

## CCUPC X DS 114-2 Final Contest

### 注意事項

1. 本次競賽時間為 13:00 至 18:00，共 300 分鐘。
2. 競賽中僅可查閱紙本參考資料，不得與他人討論測驗題目。
3. 可以使用賽事平台公告中的程式語言文件查看語法。
4. 競賽中提供計分板，最後六十分鐘將停止更新。
5. 若有任何與題目相關的問題，請透過評測系統詢問。
6. 每次提交需要通過該題目所有測試資料才可獲得分數，通過範例測試資料不代表能通過所有測試。
7. 所有題目測試資料皆經由標準輸入流讀入 (stdin)，並經過標準輸出流 (stdout) 輸出，以下為競賽可用語言之輸入輸出的範例程式碼：

#### C++

```
#include <iostream>
int main() {
    int x, y;
    std::cin >> x >> y;
    std::cout << x + y << '\n';
}
```

#### C

```
#include <stdio.h>
int main() {
    int x, y;
    scanf("%d %d", &x, &y);
    printf("%d\n", x + y);
}
```

#### Python

```
x, y = map(int, input().split())
print(x + y)
```

Java

```
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int x = scanner.nextInt();
        int y = scanner.nextInt();
        System.out.println(x + y);
    }
}
```

Kotlin

```
fun main() {
    val x = readLine()!!.trim().toInt()
    val y = readLine()!!.trim().toInt()
    writeln(x + y)
}
```

## Problem Set

ID	Problem Name	Time Limit	Memory Limit
A	畢業典禮	1 s	256 MB
B	回憶牆	1 s	256 MB
C	貼彩帶	2 s	512 MB
D	爬樓梯	1 s	256 MB
E	短路	1 s	256 MB
F	無止盡的比賽	1 s	256 MB
G	學術資源解密	3 s	256 MB
H	傳送門最短路	3 s	256 MB
I	最後的捷徑	1 s	256 MB
J	最後一場比賽	2 s	1024 MB
K	戀愛學分	2 s	1024 MB
L	好想當勝利組	1 s	1024 MB
M	那些年錯過的愛情	2 s	512 MB
N	心動的訊號	1 s	1024 MB

This page is intentionally left blank.

## Problem A 畢業典禮

Time limit: 1 second

Memory limit: 256 megabytes

### Problem Description

六月，是畢業的季節

你身為一個專業的畢業典禮司儀，但是你卻不小心把最重要的校長撥穗順序給弄亂了，你只知道撥穗順序是根據以下三個規定來安排的：

1. 學校會按照學系的字典序依序請各個學院上台
2. 在同一個學系中，獲得校長獎的優秀畢業生必須優先上台領證
3. 對於相同學系且相同獎項的學生，請維持報到順序上台

不過幸運的是，你還留著畢業生的所有報到資料，現在請你按照順序排好畢業生的撥穗順序

### Input Format

第一行輸入一個整數  $n$ ，代表畢業生的總數量。

接下來  $n$  行的每一行都有 3 個內容，分別為學系名稱 (字串)  $s_1$ 、畢業生姓名 (字串)  $s_2$ 、是否獲得校長獎 (整數，0 代表沒有，1 代表有)。

輸入順序代表報到順序。

### Output Format

請依序輸出所有上台畢業生的名字，每行一個名字。

### Technical Specification

- $1 \leq n \leq 2 \times 10^5$
- $s_1$  僅包含大寫英文字母，不含空格
- $s_2$  僅包含大小寫英文字母，不含空格
- $3 \leq |s_1|, |s_2| \leq 10$

### Sample Input 1

```
5
MATH Alice 0
CSIE Tom 0
MATH Bob 0
MATH Carol 1
CSIE Mark 1
```

### Sample Output 1

```
Mark
Tom
Carol
Alice
Bob
```

This page is intentionally left blank.

## Problem B 回憶牆

Time limit: 1 second

Memory limit: 256 megabytes

### Problem Description

畢業典禮前夕，學校準備在校園中設置一面回憶牆，讓即將畢業的同學們能在離開校園前，把這幾年留下的照片、卡片、活動紀錄與祝福文字整理起來，作為大學生活最後的紀念。

校園中共有  $N$  個地點，這些地點可能是圖書資訊大樓、共同教室大樓、活動中心、管理學院、工學院、理學院，或是中正大學裡某個大家曾經集合、拍照或趕報告的角落。校園中的部分地點之間以路段相連，每條路段皆可雙向通行。整個校園中總共有  $N - 1$  條路段，並且保證從任意一個地點出發，都可以沿著這些路段抵達其他任何一個地點。

第  $i$  個地點保存著  $C_i$  份畢業回憶素材，這些素材可能是社團活動照片、課堂合照、畢業祝福卡片、成果展紀錄，或是某次大家一起在校園裡留下的紀念影像。

