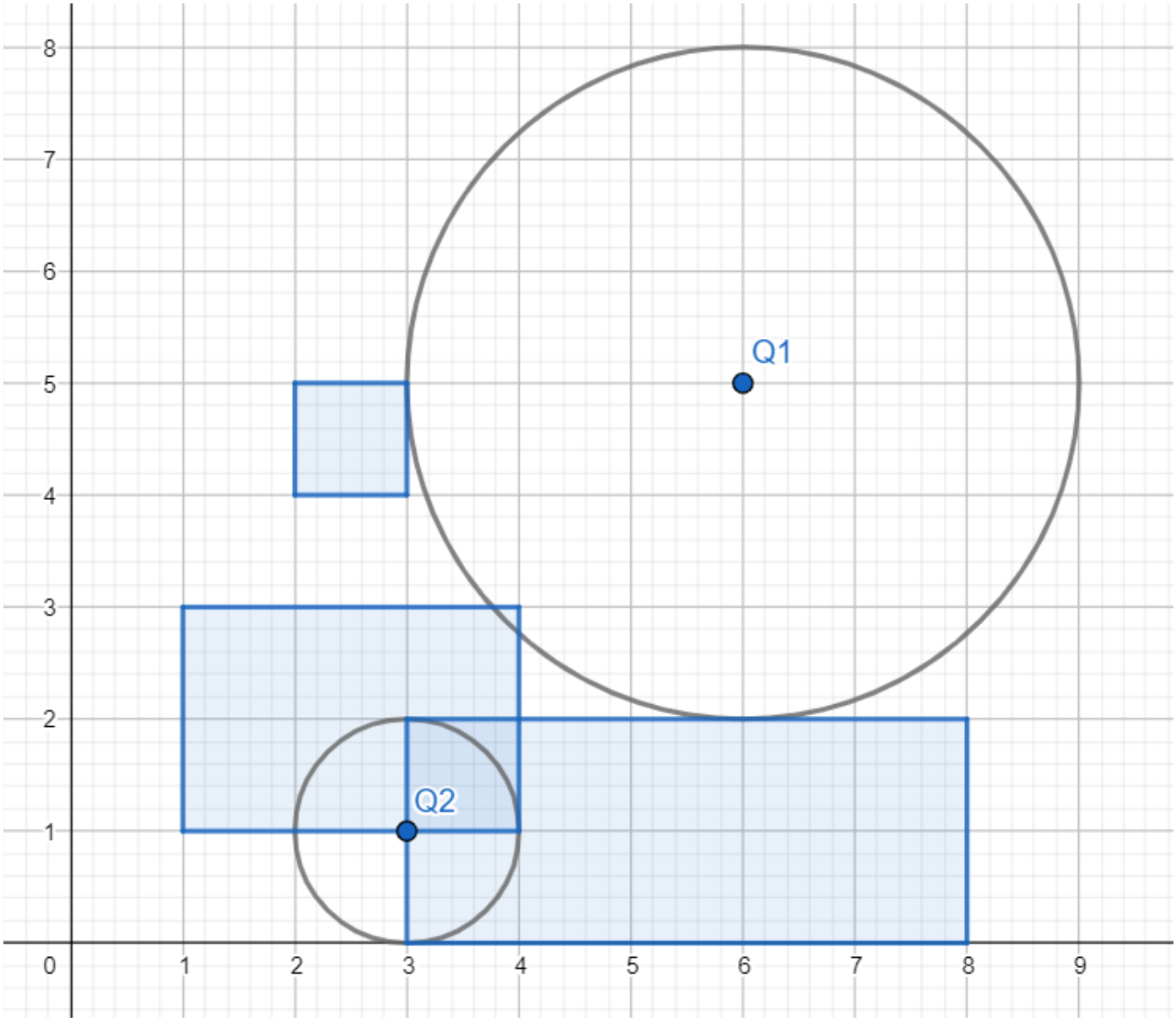


The blue areas in the graph represent a person's activity rectangle, and the j th circle represents the stage of the j th plan.

- Plan 1 can only charm the left person: 1 in total.
- Plan 2 can charm every people: 2 in total.

- Plan 3 can not charm any one: 0 in total.

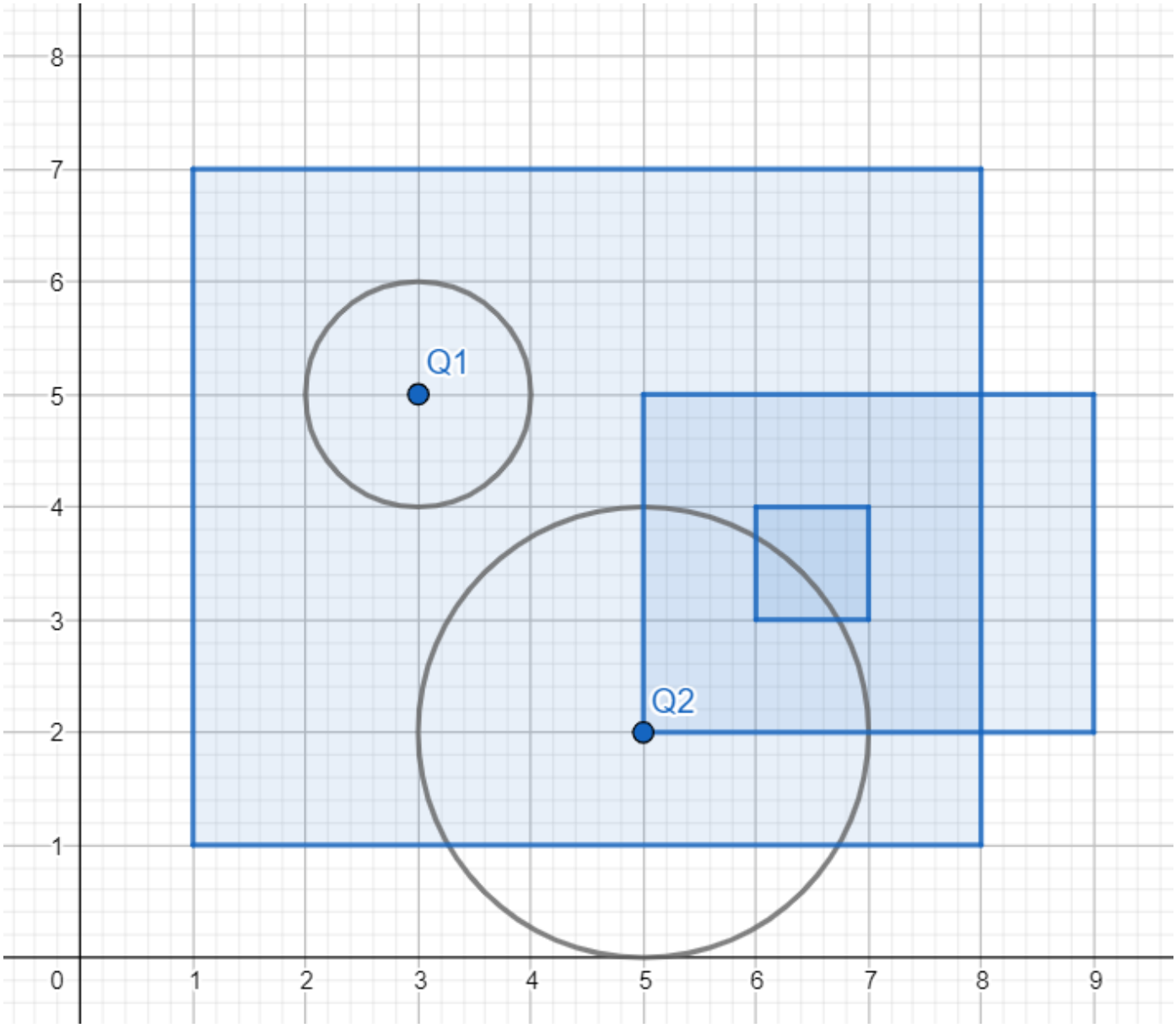
Example 2



The blue areas in the graph represent a person's activity rectangle, and the j th circle represents the stage of the j th plan.

- Plan 1 can charm every people: 3 in total.
- Plan 2 can only charm the two people at the bottom: 2 in total.

Example 3



The blue areas in the graph represent a person's activity rectangle, and the j th circle represents the stage of the j th plan.

- Plan 1 can only charm the person completely covered: 1 in total.
- Plan 2 can charm every people: 3 in total.

5_來個豬的連結 I (Where to Eat Pork I)

(10分)

問題敘述

小郃問小美哪裡可以來吃客豬！

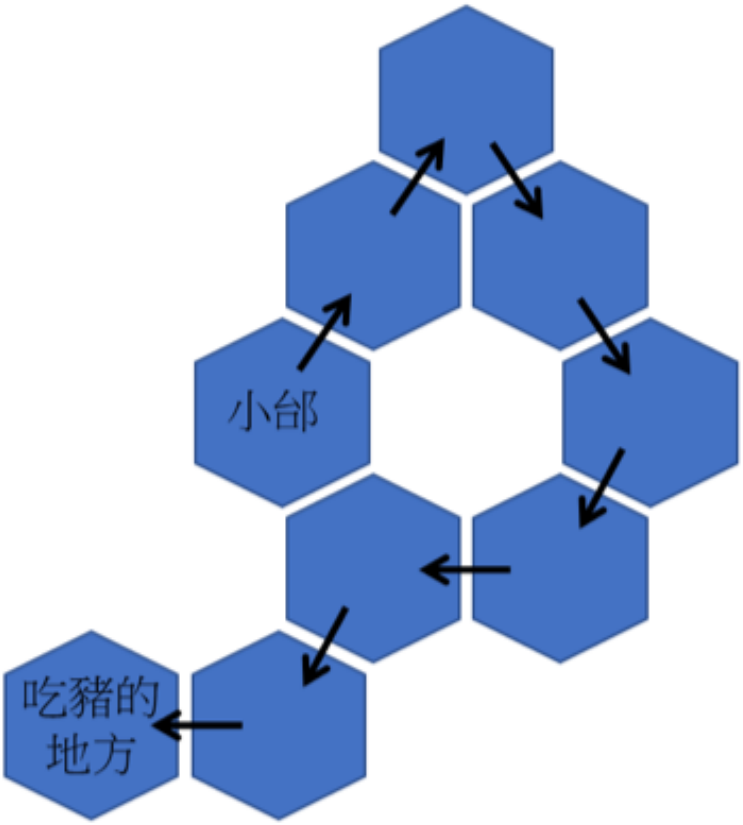
小美利用類似飛行員的方式，**以自身為中心點**，與時鐘的**奇數整點**，來表達兩個點之間的方向：

- 1點鐘方向為A方向,
- 3點鐘方向為B方向,
- 5點鐘方向為C方向,
- 7點鐘方向為D方向,
- 9點鐘方向為E方向,
- 11點鐘方向為F方向。

例如，**以自身為中心點**每次移動**一格**，走完"ABCDEF"，中心點就剛好連線成一個正六角形。

調皮的小美為了作弄小郃, 經常給了過長的路徑, 讓小郃繞了遠路才能到達目的地。

例如，小美告訴小郃的路徑字串，”AACCDEDE”, 如下圖所示, 其實簡化為 “DD”, 就可以了！



所以，要請你協助小郃，寫一支程式，幫小郃找到吃客豬**最短路徑**, 所需**移動的格數**。

輸入格式

由A-F組成長度小於或等於1000 的字串

輸出格式

最短路徑所需**移動的格數**

資料範圍

字元介於[“A”-“F”], 字串長度小於或等於 1000 個字元

Output 數值會小於或等於 1000

輸入範例1

```
AACCDEDE
```

輸出範例1

```
2
```

輸出說明1

如同題目之例舉說明，輸入”AACCDEDE”, 可以簡化為 “DD” (2 格)，故輸出為 2。

輸入範例2

```
AACC
```

輸出範例2

```
2
```

輸出說明2

由給予的輸入”AACC”, 小郤的位置到吃豬的地方, 可以簡化為 “BB” (2 格)，故輸出為 2。

輸入範例3

```
AACCACBAFED
```

輸出範例3

```
3
```

5_Where to Eat Pork I

(10 points)

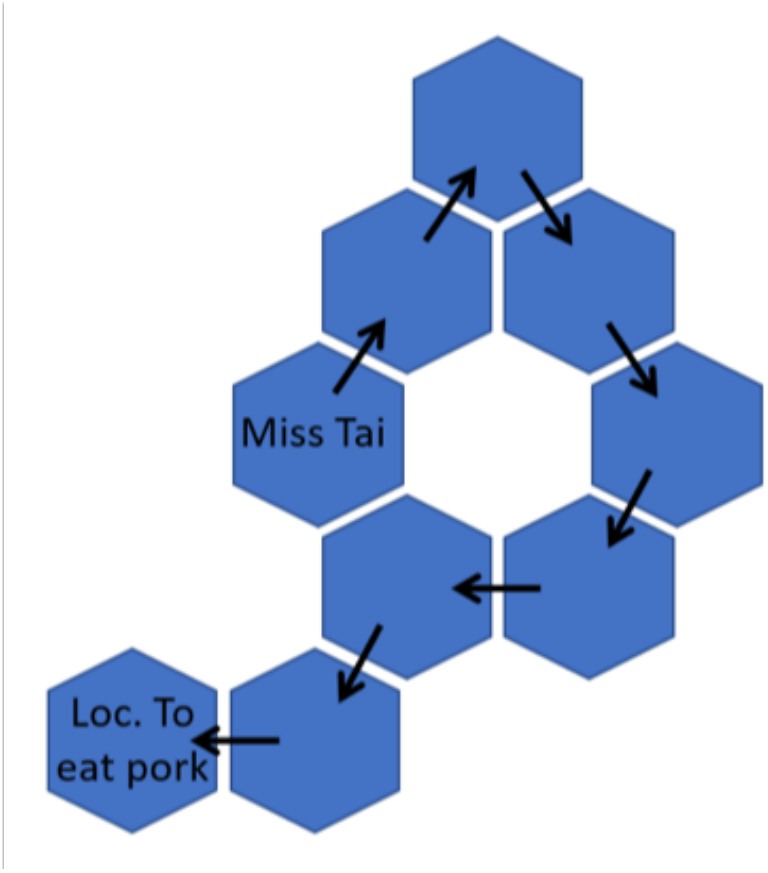
Description

Miss. Tai asked Miss Mei how to get to the place to eat pork. Miss Mei gives the instruction similar as flight pilot with the odd numbers in a clock. Such as,

- center toward 1 o'clock named as "A" direction,
- center toward 3 o'clock as "B" direction,
- center toward 5 o'clock as "C" direction,
- center toward 7 o'clock as "D" direction,
- center toward 9 o'clock as "E" direction,
- center toward 11 o'clock as "F" direction.

Miss Mei gives messy and long instructions intentionally to waste Miss Tai's time finding the place.

For example, from "AACCDEDE" instruction, Miss Tai can go to final location by shortened as "DD" – 2 steps.



So, Miss Tai requests your help find a shortest path to eat pork. Please write a program for him!

Input Format

a string; composed of "A" to "F", representing the original path in between Miss Tai and the location to eat pork.

Output Format

an integer indicates the length of shortest path

Constraints

The characters in the input string is in between “A” to “F”. The length of the input string is less or equal to 1000 characters. The output number is less or equal to 1000.

Example 1 Input

```
AACCDEDE
```

Example 1 Output

```
2
```

Description:

As in the example of the question, input "AACCDEDE", can be simplified to "DD" (2 steps), so the output is 2.

Example 2 Input

```
AACC
```

Example 2 Output

```
2
```

Description:

From “AACC” instruction, the instruction to final location can be shortened as “BB” – 2 steps.

Example 3 Input

```
AACCACBAFED
```

Example 3 Output

```
3
```

6_來個豬的連結 II (Where to Eat Pork II)

(15分)

問題敘述

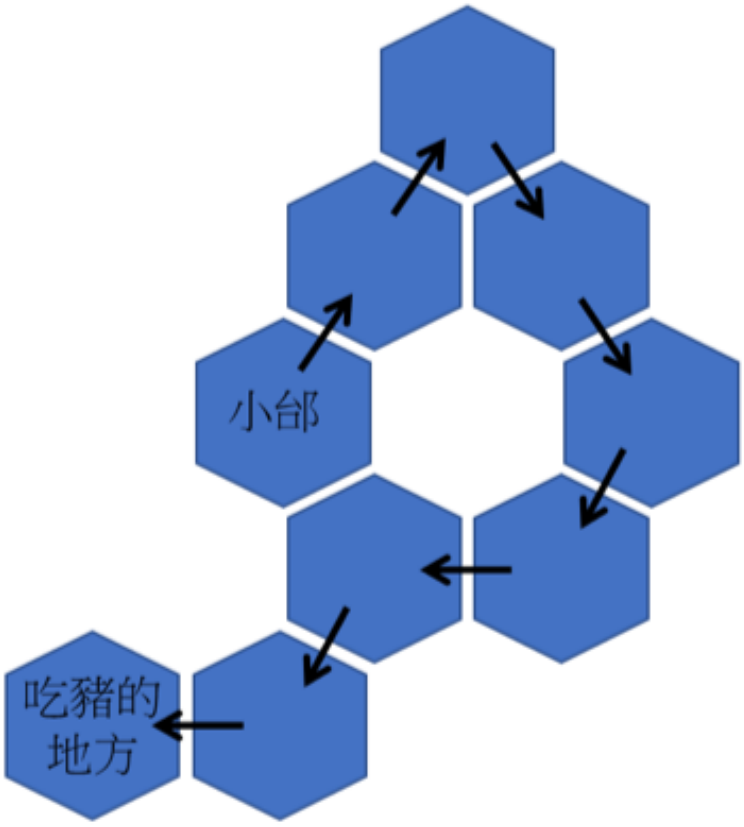
小美利用類似飛行員的方式，**以自身為中心點**，與時鐘的**奇數整點**，來表達兩個點之間的方向：

- 1點鐘方向為A方向,
- 3點鐘方向為B方向,
- 5點鐘方向為C方向,
- 7點鐘方向為D方向,
- 9點鐘方向為E方向,
- 11點鐘方向為F方向。

例如，**以自身為中心點**每次移動**一格**，走完"ABCDEF"，中心點就剛好連線成一個正六角形。

調皮的小美為了作弄小郃, 經常給了過長的路徑, 讓小郃繞遠路。但是，至少小美指示的路都是安全的。所以，為了安全，**小郃只能經過小美提到路徑所經過的點！！**

如此，當小美告訴小郃的路徑是”AACCDEDE”時，如下圖所示，小郃還是可以簡化為“CDE” 移動3格！



所以，要請你協助小郃，寫一支程式，幫小郃找到吃客豬**最近的路徑**, 需要**移動的格數**。

輸入格式

由A-F組成長度小於或等於1000 的字串

輸出格式

最短路徑所需**移動的格數**

資料範圍

- 字元介於A 到 F, 字串長度小於或等於 1000 個字元
- Output 數值會小於或等於 1000

輸入範例1

```
AACCDEDE
```

輸出範例1

```
3
```

輸出說明1

如同題目之例舉說明，輸入”AACCDEDE”, **小郤只能經過小美提到路徑所經過的點！！**，可以簡化為 “CDE” (3 格)，故輸出為 3。

輸入範例2

```
AACC
```

輸出範例2

```
3
```

輸出說明2

依據 “AACC” 的路徑, 最終的位置如圖可以簡化為 “ABC” 或 “CBA” 3 格。

輸入範例3

```
AACCACBAFED
```

輸出範例3

```
3
```

6_Where to Eat Pork II

(15 points)

Description

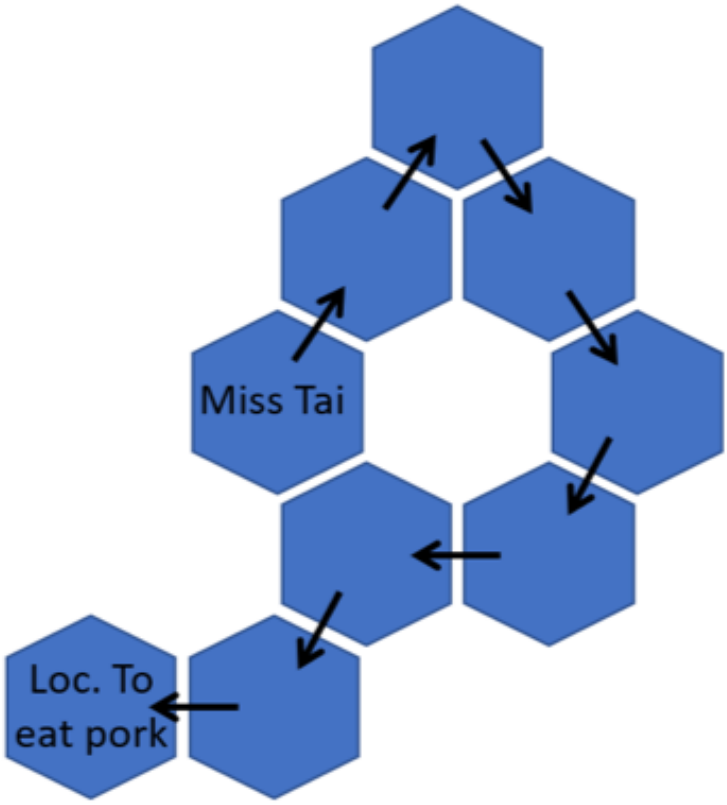
Miss. Tai asked Miss Mei how to get to the place to eat pork. Miss Mei gives the instruction similar as flight pilot with the odd numbers in a clock. Such as,

- center toward 1 o'clock as "A" direction,
- center toward 3 o'clock as "B" direction,
- center toward 5 o'clock as "C" direction,
- center toward 7 o'clock as "D" direction,
- center toward 9 o'clock as "E" direction,
- center toward 11 o'clock as "F" direction.

Miss Mei gives messy and long instructions intentionally to waste Miss Tai's time finding the place.

Somehow, Miss Tai can only go thru exist node; which Mei ever mentioned in the instruction.

For example, from "AACCDEDE" instruction, Miss Tai can go to final location by shortened as "CDE" – 3 steps.



So, Mr. Tai requests your help find a shortest path to eat pork. (The solution can only go thru exist node; which has ever mentioned in the instruction)

Input Format

a string; composed of “A” to “F”, representing the original path in between Mr. Tai and the location to eat pork

Output Format

an integer indicates the length of shortest path

Constraints

The characters in the input string is in between “A” to “F”

The length of the input string is less or equal to 1000 characters.

The output number is less or equal to 1000.

Example 1 Input

```
AACCDEDE
```

Example 1 Output

```
3
```

Description:

As in the example of the question, input "AACCDEDE", can be simplified to “CDE” – 3 steps (The solution can only go thru exist node; which has ever mentioned in the instruction).

Example 2 Input

```
AACC
```

Example 2 Output

```
3
```

Description:

From “AACC” instruction, the instruction to final location can be shortened as “ABC” or “CBA”. Both are 3 steps.

Example 3 Input

```
AACCACBAFED
```

Example 3 Output

```
3
```

7_翻硬幣問題 (Flipping Coins)

(15分)

問題敘述

小 Y 與小 P 在玩遊戲，小 Y 在桌上擺了一排硬幣，從左至右一共有 N 個，每個硬幣都有分正面與反面。為了考驗小 Y，小 P 設下了 Q 個問題要考考小 P，每個問題均為以下兩種形式之一：

- $1\ l\ r$: 代表小 P 要求小 Y 將第 l 個硬幣至第 r 個硬幣翻面。
- $2\ l\ r$: 代表小 P 希望小 Y 回答在第 l 個硬幣至第 r 個硬幣中，有多少對硬幣是不同面朝上的。

但是小 Y 很懶惰，所以他決定請你幫他回答這些問題。

輸入

輸入第一行有兩個正整數 N, Q ($1 \leq N, Q \leq 10^5$)，代表桌子上的硬幣個數以及詢問個數。
輸入第二行一個長度為 N 的 01 字串 s ($s_i \in \{0, 1\}$)，分別代表每個硬幣初始狀態，0 表示正面朝上，1 表示反面朝上。
接下來有 Q 行，每一行皆為 $1\ l\ r$ 或 $2\ l\ r$ ($1 \leq l \leq r \leq N$) 的形式，依序代表小 Y 給小 P 的考驗。

輸出

對於每一筆如 $2\ l\ r$ 的詢問，請輸出一個整數代表答案，不同詢問請以換行符號間隔。

輸入範例1

```
5 5
01001
2 3 5
1 3 5
1 2 3
2 4 5
2 1 3
```

輸出範例1

```
2
1
0
```

輸入範例2

```
5 5
01011
1 3 5
2 2 2
1 3 4
1 1 5
2 3 5
```

輸出範例2

```
0
2
```

輸入範例3

```
10 5
0001000110
1 1 10
1 5 5
1 3 10
2 1 7
2 1 9
```

輸出範例3

```
12
18
```

範例說明

在輸入範例1中，對於第一筆詢問，有兩對硬幣 $(3,5),(4,5)$ 正反面不同，經過兩次操作後，硬幣狀態變成 00010。因此對於第四筆詢問，有一對硬幣 $(4,5)$ 正反面不同。
在輸入範例2中，在所有操作完成後，硬幣的狀態為 10101。
在輸入範例3中，在所有操作完成後，硬幣的狀態為 1101100110。

7_Flipping Coins

(15 points)

Problem Description

Alice and Bob are playing a game. Alice arranges a series of coins on the table, with a total amount of N coins. Each coin has a head and tail. To give Alice a trial, Bob gives Q questions to Alice, which Alice should answer in order. Each question has one of the following forms:

- `1 l r`: Bob asks Alice to flip the coins from l^{th} coin to r^{th} coin.
- `2 l r`: Bob asks Alice to answer how many pairs of coins among l^{th} coin to r^{th} coin such that they have the different sides facing upwards.

However, Alice is indolent. Please help Alice to finish these questions.

Input

The first line of input contains two positive integers N, Q ($1 \leq N, Q \leq 10^5$), representing the number of coins and the number of question, respectively.

The second line contains a string s with length N ($s_i \in \{0, 1\}$), describing the initial state of each coin. s_i is 0 if it has heads facing upwards, and vice versa.

In the following Q lines, each line is the form of either $1\ l\ r$ or $2\ l\ r$, describing the questions in order.

Output

For each question of the form $2\ l\ r$, please output an integer in a line, representing the answer.