主辦同學希望選擇一個地點設置回憶牆，並將附近地點保存的回憶素材整理到牆上。不過，畢業典禮前的準備時間有限，工作人員不可能跑遍整個校園收集所有素材，因此回憶牆只能收錄距離設置地點不太遠的內容。

若某個地點到回憶牆所在位置之間，最少需要經過的路段數不超過  $K$ ，則該地點保存的回憶素材會被納入這面回憶牆的展示內容；若需要經過的路段數超過  $K$ ，則因為距離太遠，不會被收錄。

現在，主辦同學想知道如果把回憶牆分別設置在每一個地點，各自可以收錄到多少份畢業回憶素材。對於每個地點  $i$ ，請你計算所有與地點  $i$  的最短路段距離不超過  $K$  的地點中，保存的回憶素材總數。請依照地點編號由小到大，輸出每個地點作為回憶牆位置時，能夠收錄到的回憶素材總數。

### Input Format

第一行包含兩個整數  $N$  和  $K$ ，分別表示校園中的地點數量，以及回憶牆能收錄素材的最大路段距離。

接下來  $N - 1$  行，每行包含兩個整數  $u$  和  $v$ ，表示地點  $u$  與地點  $v$  之間有一條可雙向通行的路段。最後  $N$  行，每行包含一個整數  $C_i$ ，表示第  $i$  個地點保存的畢業回憶素材數量。

### Output Format

請依照地點編號由小到大輸出答案，共  $N$  行。第  $i$  行包含一個整數，表示若將回憶牆設置在第  $i$  個地點，所有符合收錄距離限制的地點中，畢業回憶素材的總數。

### Technical Specification

- $1 \leq N \leq 10^5$
- $1 \leq K \leq 20$
- $0 \leq C_i \leq 1000$

- 路段總數量為  $N - 1$  個
- 任意兩個地點之間皆可透過路段互相抵達
- $1 \leq u, v \leq N$
- $u \neq v$

### Sample Input 1

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

### Sample Output 1

```
15
21
16
10
8
11
```

## Problem C 貼彩帶

Time limit: 2 seconds  
Memory limit: 512 megabytes

### Problem Description

畢業典禮的裝飾現場需要貼上大量的彩帶，現在有一條長度為  $n$  的神奇彩帶，彩帶上每個位置都編織了一個特殊的字元。你手邊有  $m$  個現成的花紋模板，每個模板都是一個由字元組成的字串。你想要對彩帶進行一連串的操作。操作分為兩種：

1. 區間覆蓋：拿某個花紋模板，從彩帶的某個指定位置開始覆蓋過去。
2. 種類查詢：詢問彩帶在某個區間內，目前「總共出現了幾種不同的字元」。

為了讓彩帶看起來和諧，彩帶與模板上所使用的字元只會由大寫英文字母 (A-Z)、小寫英文字母 (a-z) 與數字 (0-9) 組成。身為一名優秀的競賽程式設計師，你能快速計算出每次詢問的答案嗎？

### Input Format

輸入第一行包含三個正整數  $n, m, q$ ，分別代表彩帶長度、模板數量以及操作筆數。

第二行包含一個長度為  $n$  的字串  $S$ ，代表彩帶一開始各個位置的字元。

接下來有  $m$  行，第  $i$  行 ( $1 \leq i \leq m$ ) 包含一個字串  $P_i$  ( $1 \leq |P_i| \leq n$ )，代表第  $i$  個模板。保證所有模板的長度總和  $\sum_{i=1}^m |P_i| \leq 2 \times 10^5$ 。

接下來有  $q$  行，每行代表一個操作，格式如下：

- $1 \ pos \ id$  ( $1 \leq pos \leq n, 1 \leq id \leq m$ )：將從位置  $pos$  開始的彩帶區段，替換為第  $id$  個模板  $P_{id}$ 。保證操作合法，即  $pos + |P_{id}| - 1 \leq n$ 。
- $2 \ l \ r$  ( $1 \leq l \leq r \leq n$ )：詢問目前彩帶在區間  $[l, r]$  內共有幾種不重複的字元。

### Output Format

對於每個操作 2，輸出一個整數，代表該區間內不重複的字元數量。每個答案各佔一行。

### Technical Specification

- $1 \leq n, q \leq 10^5$
- $1 \leq m \leq 100$
- $c \in \{A-Za-z0-9\}, \forall c \in S, P_i$
- $\sum_{i=1}^m |P_i| \leq 2 \times 10^5$