Sample Input 1

```
5 5
01001
2 3 5
1 3 5
1 2 3
2 4 5
2 1 3
```

Sample Output 1

```
2
1
0
```

Sample Input 2

```
5 5
01011
1 3 5
2 2 2
1 3 4
1 1 5
2 3 5
```

Sample Output 2

```
0
2
```

Sample Input 3

```
10 5
0001000110
1 1 10
1 5 5
1 3 10
2 1 7
2 1 9
```

Sample Output 3

```
12
18
```

Hints

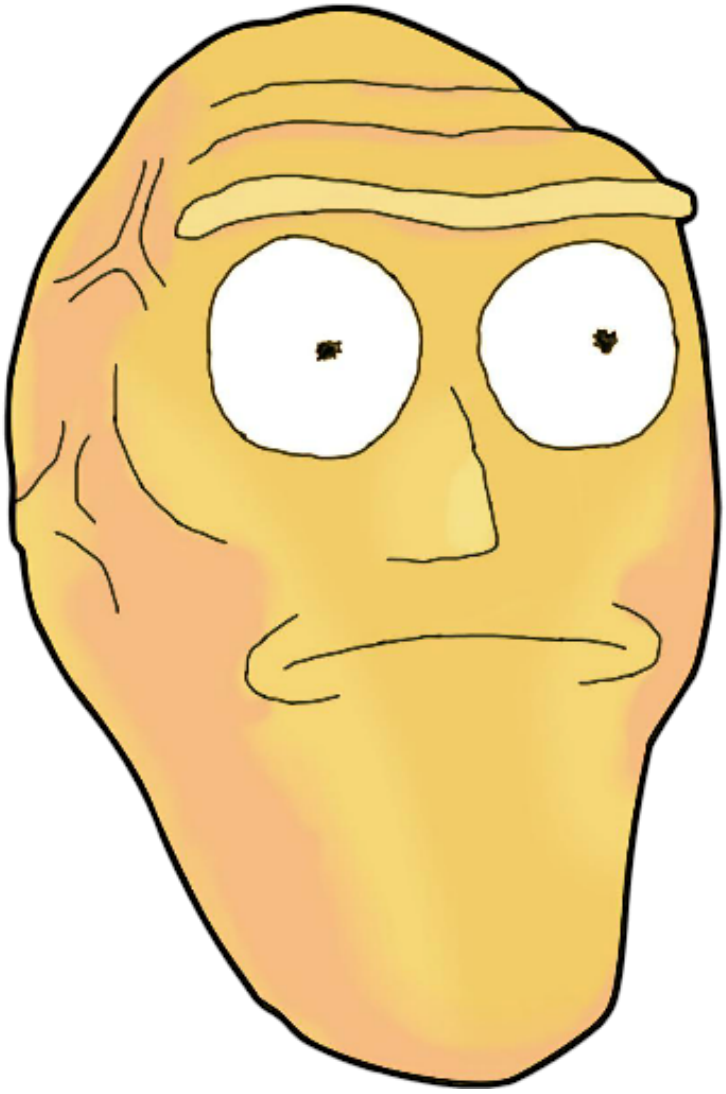
In Sample Input 1. For the first question, there are two pairs of coins $(3, 5), (4, 5)$ which has different sides facing upwards. After two operations $1\ 3\ 5$ and $1\ 2\ 3$, the state of coins become 00010. Therefore, for the fourth question, only one pair of coins $(4, 5)$ has different sides facing upwards.
In Sample Input 2, the final sequence of coins is 10101.
In Sample Input 3, the final sequence of coins is 1101100110.

8_讓我瞧瞧你的能耐 （Show Me What You Got）

(5分 / 10分)

問題敘述

喀姆隆星人是種與行星大小相近，樣貌像巨大人頭的生物，生活在Cygnus-5空間。
據某位名為瑞克的科學家所言，喀姆隆星人以低等生物的才藝表演為食。
而具體而言，喀姆隆星人會將存有智慧物種的星球傳送到他們所在的空間，令他們上演一場名為*行星音樂*的實境秀。
在這實境秀中，各個行星將要使出渾身解數以在音樂競賽中脫穎而出。
因為落敗或被取消資格的行星將被巨大的等離子射線炮解體。



現在，有N顆行星已經被傳送到喀姆隆星人所在的空間，這N個行星由左到右排成一排。
其中，每顆行星都有一個分數代表他們音樂的價值，第i顆行星的價值以a_i表示。
為了選出*行星音樂*的最終贏家，在每一集實境秀結束時喀姆隆星人將會使用等離子射線摧毀當前分數最低的行星(如果有多個行星分數最低，喀姆隆星人會摧毀最左邊的行星)

在每一集的開始，若現在還存活的行星是k個連續的區間，而第i個區間為$[l_i, r_i]$，則這集的評價將會等於以下的式子：

$$\sum_{i=1}^{k}(r_i - l_i + 1) \min_{l_i \le j \le r_i} a_j$$

注意到如果只剩下一個行星了，還會有一集頒獎典禮，頒獎典禮的評價會等於這顆行星的音樂價值。

身為*行星音樂*的導演，你想知道評價最高的一集評價會是多少。

輸入格式

第一行包含一個整數N。
第二行有N個整數，第i個代表a_i

輸出格式

輸出最高評價的一集的評價佔一行。

資料範圍

- $1 \le N \le 200000$
- $1 \le a_i \le 10^9$

子任務

在價值5分的子測資中，保證$1 \le N \le 3000$

輸入範例 1

```
3
1 2 3
```

輸出範例 1

```
4
```

輸入範例 2

```
3
1 100 1
```

輸出範例 2

```
100
```

輸入範例 3

```
10
100 2 6 2 10 56 50 40 36 90
```

輸出範例 3

```
310
```

範例說明

Input Example 1:
第二集的評價是4，[2, 3]這個唯一的區間的最小值是2，長度是2，2*2 = 4。

Input Example 2:
第三集，也就是頒獎典禮，評價是100。
[2, 2]這個區間的最小值是100，長度是1，100*1 = 100

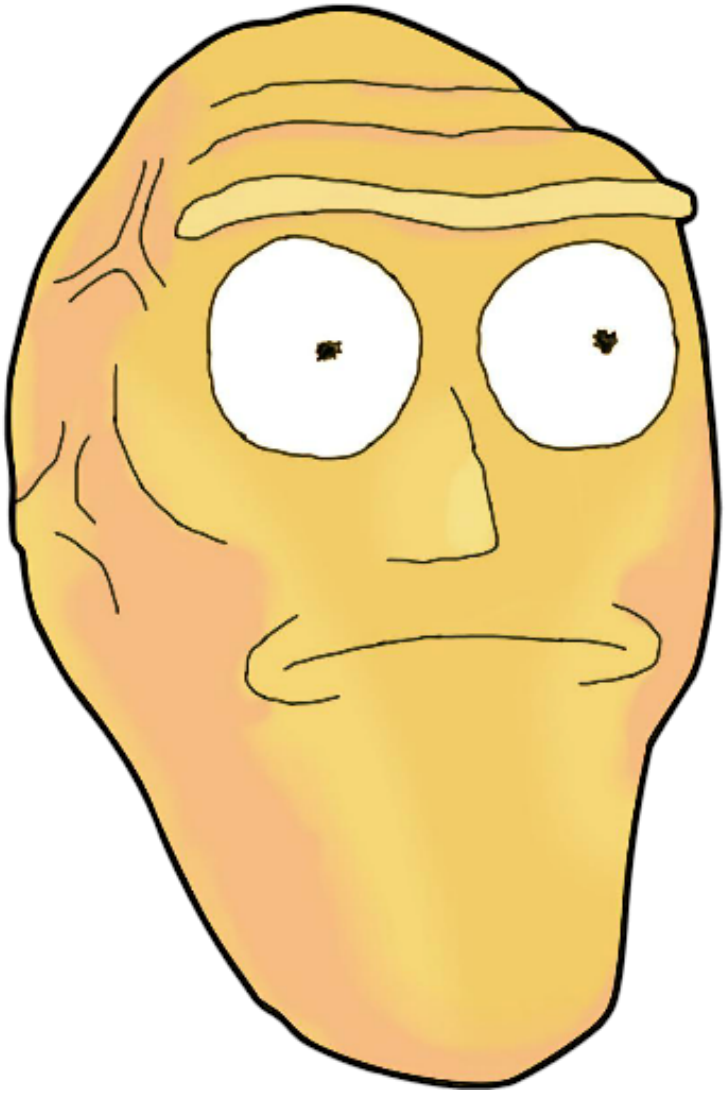
Input Example 3:
第六集, 100 + 40*3 + 90 = 310
在第六集共有3個區間，分別為[1, 1], [6, 8], [10, 10]，每個區間中的最小值分別為，100, 40, 90，乘上該區間的長度後相加為310。

8_Show Me What You Got

(5 points / 10 points)

Description

"Cromulons" are a species of planet-sized beings shaped as giant human heads, and are native to the Cygnus-5 expanse. According to a scientist named Rick Sanchez, the Cromulons "feed on the talent and showmanship of less-evolved lifeforms". They do this by capturing planets with intelligent lifeforms on them, teleporting these planets to the expanse of the universe they inhabit, and run a reality television named *Planet Music*, in which these planets compete to perform music which satisfies them. Planets that lose or are disqualified are disintegrated by a massive plasma ray cannon.



Now, there're N planets which have been teleported to the expanse the Cromulons inhabit. These N planets were arranged in a row from left to right.
Moreover, each planet has a score a_i representing the value of its music.
To get the winner of *Planet Music*, Cromulons use their massive plasma ray cannon to destroy the planet with the minimum value at the end of each episode until there's only one planet left(if there're multiple planets with the minimum value, they'd destroy the leftmost one)

At the beginning of each episode, suppose there are k continuous segments of planets which is still alive, and the ith segment is $[l_i, r_i]$, the rating of this episode is given by the formula below:

$$\sum_{i=1}^{k}(r_i - l_i + 1) \min_{l_i \le j \le r_i} a_j$$

Notice that if there's only one planet left, there'd be an award ceremony episode, whose rating is the score of the planet's value.

As the director of *Planet Music*, you want to know what's the highest rating among all episodes.

Input Format

First line contains an integer N.
Second line contains N integers, the ith one represents a_i

Output Format

Print the highest rating among all episodes in one line.

Constraints

- $1 \le N \le 200000$
- $1 \le a_i \le 10^9$

Subtasks

For a subtask worth 5 points, $1 \le N \le 3000$

Input Example 1

```
3
1 2 3
```

Output Example 1

```
4
```

Input Example 2

```
3
1 100 1
```

Output Example 2

```
100
```

Input Example 3

```
10
100 2 6 2 10 56 50 40 36 90
```

Output Example 3

```
310
```

Example Explanation

Input Example 1:
episode 2 has the highest rating 4.
In episode 2, [2, 3] is the only segment, and the minimum number in [2, 3] is 2, the length of [2, 3] is 2, so the rating is 2*2 = 4.

Input Example 2:
episode 3, which is the award ceremony has the highest rating 100.
In episode 3, [2, 2] is the only segment, and the minimum number in [2, 2] is 100, the length of [2, 2] is 1, so the rating is 100*1 = 100.

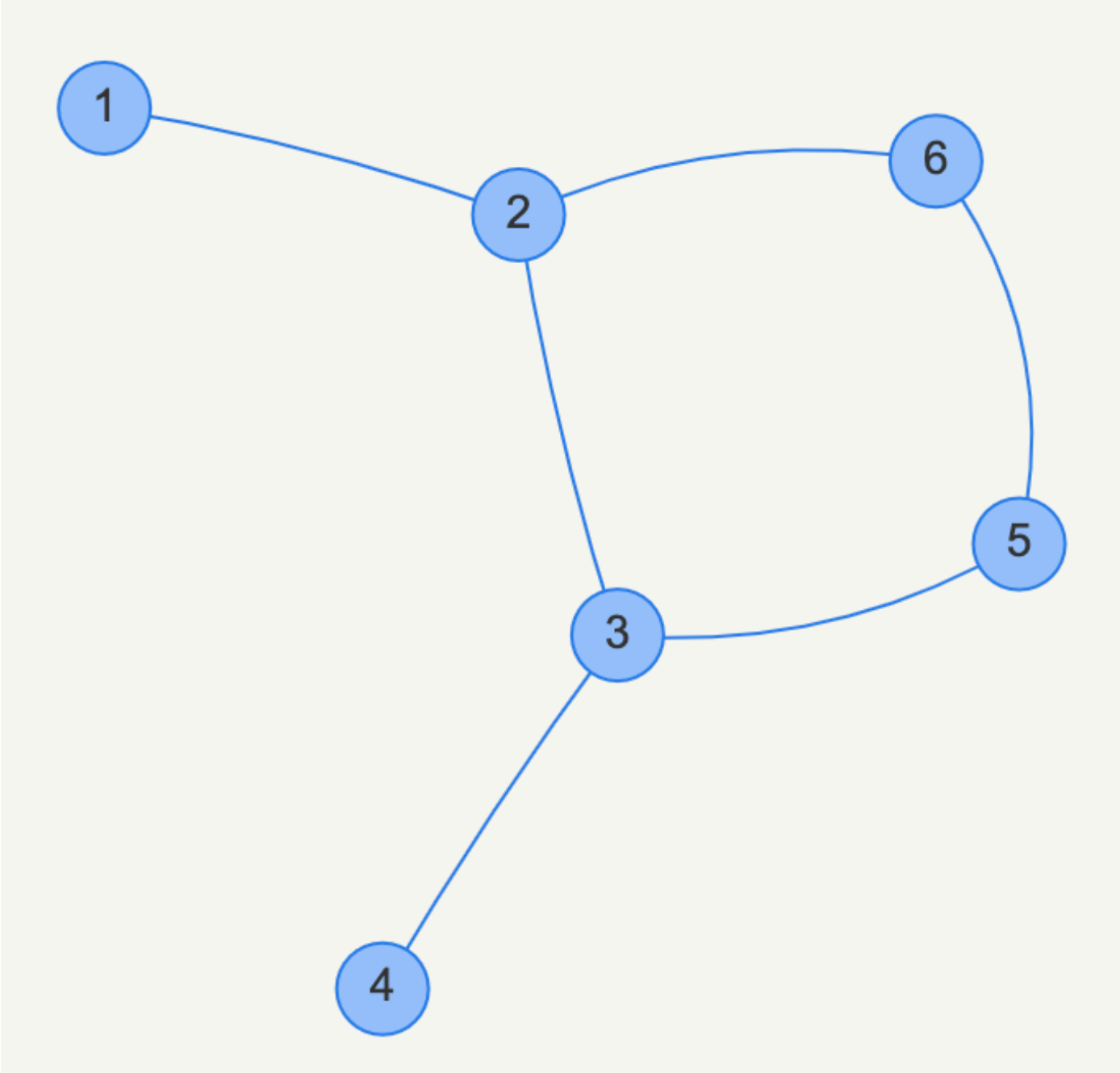
Input Example 3:
episode 6, 100 + 40*3+90 = 310
In episode 6, there're 3 segment [1, 1], [6, 8], [10, 10].
The miniumum number of each segments is 100, 40, 90, multiply each of them with the length of the segment and sum up, the rating is 310.

9_仙人掌 (Opuntia dillenii)

(6分 /9分)

問題敘述

點獨立集是一張圖中一些兩兩不相鄰的節點所形成的集合，而一個點獨立集的點權和就是集合內節點點權的和，而我們可以定義一張圖上的「最大點權獨立集」是這張圖上能找到點權和最大的獨立集。



舉例而言，$\{2, 4, 5\}$ 是這張圖上的其中一組點獨立集，而他的點權和就是這三個點點權的和。

給你一張由 N 個節點與 M 條邊構成的無向簡單連通圖，其中圖上的任意節點都只會存在於至多一個簡單環內，求這張圖中最大點權獨立集的點權和為何？

輸入格式

第一行有兩個正整數 N, M，分別代表點數以及邊數。

接下來的一行中包含了 N 個正整數 $v_1, v_2, \cdots, v_N$，其中 v_i 代表第 i 個節點的權重。

接下來的 M 行中每行有兩個正整數 a_i, b_i，代表節點 a_i 和節點 b_i 有一條邊

- $N \leq 10^5$
- $M \leq 2 \cdot 10^5$
- $v_i \leq 10^3$

輸出格式

請輸出一個正整數代表在這張圖最大點權獨立集的點權和。

子任務

- 子任務 1 滿足 $M = N - 1$ （6分）。
- 子任務 2 沒有特別限制 (9分)。

範例輸入 1

```
10 9
3 9 6 4 7 3 1 2 4 5
10 1
10 4
9 6
3 7
5 8
2 5
3 9
2 3
2 10
```

範例輸出 1

```
23
```

範例說明 1

$\{1, 3, 4, 5, 6\}$ 是這張圖上的最大點獨立集

範例輸入 2

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

```
6 1
2 7
7 8
8 2
5 7
3 9
9 10
10 3
8 9
4 11
11 12
12 4
8 4
```

範例輸出 2

```
31
```

範例說明 2

$\{2, 6, 10, 12\}$ 是這張圖上的最大點獨立集

範例輸入 3

```
9 11
145 851 174 670 571 747 238 391 689
1 4
4 5
5 1
2 6
6 7
7 2
1 7
3 8
8 9
9 3
7 9
```

範例輸出 3

```
2210
```

範例說明 3

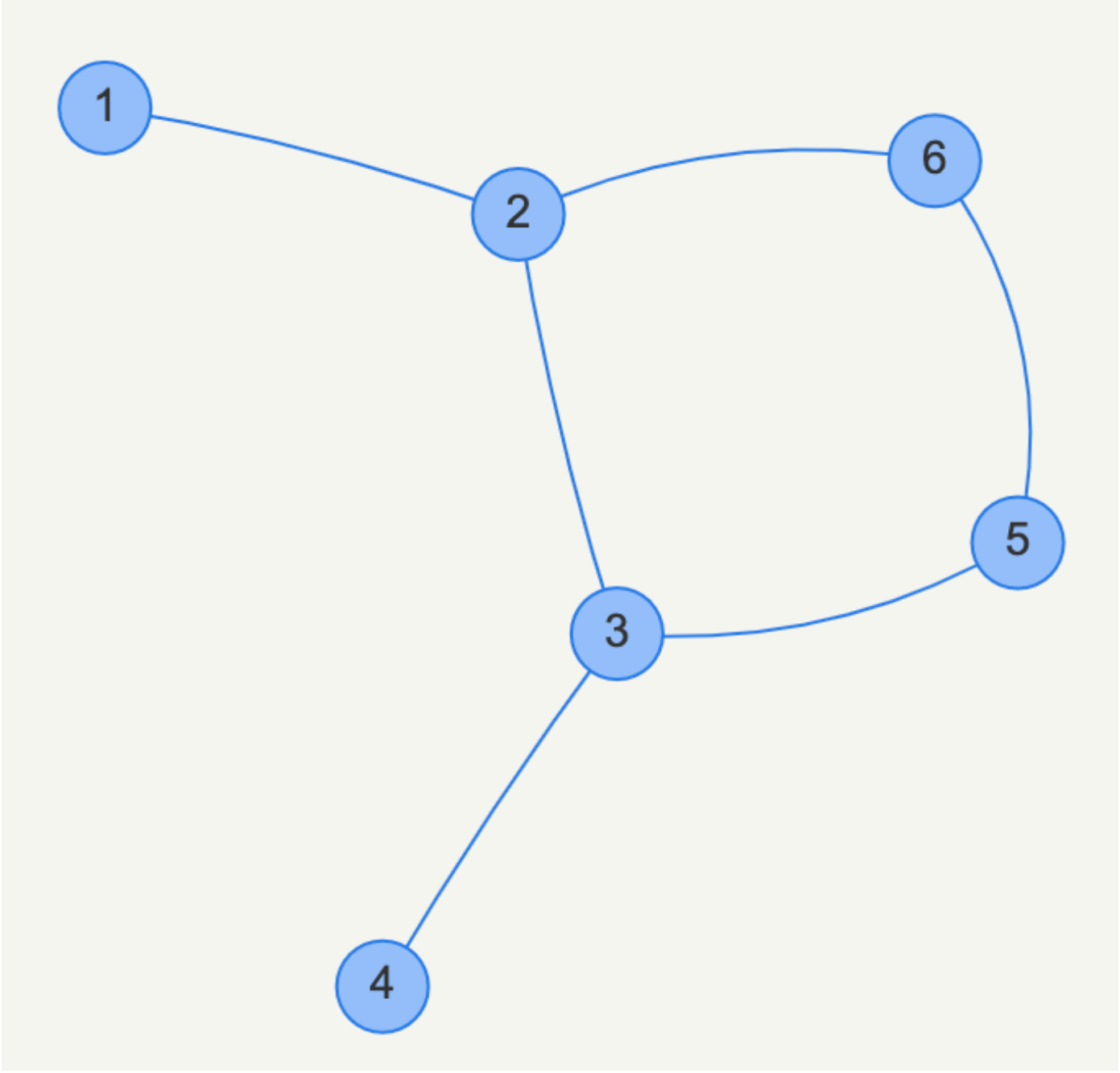
$\{2, 4, 9\}$ 是這張圖上的最大點獨立集

9_Opuntia dillenii

(6 points /9 points)

Problem Statement

An independent set of a graph is a set of vertices in a graph with no two of which are adjacent. The sum of an independent set is the sum of the value of the vertices in the set. We can then define "Maximum independent set" for a graph as the independent set we can find in the graph with the maximum sum.



For example, $\{2, 4, 5\}$ is a valid independent set of the graph.

Given a simple connected graph with N vertices and M edges, while each vertex in the graph will be in at most one cycle. Please find the sum of the maximum independent set.

Input Format

The first line of the input contains two integers N, M -— the number of vertices and the number of edges in the graph.

The second line of the input contains N integers $v_1, v_2, \cdots, v_N$ —- the value of the i-th vertex.

The following M line of the input each contains two integers a_i, b_i -- there is an edge between vertex a_i and b_i in the graph

- $N \leq 10^5$
- $M \leq 2 \cdot 10^5$
- $v_i \leq 10^3$

Output Format

Print the sum of the maximum independent set.

Subtasks

- Subtask 1 satisfy that $M = N - 1$ (6 points).
- Subtask 2 has no other constraint (9 points).

Sample Input 1

```
10 9
3 9 6 4 7 3 1 2 4 5
10 1
10 4
9 6
3 7
5 8
2 5
3 9
2 3
2 10
```

Sample Output 1

```
23
```

Sample description 1

$\{1, 3, 4, 5, 6\}$ is the maximum independent set.

Sample Input 2

```
12 15
1 9 2 1 6 8 9 2 3 4 7 10
1 5
5 6
6 1
2 7
7 8
8 2
5 7
3 9
9 10
10 3
8 9
4 11
11 12
12 4
8 4
```

Sample Output 2

```
31
```

Sample description 2

$\{2, 6, 10, 12\}$ is the maximum independent set.

Sample Input 3

```
9 11
145 851 174 670 571 747 238 391 689
1 4
4 5
5 1
2 6
6 7
7 2
1 7
3 8
8 9
9 3
7 9
```

Sample Output 3

```
2210
```

Sample description 3

$\{2, 4, 9\}$ is the maximum independent set.

10_打牌策略 (How to Play)

(20分)

問題敘述

橋牌，一種風靡在世界上的紙牌遊戲，因為規則複雜而帶有競技性受到許多人歡迎。今天精誠企業辦了一場比賽，而在這場比賽中，我們所使用的撲克牌與一般的牌並不一樣，一副牌共有 N 種花色，每種花色共有 M 種數字，而四位玩家每位會拿到 $\frac{N\times M}{4}$ 張牌。
在橋牌進行中，你看完牌後發現為了得到高分，必須滿足某些條件才有可能，你想知道完成這些條件中至少 y 個的機率有多高。
條件可以有下列兩種之一:

1. 對方在某種花色的分配滿足一定情況，例如在某個 $N=4$，$M=13$ 的牌局，你在某個花色拿了 9 張，而你希望上家與下家分別恰拿 2 張。
2. 某張牌恰好在上家或下家。

輸入格式

輸入第一行有兩個整數N,M，意思於題敘中敘述。

第二行有 N 個整數 a_i ，分別代表你在第 i 種花色所持有的張數，保證總和為 $\frac{N\times M}{2}$

第三行有兩個整數 x,y，代表有 x 個限制條件，以及需要至少完成 y 個條件才能得分

之後有 x 行，每行代表一個限制：

每行一開始會有一個整數 t_j。

- 如果 $t_j=1$，之後會跟隨兩個整數 c_j,d_j ，代表上家在第 c_j 種花色要有恰 d_j 張，保證不會有相同的條件。
- 反之若 $t_j=2$， 之後跟隨兩個整數 e_j,f_j 代表你希望花色 e_j 的某張牌在上家(當$f_j=1$)或下家(當$f_j=2$)。

你可以假設每個 $t_j=2$ 所代表的特定牌都不一樣，且花色總數不會超過對方所持有的張數。

輸出格式

如果完成的機率是 $\frac{P}{Q}$，請輸出 $P\times Q^{-1}(MOD\ 998244353)$
其中我們稱$Q'=Q^{-1}(MOD\ 998244353)$，若 $Q\times Q'=1(MOD\ 998244353)$。

資料範圍

- $1\le N\le 20, 1\le M\le 10^5$，且$N\times M$為4的倍數。
- $1\le a_i\le M$。
- $0\le y\le x\le 15$。
- $1\le t_j\le 2$。
- $1\le c_j\le N$。
- $0\le d_j\le M-a_{c_j}$。
- $1\le e_j\le N$。

- $1 \leq f_j \leq 2$。

輸入範例1

```
4 13
7 7 6 6
1 1
2 1 1
```

輸出範例1

```
499122177
```

範例說明1

該牌在上家的機率為:$1 \times 2^{-1} (MOD\ 998244353)$

輸入範例2

```
4 13
7 7 6 6
1 0
2 1 1
```

輸出範例2

```
1
```

範例說明2

因為不須完成條件，因此無論如何都能完成

輸入範例3

```
4 13
7 7 6 6
2 2
1 1 5
1 1 4
```

輸出範例3

```
0
```

範例說明3

需同時完成兩個矛盾的條件，因此機率為0

10_How to Play

(20 points)

Description

Bridge, a kind of well-known poker game in the world, is popular for the complex rules and competitiveness. Today, SYSTEX is holding a Bridge tournament. In this game, we use special poker, which has N denomination suits and every suit has M numbers. Each player will have $\frac{N\times M}{4}$ cards in their hands. During the Bridge game, if the cards satisfies one of the following two conditions, the player will get a high score. Please calculate the probability that satisfies at least y conditions in all x conditions.

Following are the two conditions

1. In one suit the card satisfies an exactly distributed.
   EX: In the $N = 4, M = 13$ game. In suit 1, you and your partner have 9 cards and you want both of your opposites to have 2 cards in this suit.
2. One exact card is at the left or right opposite.

Input Format

There are two integers N, M in the first line. The meaning has been explained in the description.

There are N integer a_i in the second line. Which is the number of cards in the ith suit you have. The sum of a_i is equal to $\frac{N\times M}{2}$.

There are two integers x, y in the third line. The meaning has been explained in the description.

There are x lines below, every line is one condition.

In every next x line, it will begin with an integer t_j.

- If $t_j = 1$, it will follows with two intger c_j, d_j ,which means the left opposite must have d_j cards in suit c_j.
- Else if $t_j = 2$, it will follows with two intger e_j, f_j ,which means one exact card with suit e_j must be at the left(when $f_j = 1$)or right(when $f_j = 2$) opposite.

You can assume that every $t_j = 2$'s card is different, and the sum of suit i will not overflow the number of opposites have in that suit i.

Output Format

If the probability is $\frac{P}{Q}$.Please output $P \times Q^{-1}(MOD\ 998244353)$
We call $Q' = Q^{-1}(MOD\ 998244353)$,if $Q \times Q' = 1(MOD\ 998244353)$。

Constraints

- $1 \leq N \leq 20, 1 \leq M \leq 10^5$, and $N \times M = 0(MOD\ 4)$
- $1 \leq a_i \leq M$

- $0 \leq y \leq x \leq 15$
- $1 \leq t_j \leq 2$
- $1 \leq c_j \leq N$
- $0 \leq d_j \leq M - a_{c_j}$
- $1 \leq e_j \leq N$
- $1 \leq f_j \leq 2$

Sample Input 1

```
4 13
7 7 6 6
1 1
2 1 1
```

Sample Output 1

```
499122177
```

Sample1 Explanation

The probability with that card on left opposite is $\frac{1}{2}$:$1 \times 2^{-1} (MOD\ 998244353)$

Sample Input 2

```
4 13
7 7 6 6
1 0
2 1 1
```

Sample Output 2

```
1
```

Sample2 Explanation

You can always finish.