### Sample Input 1

```
8 2 4
aB01aB01
XyZ
99
2 2 6
1 3 1
1 6 2
2 2 7
```

### Sample Output 1

```
4
5
```

### Note

在範例測資 1 中，對於彩帶 aB01aB01：

- 2 2 6：B01aB 有 4 種字元 ( B,0,1,a )。
- 1 3 1：彩帶變為 aBXyZB01。
- 1 6 2：彩帶變為 aBXyZ991。
- 2 2 7：BXyZ99 有 5 種字元 ( B,X,y,Z,9 )。

## Problem D 爬樓梯

Time limit: 1 second  
Memory limit: 256 megabytes

### Problem Description

6/6 是中正大學的畢業典禮，阿呆起床發現他居然睡過頭了，衝到大禮堂氣喘吁吁，他想要用最省體力的方式衝上樓梯。

這段樓梯共有  $N$  階，每階的高度為  $h_i$ 。小明從第 1 階出發，目標是抵達第  $N$  階。他每次可以跳 1 到  $K$  階（即從第  $i$  階跳到任意第  $j$  階，其中  $i < j \leq i + K$ ）。從第  $i$  階跳到第  $j$  階時，需要消耗  $|h_i - h_j|$  的體力。

請問小明從第 1 階抵達第  $N$  階，最少需要消耗多少體力？

### Input Format

第一行包含兩個正整數  $N$  與  $K$ ，分別代表樓梯階數與每次最多可跳的階數。

第二行包含  $N$  個正整數  $h_1, h_2, \dots, h_N$ ，代表每階樓梯的高度。

### Output Format

輸出一個整數，代表小明從第 1 階抵達第  $N$  階所需消耗的最少體力。

### Technical Specification

- $2 \leq N \leq 10^5$
- $1 \leq K \leq \min(N - 1, 100)$
- $1 \leq h_i \leq 10^9$

### Sample Input 1

```
5 2
10 30 40 20 50
```

### Sample Output 1

```
40
```

This page is intentionally left blank.

## Problem E

### 短路

Time limit: 1 second

Memory limit: 256 megabytes

### Problem Description

隊長帝維畢業後靠著競程大家的幫助建立了帝維帝國，帝維帝國有  $N$  個城市（編號為 1 至  $N$ ）與  $M$  條連接城市的雙向道路，每條道路都有各自的通行時間。在這些城市中，有  $K$  個城市被指定為「核心物流中心」。

為了應對未來可能發生的天災，帝國運輸部想要評估每個城市的「備援能力」。我們定義任何一個城市  $u$  的「備援距離」為**從城市  $u$  出發，到達兩個不同的核心物流中心的最短路徑長度之和**。

請計算並輸出所有城市的備援距離。如果某個城市無法到達至少兩個不同的物流中心，則該城市的備援距離定義為  $-1$ 。

### Input Format

第一行包含三個正整數  $N, M, K$ ，分別代表城市數、道路數與核心物流中心的數量。

第二行包含  $K$  個互不相同的正整數，代表這  $K$  個核心物流中心的城市編號。

接下來  $M$  行，每行包含三個正整數  $u, v, w$ ，代表城市  $u$  與城市  $v$  之間有一條通行時間為  $w$  的雙向道路。

### Output Format

輸出包含  $N$  行。第  $i$  行輸出一個整數，代表城市  $i$  的備援距離。

### Technical Specification

- $2 \leq N \leq 10^5$
- $1 \leq M \leq 2 \times 10^5$
- $2 \leq K \leq N$
- $1 \leq u, v \leq N, u \neq v$
- $1 \leq w \leq 10^4$
- 圖可能不連通，也可能有重邊。

**Sample Input 1**

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

**Sample Output 1**

```
5
2
5
7
2
```

## Problem F 無止盡的比賽

Time limit: 1 second  
Memory limit: 256 megabytes

### Problem Description

畢業季將至。在 324 教室的角落，兩位傳奇選手正在進行他們大學生涯最後一場練習賽。這場比賽對他們而言，不僅是技術的切磋，更是對這四年競程生涯的回憶。

比賽一局接著一局進行，每一局必會分出勝負，勝者得 1 分。爲了讓這場最後的對決更有趣味，他們約定了一個規則：

「只有當其中一隊**領先達到 3 分**時，這段競爭關係才會劃下句點，勝方贏得最終的榮耀並結束比賽。」

然而，或許是兩人實力過於接近，又或者是潛意識裡誰也不想先說再見，比賽進行了驚人的  $n$  局，卻依然沒有分出最後勝負（意即在第 1 局到第  $n$  局的任何時刻，兩隊的分數差距從未達到 3 分）。

坐在窗邊看著夕陽，你突然好奇，這段長達  $n$  局且「尚未結束」的回憶，究竟有多少種可能的勝負序列？由於可能性非常多，請將答案對  $10^9$  取模（ $\text{mod } 10^9$ ）。

### Input Format

輸入僅包含一個整數  $n$ ，代表目前已經進行的比賽局數。

### Output Format

輸出一個整數，代表  $n$  局比賽後，所有符合題目條件（比賽未結束）的可能勝負序列種數對  $10^9$  取模後的結果。

### Technical Specification

- $1 \leq N \leq 10^{18}$

#### Sample Input 1

4

#### Sample Output 1

12

#### Sample Input 2

5

#### Sample Output 2

18

This page is intentionally left blank.