Sample Input 3

```
4 13
7 7 6 6
2 2
1 1 5
1 1 4
```

Sample Output 3

```
0
```

Sample3 Explanation

you can't finish this 2 conditions at the sametime.

11_我想玩個遊戲 （I Want to Play a Game）

(20 分)

問題敘述

MM跟TT發明了一個新遊戲。在這個遊戲中，由MM先手，兩人輪流改變遊戲盤面。
一開始，這個盤面有N堆石頭，第i堆有a_i顆石頭。
在每個玩家的回合，這名玩家可以選擇以下兩種操作。

1. 從一個**多於一顆石頭**的石頭堆中移去一顆石頭。
2. 選一個石頭堆，這個石頭堆中的石頭數必須滿足：存在一個大於1的整數k，使得$k(k+1)/2$剛好是這個石頭堆的石頭數。移除這個石頭堆，並放另外k堆石頭到這個盤面中，其中第i堆石頭有i顆石頭。也就是，將石頭數為$1, 2, 3, \ldots, k$的石頭堆加入到盤面中。

無法進行任何操作的玩家輸。
現在，MM跟TT遊玩這個遊戲T次，每次給定N跟一個數列a_i
假設MM跟TT都使用最佳策略遊玩，請問誰會成為贏家？

輸入格式

第一行包含一個整數T。
接下來包含T筆遊戲的初始狀態。
每場遊戲的初始狀態由兩行表示。
第一行包含一個整數N。
第二行有N個整數，第i個代表a_i

輸出格式

輸出T行，代表每場遊戲的結果。
如果MM贏，輸出"MM"，不然輸出"TT" (不包含雙引號)

資料範圍

- $1 \leq T \leq 20$
- $1 \leq N \leq 200000$
- $1 \leq a_i \leq 10^9$
- $1 \leq \sum N \leq 200000$

輸入範例 1

```
1
3
1 2 3
```

輸出範例 1

```
MM
```

輸入範例 2

```
1
3
1 1 1
```

輸出範例 2

```
TT
```

輸入範例 3

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

輸出範例 3

```
TT
```

範例說明

在輸入範例1中，MM可以將3換成2, 1，接下來TT只能將2換成1，則MM再將剩餘的一個2換成1，則TT無操作可做。

在輸入範例2中，MM一開始就無操作可做，所以TT獲勝。

在輸入範例3中，一種雙方都採取最佳策略的遊戲過程如下：

MM: 5 -> 4, 盤面變為 : 1 2 3 4 4

TT: 3 -> 2, 盤面變為 : 1 2 2 4 4

MM: 4 -> 3, 盤面變為 : 1 2 2 3 4

TT: 4 -> 3, 盤面變為 : 1 2 2 3 3

MM: 2 -> 1, 盤面變為 : 1 1 2 3 3

TT: 2 -> 1, 盤面變為 : 1 1 1 3 3

MM: 3 -> 2, 盤面變為 : 1 1 1 2 3

TT: 3 -> 2, 盤面變為 : 1 1 1 2 2

MM: 2 -> 1, 盤面變為 : 1 1 1 1 2

TT: 2 -> 1, 盤面變為 : 1 1 1 1 1

MM無操作可做，故TT獲勝。

11_I Want to Play a Game

(20 points)

Description

MM and TT have invented a new game. In this game, two players take turns modifying the game state with MM moving first. Initially, the game is set up so that there are N piles of stones, with the ith pile containing a_i stones. During each player's turn, the player do either:

1. Remove a single stone from a chosen pile that contains **more than one stone**.
2. Choose a pile of stones with size equal to $k(k+1)/2$ for some integer $k > 1$, and replace it with k piles of stones, the ith pile containing i stones. i.e. Add piles of stones with size $1, 2, 3, \ldots, k$ into the game.

The player who cannot remove any stone loses.
Now, MM and TT would like to play this game for T times, each time given N and a sequence a_i.
Suppose both MM and TT play optimally, who'll be the winner?

Input Format

First line contains an integer T, represents the times that TT and MM would play.
After that, there'll be T sets of N and a sequence a.
In each set, first line contains an integer N.
The second line contains N integers, the ith one represents a_i

Output Format

Print T line, each represents the result of that game.
If MM wins, print "MM", otherwise print "TT" (without quotes).

Constraints

- $1 \leq T \leq 20$
- $1 \leq N \leq 200000$
- $1 \leq a_i \leq 10^9$
- $1 \leq \sum N \leq 200000$

Input Example 1

```
1
3
1 2 3
```

Output Example 1

```
MM
```

Input Example 2

```
1
3
1 1 1
```

Output Example 2

```
TT
```

Input Example 3

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

Output Example 3

```
TT
```

Example Explanation

In Input Example 1, MM would replace 3 with 2 first. In TT's turn, TT can only replace a 2 with a 1, and MM will replace the last 2 with 1. After that, TT will have no operation to do.

In Input Example 2, MM has no operations to do at the beginning, so TT wins.

In Input Example 3, a possible process of the game is given below:

MM: 5 -> 4, the state becomes : 1 2 3 4 4

TT: 3 -> 2, the state becomes : 1 2 2 4 4

MM: 4 -> 3, the state becomes : 1 2 2 3 4

TT: 4 -> 3, the state becomes : 1 2 2 3 3

MM: 2 -> 1, the state becomes : 1 1 2 3 3

TT: 2 -> 1, the state becomes : 1 1 1 3 3

MM: 3 -> 2, the state becomes : 1 1 1 2 3

TT: 3 -> 2, the state becomes : 1 1 1 2 2

MM: 2 -> 1, the state becomes : 1 1 1 1 2

TT: 2 -> 1, the state becomes : 1 1 1 1 1

MM have no operations to do, so TT wins.

12_咖啡漬問題 （Coffee Stains）

(4分 /6分 /10分)

時間限制：5秒

記憶體限制：512 MB

問題敘述

最近打算參加 YTP 的三個人《木木隊》的三人：小木、小林、小森約到了咖啡廳來進行團練。在團練的時候，他們各自點了一杯飲料，而在一整天的刷題之後，發現杯子居然在桌子上留下了一點一點的咖啡漬！被程式沖昏頭的他們，突發奇想想要從咖啡漬中找出他們咖啡杯原本的位子。然而，他們的杯子已經被收走了，連大小都不記得了 —— 你可以幫幫他們找出杯子的大小與在桌子上的位置嗎？

桌子可以表示為一個以 $(0,0)$ 為中心的二維平面，而他們找到了 N 個咖啡漬，其中第 i 個咖啡漬的座標為 (x_i,y_i)。保證每一個點的座標都相異。請找出三個圓，使得每一個點都至少在一個圓上面。當然，有可能其中一個人的咖啡杯完全沒有造成咖啡漬，也就是不一定要每個圓都有點在它上面，且保證所有輸入資料有至少一個解。

提示：給定三個點 $A(x_1,y_1),B(x_2,y_2),C(x_3,y_3)$，則三角形 $\triangle ABC$ 的外心 $O(x,y)$ 的座標可表示為

$$x=\frac{(y_3-y_2)\times d_1-(y_2-y_1)\times d_2}{K}$$
$$y=\frac{(x_2-x_1)\times d_2-(x_3-x_2)\times d_1}{K}$$

此處
$d_1=\left(x_2^2+y_2^2\right)-\left(x_1^2+y_1^2\right), d_2=\left(x_3^2+y_3^2\right)-\left(x_2^2+y_2^2\right),$
$K=2\left((y_3-y_2)(x_2-x_1)-(y_2-y_1)(x_3-x_2)\right)$。

輸入格式

輸入的第一行有一個數字 N（$1\le N\le 10^5$），代表有幾個咖啡漬。
接下來的 N 行中，第 i（$1\le i\le N$）行有兩個數字 x_i,y_i （$0\le x_i,y_i\le 10^6$）代表第 i 個咖啡漬的座標。

保證輸入一定有解。

輸出格式

前三行，請各輸出三個數字X_k,Y_k,R_k（$1\le k\le 3$），代表第 k 個咖啡杯的中心位於 (X_k,Y_k)，且半徑為 R_k。**請滿足** $0\le X_k,Y_k,R_k\le 2\times 10^6$。

第四行，請輸出一個長度為 N，且僅由 `1`、`2`、`3` 三個字元所組成的字串 S，其第 i（$1\le i\le N$） 個字元 S_i 代表其屬於第幾個咖啡杯（咖啡杯持有者的名字裡面有幾個「木」）。

要通過一筆測試資料，必須對於所有的 $1\le i\le N$ 都滿足 $\frac{(x_i-X_{S_i})^2+(y_i-Y_{S_i})^2-R_{S_i}^2}{\max(1,R_{S_i}^2)}<10^{-6}$

請注意：

- 不論你有沒有用到三個圓，請 **還是輸出三個圓的資料**，否則會得到 WA！！

倘若有多組可能的解，輸出任意一組即可。

資料範圍

- $1 \leq N \leq 10^5$
- $0 \leq x_i, y_i \leq 10^6$

子任務

此外，這一題有部分分數：

限制	分數
至多一個咖啡杯有造成咖啡漬	4
至多兩個咖啡杯有造成咖啡漬	6
無額外限制	10

輸入範例 1

```
10
2.0 1.0
4.0 1.0
3.0 2.0
3.0 0.0
5.0 10.0
11.0 10.0
8.0 13.0
8.0 7.0
0.0 1.0
0.0 0.0
```

輸出範例 1

```
3 1 1
8 10 3
0 0.5 0.5
1111222233
```

輸入範例 2

```
6
0.5 0.5
1.0 1.0
2.0 2.0
3.0 3.0
4.0 4.0
5.0 5.0
```

輸出範例 2

```
0.75 0.75 0.353553
2.5 2.5 0.707106
4.50 4.5 0.707106
112233
```

輸入範例 3

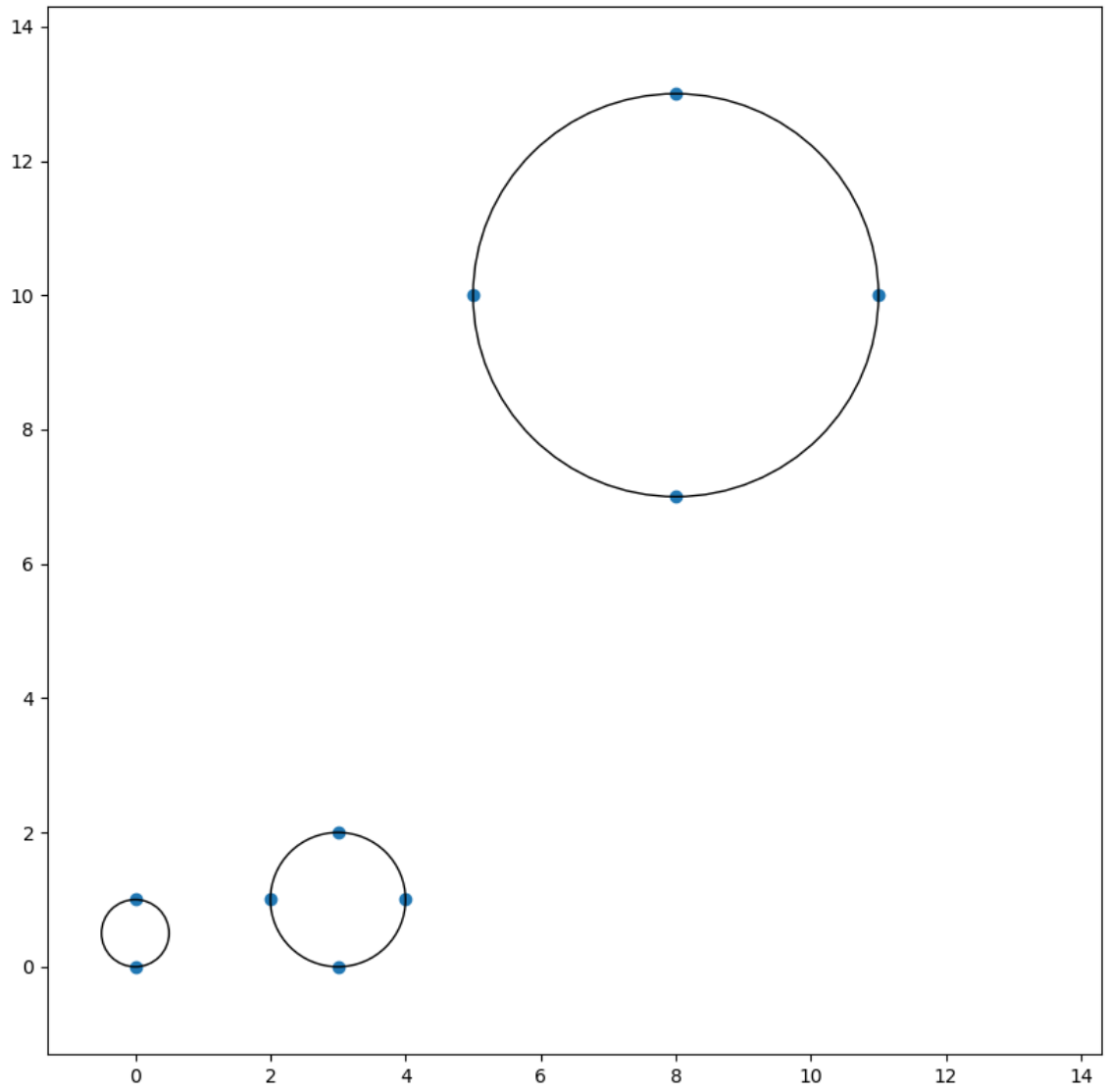
```
1
0.7 1.2
```

輸出範例 3

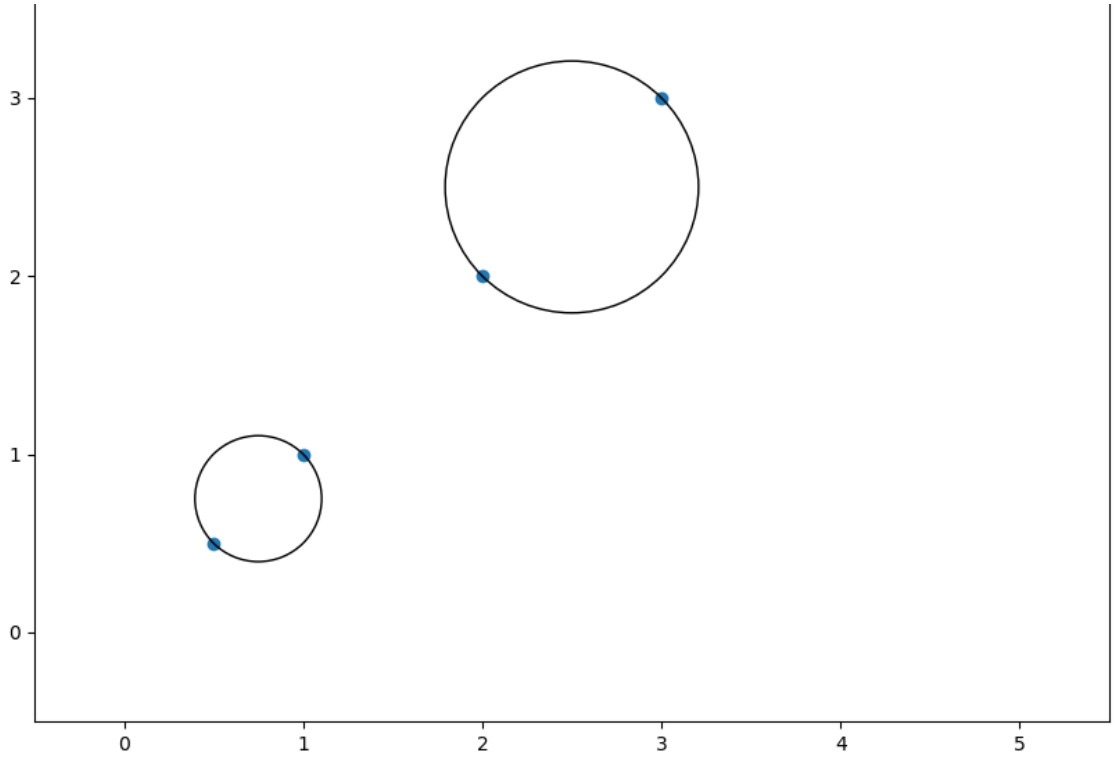
```
0.7 1.2 0.0
0.0 0.0 0.0
0.0 0.0 0.0
1
```

範例解釋

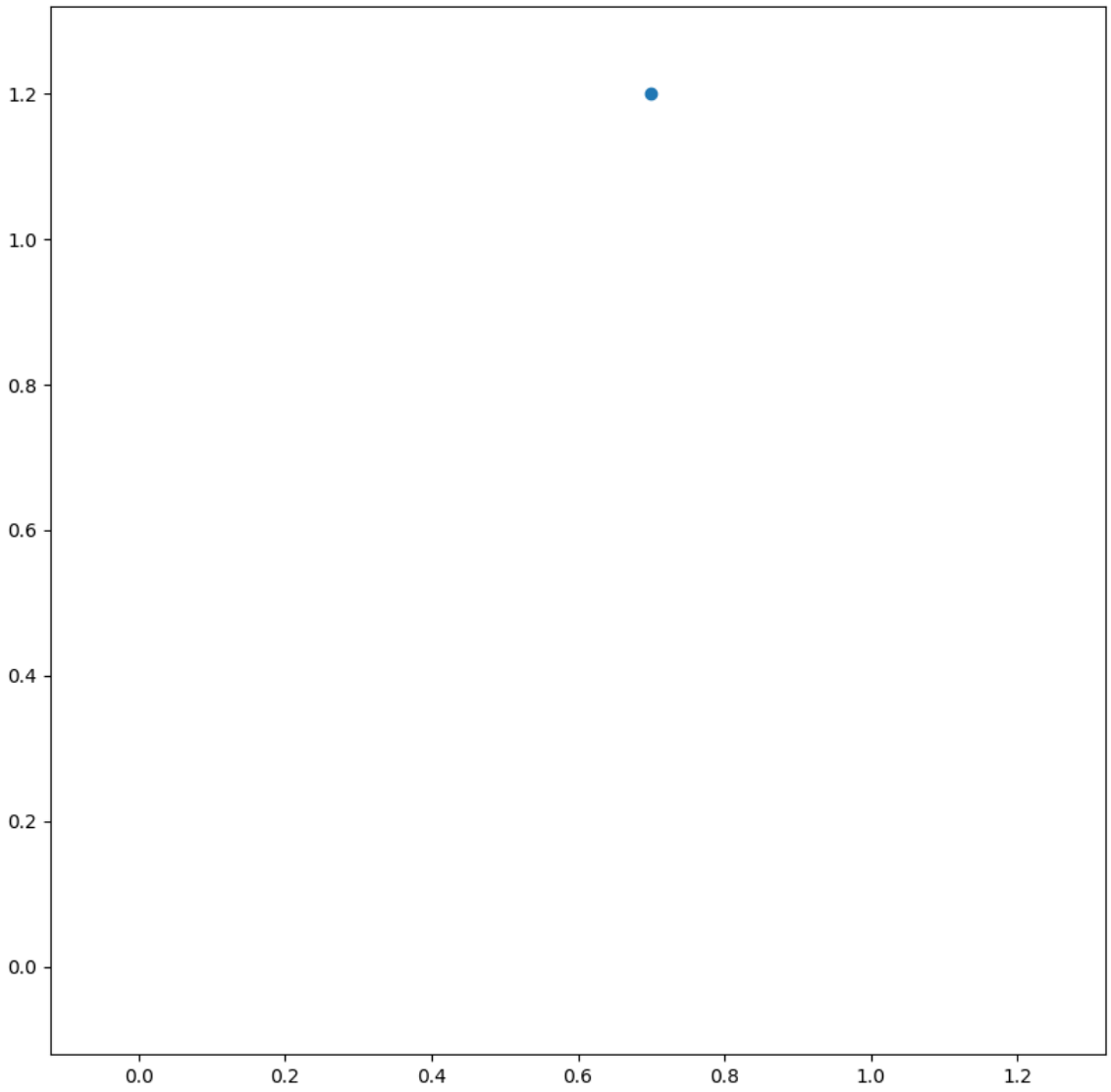
第一筆測試資料：



第二筆測試資料：



第三筆測試資料：


1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2

12_Coffee Stains

(4 points /6 points /10 points)

Time limit: 5 seconds

Memory limit: 512 MB

Description

Team Numeros is hard at work training for this year's YTP programming competition! The three team members, Hito, Futa, and Mi often come to Cafe Bene to practice. They always order a drink each and then start a session.

One day, after a session, they found that they had left coffee stains on the table! Instead of cleaning the mess up, the trio started to wonder if they could reconstruct the positions and sizes of their coffee mugs just from the coffee stains...

The table can be thought of as a Cartesian plane, centered at $(0, 0)$. There are N ($1 \leq N \leq 10^5$) coffee stains represented as points, the i'th of which is located at (x_i, y_i) ($0 \leq x_i, y_i \leq 10^6$). All points have distinct coordinates. Please find three circles such that all the points lie on at least one circle. Of course, it's entirely possible for a mug to not leave any stains, and have no points lie on it.

Hint: Given three points $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$, the circumcentre $O(x, y)$ of triangle $\triangle ABC$ can be given by

$$x = \frac{(y_3 - y_2) \times d_1 - (y_2 - y_1) \times d_2}{K}$$
$$y = \frac{(x_2 - x_1) \times d_2 - (x_3 - x_2) \times d_1}{K}$$

, where

$d_1 = \left(x_2^2 + y_2^2\right) - \left(x_1^2 + y_1^2\right)$, $d_2 = \left(x_3^2 + y_3^2\right) - \left(x_2^2 + y_2^2\right)$, and
$K = 2\left((y_3 - y_2)(x_2 - x_1) - (y_2 - y_1)(x_3 - x_2)\right)$.

Input Format

On the first line, there will be an integer N, the number of coffee stains. Then, for the next N lines that follow, the i'th line will contain two numbers x_i, y_i, the coordinates of the i'th coffee stain.

It's guaranteed that a solution exists for all of the test data.

Output Format

In the first three lines, please output three numbers on each line: X_k, Y_k, R_k, meaning that the k'th ($1 \leq k \leq 3$) circle is centered at (X_k, Y_k), and that its radius is R_k. **In addition, you must have $0 \leq X_k, Y_k, R_k \leq 2 \times 10^6$.**

In the fourth line, please output a string S of length N consisting of the characters `1`, `2`, and `3`. The i'th character of this string S_i is the circle that the i'th point lies on.

To pass the test case, your output must satisfy

$$\frac{(x_i-X_{S_i})^2+(y_i-Y_{S_i})^2-R_{S_i}^2}{\max(1,R_{S_i}^2)} < 10^{-6}$$

for all $1 \leq i \leq N$.

Please note: Regardless of whether you use three circles, please still output the full three circles, otherwise you will receive a WA verdict.

You may output any set of solutions that satisfies the given constraints.

Constraints

- $1 \leq N \leq 10^5$
- $0 \leq x_i, y_i < 10^6$

Subtasks

In addition, there are subtasks which can be solved for partial credit. Do these first!

Additional Constraints	Score
At most one cup caused stains	4
At most two cups caused stains	6
No additional constraints	10

Sample Input 1

```
10
2.0 1.0
4.0 1.0
3.0 2.0
3.0 0.0
5.0 10.0
11.0 10.0
8.0 13.0
8.0 7.0
0.0 1.0
0.0 0.0
```

Sample Output 1

```
3 1 1
8 10 3
0 0.5 0.5
1111222233
```

Sample Input 2

```
6
0.5 0.5
1.0 1.0
2.0 2.0
3.0 3.0
4.0 4.0
5.0 5.0
```

Sample Output 2

```
0.75 0.75 0.353553
2.5 2.5 0.707106
4.50 4.5 0.707106
112233
```

Sample Input 3

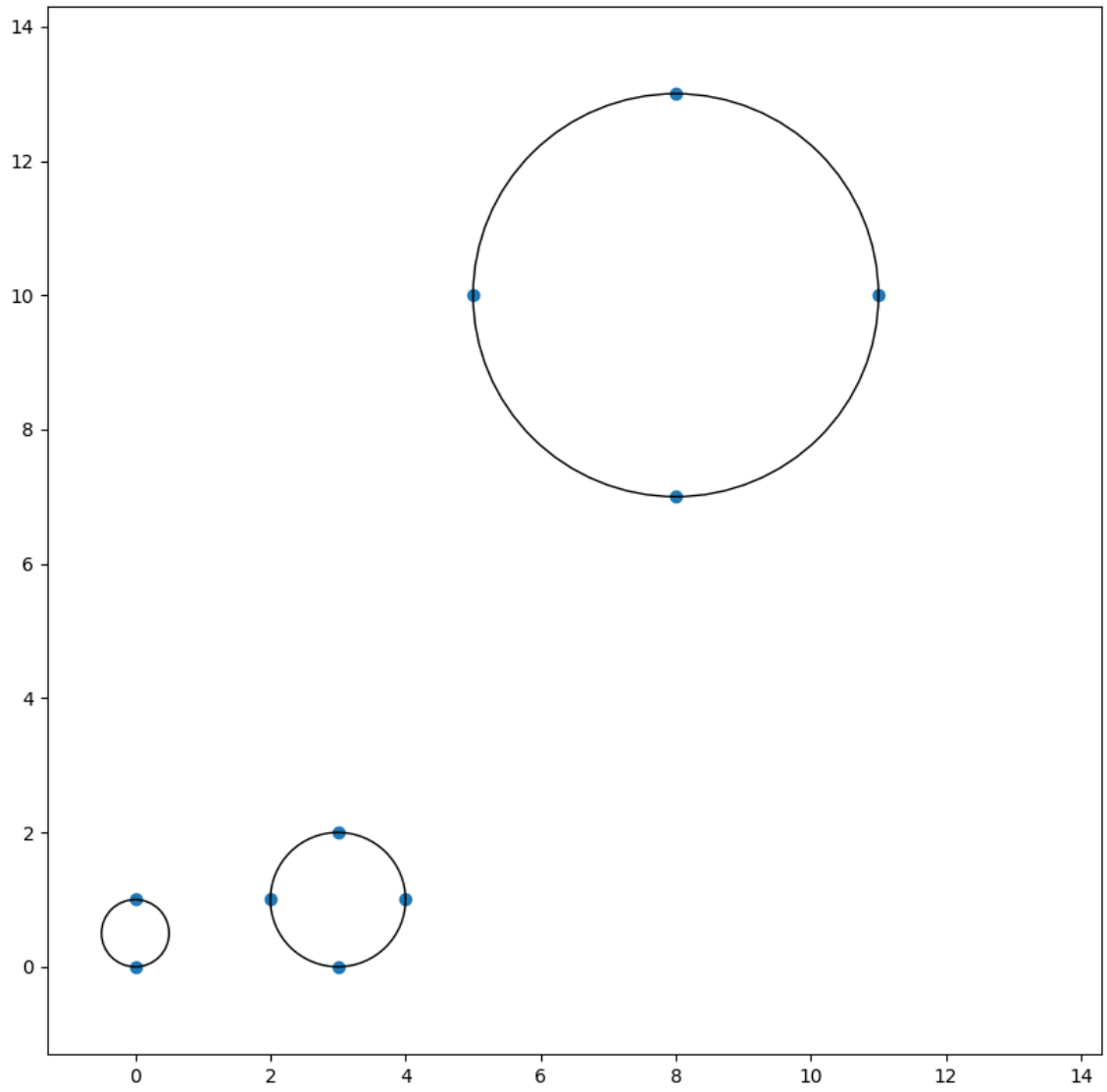
```
1
0.7 1.2
```

Sample Output 3

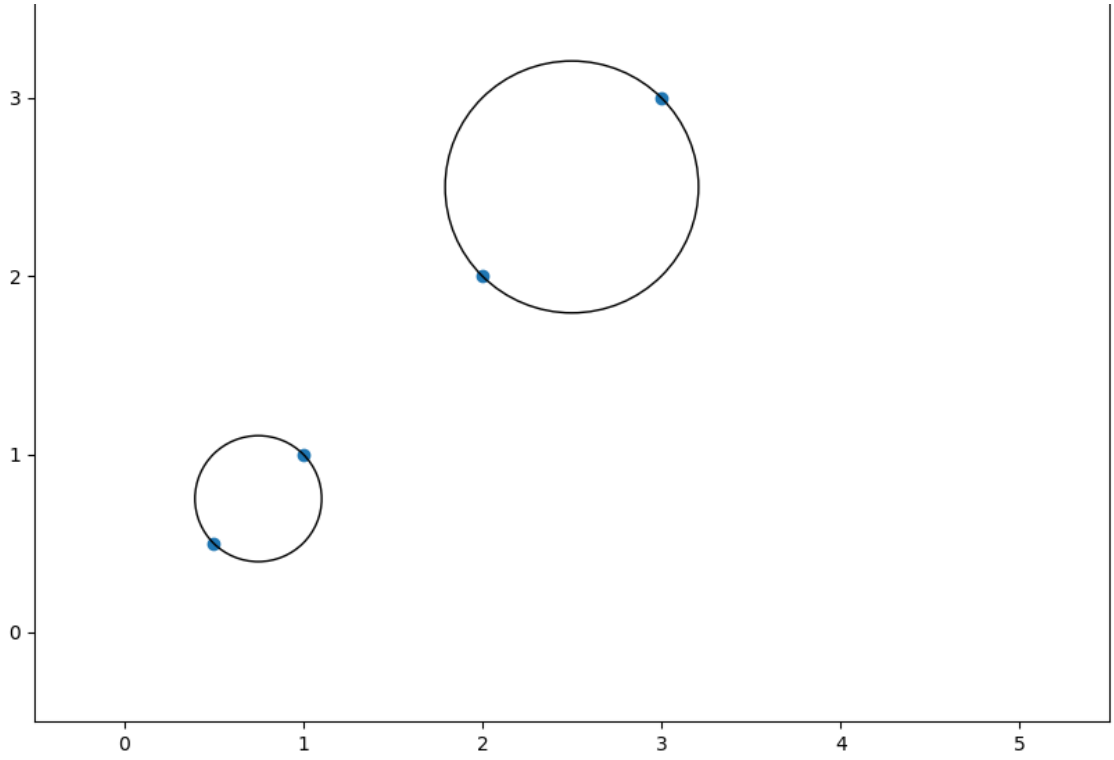
```
0.7 1.2 0.0
0.0 0.0 0.0
0.0 0.0 0.0
1
```

Sample Explanation

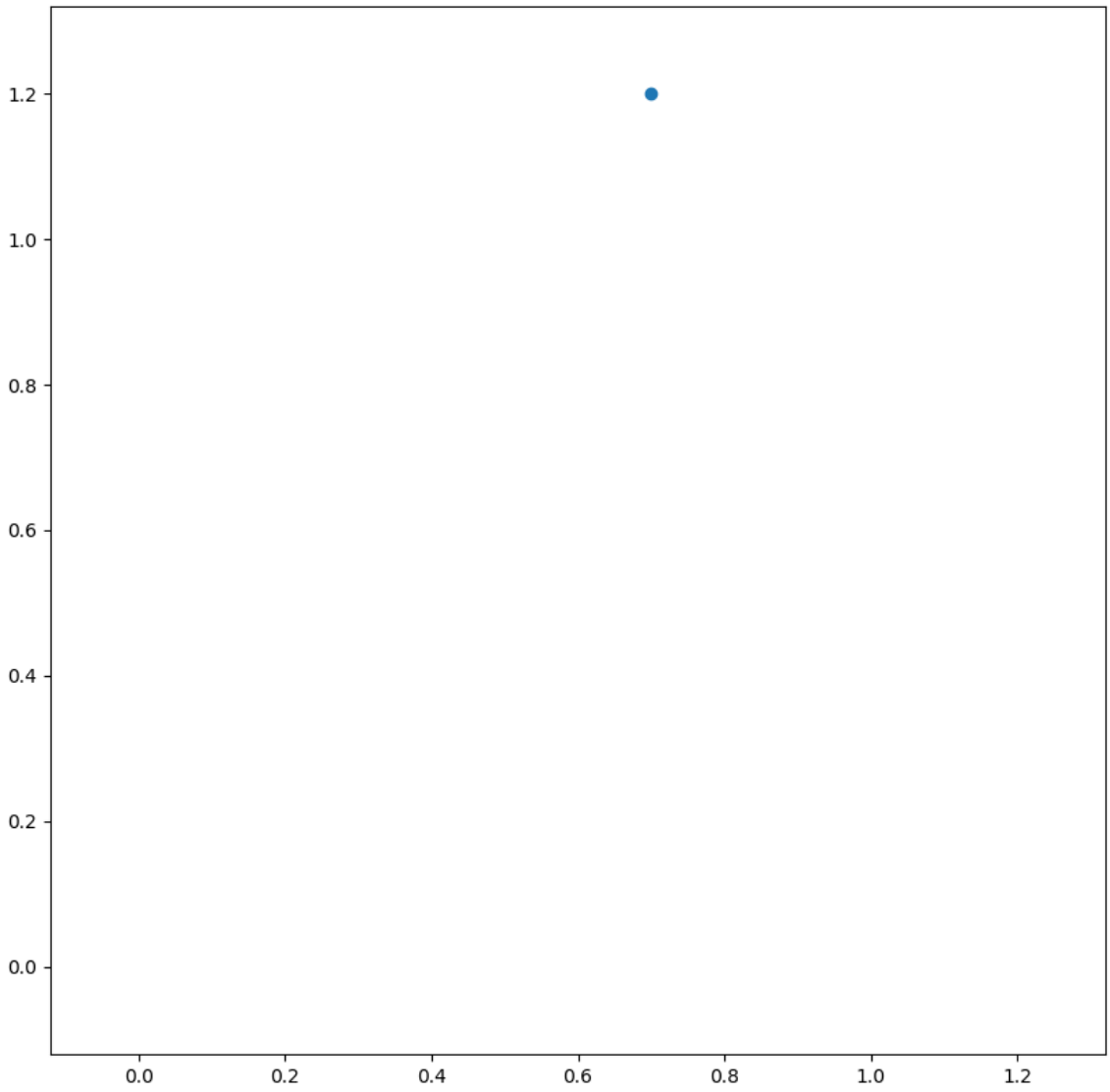
The diagram of the output of the first sample test:



the second sample test:



the third sample test:


1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2

13_買一送一 （Giveaway）

(3分 /17分)

時間限制: 1 second

記憶體限制: 512 MB

問題敘述

蛋餅是 YTP 便利商店的員工，他所工作的分店現在正在進行買一送一的促銷。眾所皆知，買一送一也可以發生在不同的商品之間，只是如果兩個商品的性質或價位差距太大，就不能吸引客人購買。現在店裡一共剩下 N 個商品，編號從 $1 \sim N$，某些商品之間會存在「搭配關係」，店裡所有物品的「搭配關係」的集合稱為這家店的「促銷方案」。由於同樣的商品只有一個，自然不可能在購買的同時贈送商品本身，或者是有兩個搭配關係牽涉到的物品完全相同。有了「促銷方案」後，每當顧客選擇要買一系列的商品，YTP 便利商店會幫顧客從這些商品中選擇一些「搭配關係」，使得被包含在「搭配關係」內的商品都能以買一送一的方式銷售出去，來幫助顧客以最省錢的方法購買到這些商品。這樣選擇搭配關係的行為被稱做一種「銷售方式」。當然，同一個商品是不可以被同時包含在兩個被選中的搭配關係內的，這對商家來說將會是一種虧損。

蛋餅發現，他們店裡這 N 個商品的促銷方案，如果有顧客同時購買 N 個商品，只會有一種可能的銷售方式，而且當前的促銷方案是所有促銷方案中包含最多搭配關係的促銷方案。

麻煩的是，蛋餅把商品全部賣出去之後蛋餅忘了他的銷售方式是什麼，因此他沒辦法好好記帳。請幫蛋餅還原出一種跟他記憶相符的促銷方案。如果有很多種可能的促銷方案，輸出任何一個都會獲得 Accepted；如果不存在這樣的促銷方案，請告訴蛋餅他記錯了。

輸入格式

輸入包含一個正整數 N，代表蛋餅所在的店裡剩下的商品數量。

輸出格式

如果這樣的促銷方案不存在，輸出一行 `-1` 表示蛋餅記錯了。

否則第一行輸出一個非負整數 M，表示有幾個搭配關係。接著輸出 M 行，每行包含兩個正整數 a_i, b_i，表示 a_i 和 b_i 這兩個商品具有搭配關係。

資料範圍

- $1 \leq N \leq 500$

子任務

- 子任務 1 滿足 $N\ le 10$（3分）
- 子任務 2 沒有其他限制（17分）

輸入範例 1

```
2
```

輸出範例 1

```
1
1 2
```

輸入範例 2

```
3
```

輸出範例 2

```
-1
```

輸入範例 3

```
4
```

輸出範例 3

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

範例說明

在第一筆範例中，可以證明只有這一種促銷方案。

在第二筆範例中，明顯不存在滿足條件的促銷方案。

在第三筆範例中，唯一的銷售方式是商品 1 搭配商品 2、商品 3 搭配商品 4，並且可以證明不存在任何一種促銷方案有更多種搭配關係。另外，以下促銷方案也可以獲得 Accepted：

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

此促銷方案唯一的銷售方式是：

```
1 4
2 3
```

以下促銷方案會獲得 Wrong Answer：

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

因為他同時包含：

```
1 2
3 4
```

和

```
2 3
1 4
```

這兩種銷售方式，不滿足題目的條件。

13_Giveaway

(3 points /17 points)

Time limit: 1 second

Memory limit: 512 MB

Description

Omelet works in YTP Convenience Store, which is hosting a buy-one-get-one-free campaign. As everyone knows, buy-one-get-one-free includes buying one item and getting another one for free, but if there is a large gap in price or nature, it can not attract customers to buy them. Now, there are N product in the store, numbered from 1 to N. Some of them exist in "collocation relation", which means that they can be sold in a buy-one-get-one-free relation. All the collocation ralation form a "promotion proposal". Because there is only one item for each product, it is impossible to buy one and get the same one for free. There is also no duplicate collocation ralation because it is unnecessary. Based on the promotion proposal, when a customer is going to buy a series of products, YTP Convenience Store will choose some collocation relation for the customer sush that all the products can be sold in the buy-one-get-one-free way to help customers to purchase these products in the lowest price. The way to choose the collocation relations is called a "sale plan". Of course, one product can not be include by two collocation relation, it is a loss to the store.

Omelet finds out that the promotion proposal in the store makes that there is only one sale plan to sell all the products out in one time, and the promotion proposal in the store includes the most collocation relations among all possible ones.

Unfortunately, Omelet sold all the product but forgot how the sale plan distributed, so he could not keep accounts correctly. Please help Omelet reproduce a promotion proposal according the above condition he remembers. If there are multiple promotion proposal that match his memory, you can print any of them. If the promotion proposal does not exists, please tell Omelet that he misremembered.

Input Format

The input includes a positive integer N, indicating the number of product in the store.

Output Format

If the promotion proposal does not exists, print a line `-1`, indicating that Omelet misremembered.

Otherwise, print a non-negative integer M in the first line, indicating the number of collocation relation(s). Then, print M line(s), each contains two positive integer a_i, b_i, indicating there is a collocation relation between product a_i and b_i.

Constraints

- $1 \leq N \leq 500$

Subtasks

- Subtask 1 satisfies that $N \leq 10$. (3 points)
- Subtask 2 has no additional constraint. (17 points)

Input Example 1

```
2
```

Output Example 1

```
1
1 2
```

Input Example 2

```
3
```

Output Example 2

```
-1
```

Input Example 3

```
4
```

Output Example 3

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

Example Explanation

In the first example, it can be proved that there is only one sale plan.

In the second example, it is trivial that there is no sale plan satisfying the condition.

In the third example, the only sale plan is to sell product 1 with product 2, product 3 with product 4. It can be proved that there is no promotion proposal containing more collocation relations. Moreover, the following promotion proposal can also get Accepted:

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

The only legal sale plan of the above promotion proposal is:

```
1 4
2 3
```

The following promotion proposal will get Wrong Answer:

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

The reason is that it contain:

```
1 2
3 4
```

and

```
2 3
1 4
```

these two sale plans, which violates the condition in the description.

14_RGB 染色遊戲 （RGB Coloring Game）

(4分 /8分 /13分)

時間限制: 3 seconds

記憶體限制: 512 MB

問題敘述

Iofi 是一位繪畫家，一天，她收到來自 Yofi 送給她的一張上色到一半的圖，這張圖由 N 點 M 邊構成。在 Iofi 收到禮物時，她發現所有的邊都已經被 Yofi 分別塗成紅（`R`）、綠（`G`）、藍（`B`）三種顏色的其中一種了，但所有的點卻都沒有上色。

因此，身為繪畫家的 Iofi 希望能同樣幫這些點分別上紅、綠、藍三種顏色的其中一種，不過為了美感，Iofi 決定制定一些規則來防止這些顏色被上得太隨便：

1. 若一條邊是**紅色**的，則他連接的兩個點不可以是「藍色、藍色」或「藍色、綠色」
2. 若一條邊是**藍色**的，則他連接的兩個點不可以是「紅色、紅色」或「紅色、綠色」
3. 若一條邊是**綠色**的，則他連接的兩個點不可以是「藍色、紅色」

注意以上的規則都沒有點的順序性，例如在第 3 條規則中，只要一邊是藍色、一邊是紅色，就是不符合 Iofi 的要求。

除此之外，為了防止整張圖變得綠油油一片，Iofi 還特別限制了一些位置**不可以被塗成綠色**。你能在滿足她的要求下完成塗色嗎？請寫一支程式在輸入 Iofi 得到的圖和她的限制後，給出一組符合規則的方案，或告訴 Iofi 其實不存在任何一種塗色方案可以滿足她的要求。

輸入格式

輸入首行有兩個正整數 N, M，代表點的數量以及邊的數量。

接下來一行一個字串 S，代表 Iofi 針對綠色的限制，若 S 的第 i 個字元為 `1` 表示第 i 個點不可以塗成綠色；`0` 則是沒有限制。

接下來 M 行，每行兩個正整數 u_i, v_i 和一個字元 c_i，代表有一條顏色為 c_i 的邊連接著點 u_i 和 v_i。

其中，`R` 代表紅色、`G` 代表綠色、`B` 代表藍色。

輸出格式

若總是找不到一種塗色方法滿足 Iofi 的要求，輸出 `NO` 於一行。

否則，輸出 `Yes` 於一行後，輸出一個由 `R`, `G`, `B` 三種字元組成、長度為 N 的字串，代表一種符合規則的塗色方案，其中第 i 個字元代表這組方案中要將第 i 個點塗成該顏色。

其中，`R` 代表紅色、`G` 代表綠色、`B` 代表藍色。

若有多種可能符合規則的塗色方案，輸出任何一種皆可。

資料範圍

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq M \leq 5 \times 10^5$
- S 是一個長度恰為 N 的 `01` 字串
- $1 \leq u_i, v_i \leq N, u_i \neq v_i$
- c_i 為 `R`, `G`, `B` 其中一個字元

子任務

- 子任務 1 滿足 $N \leq 10, M \leq 50$。(4分)
- 子任務 2 滿足 S 中的每個字元皆為 `1`。也就是說，沒有點可以被塗成綠色。(8分)
- 子任務 3 沒有特別限制。(13分)

輸入範例 1

```
3 4
110
1 2 B
1 2 R
1 3 G
2 3 G
```

輸出範例 1

```
Yes
BRG
```

輸入範例 2

```
5 7
11001
1 3 B
1 4 R
2 3 B
2 4 G
2 5 G
3 5 R
4 5 B
```

輸出範例 2

```
No
```

輸入範例 3

```
5 6
11111
1 2 B
1 4 R
1 3 R
1 5 B
2 3 B
3 5 G
```

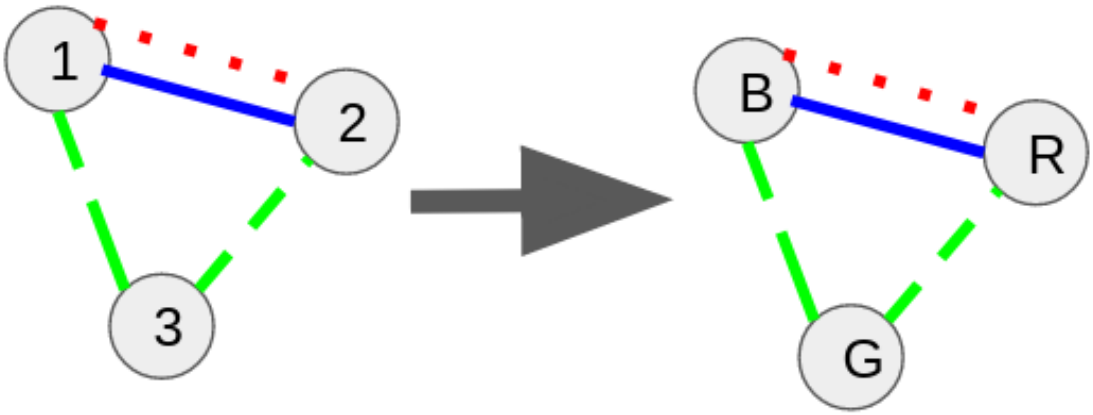
輸出範例 3

```
Yes
BBRRR
```

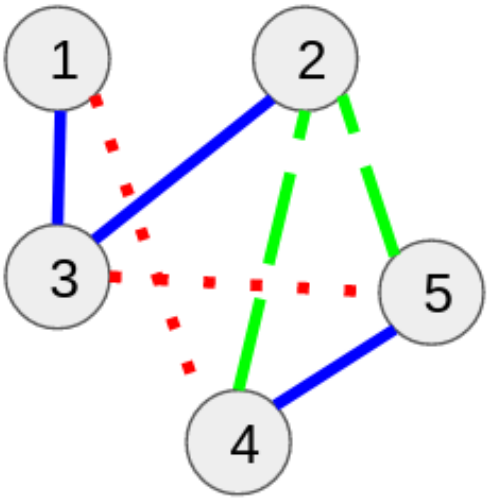
範例說明

- 為了防止圖片被灰階處理，邊除了用顏色示意外，我們也用實線代表藍色、虛線代表綠色、點線代表紅色，而點的顏色則是直接用文字處理。

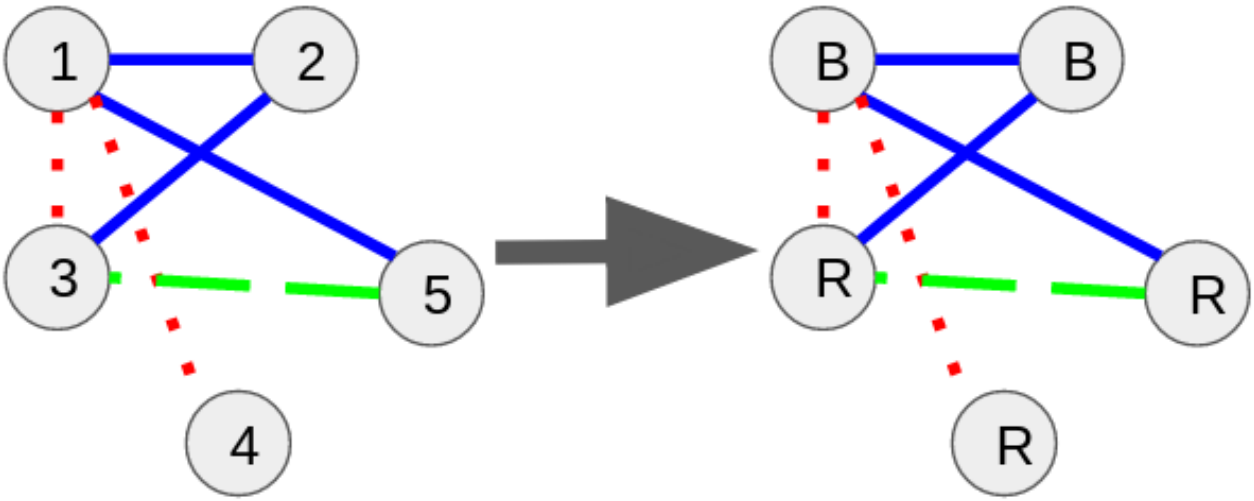
第一筆範例的圖片如下圖，注意到不論輸出 `BRG` 還是 `RBG` 都是正確的。



第二筆範例的圖片如下圖，但無論 lofi 怎麼塗色，她都不可能找到一組合乎規則的塗法，因此你要輸出 `NO`。



第三筆範例的圖片如下圖，注意到由於 $S =$ `11111`，這代表所有的點都不能塗成 `G`，這符合子任務 1 的限制。



14_RGB Coloring Game

(4 points /8 points /13 points)

Time limit: 3 seconds

Memory limit: 512 MB

Description

One day, Iofi, a painter, received a gift from Yofi, a half-colored graph formed by N vertices and M edges. After Iofi received her gift, she found all of the edges were colored red(`R`), green(`G`), or blue(`B`) by Yofi, respectively. But the graph remained non-colored for all vertices.

Therefore, as a painter, Iofi wants to color red, green, or blue for all the vertices just as the edges. But for beauty, Iofi decides to define some rules to prevent these colors from being applied too casually.

1. If an edge is colored **red**, then the two vertices connected by it cannot be colored "blue, blue" or "blue, green", respectively.
2. If an edge is colored **blue**, then the two vertices connected by it cannot be colored "red, red" or "red, green", respectively.
3. If an edge is colored **green**, then the two vertices connected by it cannot be colored "blue, red", respectively.

Notice that there isn't any order of the vertices in the above rules. Take the third rule as an example, as long as one side is blue and the other side is red, it does not meet the requirements of Iofi.

Besides, in order to prevent the whole graph from turning green everywhere, Iofi also gives some special restrictions that we **cannot color green** on some vertices. Can you give a coloring under her constraint? Please write a program that outputs a coloring that fits her rules after reading the graph Iofi received and her constraint. Or tell Iofi that there isn't any coloring that works to her rules.

Input Format

The first line of the input contains two positive integers N, M, indicating the number of vertices and edges.

The second line of the input contains a string S, indicating the constraint about green given by Iofi. If the i-th character of S is `1`, then the i-th vertex cannot be colored green; or `0` for no constraint.

Then, following by M lines, each line contains two positive integers and one character, indicating that there is an edge with color c_i that connects vertices u_i and v_i.

Among them, `R` represents red, `G` represents green, and `B` represents blue.

Output Format

If there isn't any coloring that works to Iofi's rules, output `NO` in a single line.

Otherwise, output `Yes` in the first line, and then output a string with length N, which compose of characters `R`, `G`, `B`, indicating a coloring that fit the rules. The i-th characters represent that you want to color the i-th vertex into that color.

Among them, `R` represents red, `G` represents green, and `B` represents blue.

If there are multiple possible coloring that fit the rules, output any of them.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq M \leq 5 \times 10^5$
- S is a `01` string with length N
- $1 \leq u_i, v_i \leq N, u_i \neq v_i$
- c_i belongs to one of the characters `R`, `G`, `B`

Subtasks

- Subtask 1 satisfies that $N \leq 10, M \leq 50$. (4 points)
- Subtask 2 satisfies that all of the characters of S are `1`. That is, none of the vertices can be colored green. (8 points)
- Subtask 3 has no additional constraint. (13 points)

Input Example 1

```
3 4
110
1 2 B
1 2 R
1 3 G
2 3 G
```

Output Example 1

```
Yes
BRG
```

Input Example 2

```
5 7
11001
1 3 B
1 4 R
2 3 B
2 4 G
2 5 G
3 5 R
4 5 B
```

Output Example 2

```
No
```

Input Example 3

```
5 6
11111
1 2 B
1 4 R
1 3 R
1 5 B
2 3 B
3 5 G
```

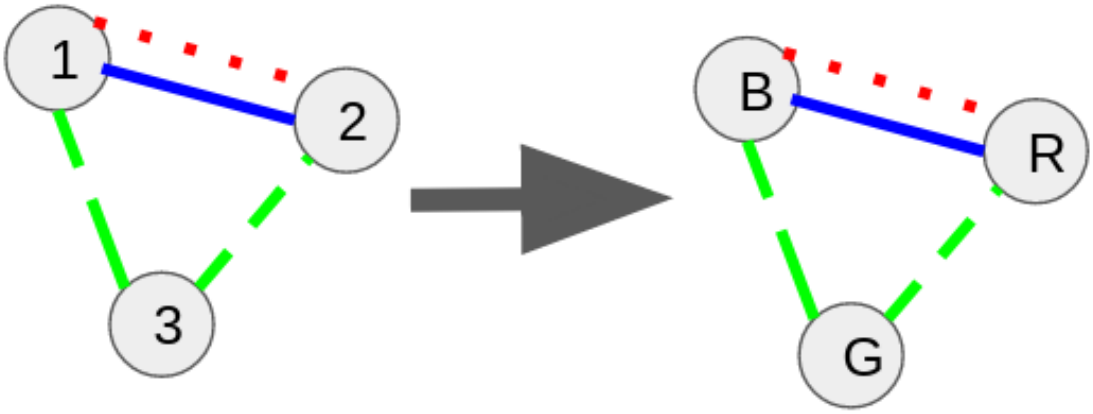
Output Example 3

```
Yes
BBRRR
```

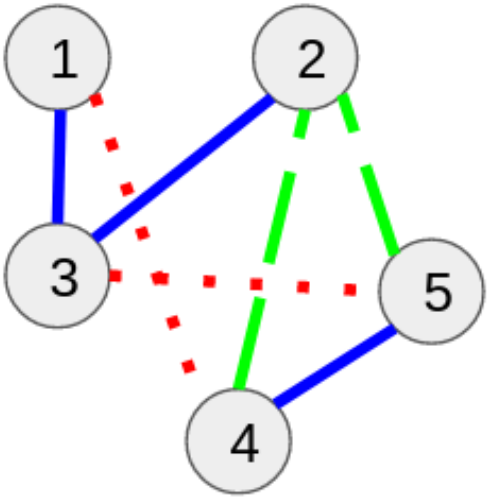
Example Explanation

- In order to prevent the image from being grayscaled, in addition to using color to indicate the edges, we also use solid lines to represent blue, long dotted lines to represent green, short dotted lines to represent red, and the color of the vertices is directly represented by text.

The figure of the first example is shown below, notice that both `BRG` and `RBG` are correct answers.



The figure of the second example is shown below, but no matter how Iofi paints, she can't find a coloring that fit the rules, so you have to output `No`.



The figure of the third example is shown below, notice that because $S =$ `11111`, this means that all vertices cannot be painted with `G`, which fits the constraint of subtask 1.