## Problem G 學術資源解密

Time limit: 3 seconds

Memory limit: 256 megabytes

### Problem Description

資工系的傳奇學長「虔敬」今天就要畢業離校了。在實驗室的四年生活裡，他除了寫程式，最自豪的就是他那高達 10TB 的「愛情片」收藏。

學長深知這些資源對學弟妹的身心健康（與學術研究）至關重要，因此他在離校前，將這些珍貴檔案存在了實驗室伺服器的祕密隱藏資料夾中。然而，爲了不讓這些檔案被教授發現，也爲了考驗學弟妹的演算法實力，他將這  $T$  個影片檔案全部加密了。

每個檔案的名稱都是一個正整數  $N$ 。而該檔案的解鎖密碼，就是  $N$  的**最大質因數**。

學長留下一張紙條：「這是最後的溫柔。系辦將在 1 秒後格式化這顆硬碟，如果你們能在那之前算出這一百萬個密碼，這些學術資源就傳承給你們了！」

身爲渴望知識的學弟妹，請趕快寫出程式，在硬碟被格式化前拯救這批珍貴資源！

### Input Format

第一行包含一個正整數  $T$ ，代表學長留下的影片檔案總數。

接下來的  $T$  行，每行包含一個正整數  $N$ ，代表每個影片的檔案編號。

### Output Format

對於每一個檔案編號  $N$ ，請輸出其解鎖密碼（即  $N$  的最大質因數）。

### Technical Specification

- $1 \leq T \leq 3 \times 10^6$
- $2 \leq N \leq 10^7$

#### Sample Input 1

```
5
2
3
10
10000000
9999991
```

#### Sample Output 1

```
2
3
5
5
9999991
```

This page is intentionally left blank.

## Problem H 傳送門最短路

Time limit: 3 seconds

Memory limit: 256 megabytes

### Problem Description

你被困在一個  $R \times C$  的矩形二維迷宮中，若無法在時間內逃脫，你的畢業證就要被蔡少正收走了。迷宮由空地（標記為  $.$ ）與牆壁（標記為  $\#$ ）組成。在迷宮的某些空地中，隱藏著  $N$  個未激活的量子傳送陣（標記為  $K$ ）。你的目標是激活所有的傳送陣。你可以自由選擇從任意一個傳送陣作為起點激活並出發。在迷宮中移動時，規則如下：

- 步行移動：你可以從當前格子移動到上下左右四個相鄰的空地（ $.$  或  $K$ ），每移動一格需要消耗 1 單位能量。你不能穿過牆壁。
- 傳送移動：一旦某個傳送陣被你激活（即你步行到達過該格子），它就會永久保持激活狀態。此後，你可以在任意兩個「已激活」的傳送陣之間進行瞬間傳送，且傳送不消耗任何能量。

請計算出：為了將這  $N$  個傳送陣全部激活，你最少需要消耗多少單位的能量？如果因為牆壁阻隔，導致某些傳送陣無論如何都無法步行到達，請輸出  $-1$ 。

### Input Format

第一行包含兩個正整數  $R$  和  $C$ ，代表迷宮的列數與行數。

接下來  $R$  行，每行包含一個長度為  $C$  的字串，代表迷宮的結構。

- $.$  代表普通空地。
- $\#$  代表牆壁。
- $K$  代表未激活的傳送陣（其本質也是空地，可以通行）。

輸入保證迷宮中  $K$  的總數量為  $N$ 。

### Output Format

輸出一個整數，代表激活所有傳送陣所需的最小能量。若無法全部激活，請輸出  $-1$ 。

### Technical Specification

- $1 \leq R, C \leq 1000$
- $1 \leq R \times C \leq 10^5$
- $0 \leq N \leq 1500$

### Sample Input 1

```
6 6
K...##
..#...
K.##.K
##...K
....#.
...K..
```

### Sample Output 1

```
14
```

## Problem I 最後的捷徑

Time limit: 1 second  
Memory limit: 256 megabytes

### Problem Description

畢業典禮前，競程決定安排一場最後的校園巡禮。大家把這幾年待過的教室、實驗室和各地比賽會場整理成  $N$  個回憶地點，編號為  $1, 2, \dots, N$ 。

校園中有  $M$  條捷徑路線。每條捷徑都可以在兩個地點之間雙向通行，通過一條捷徑需要 1 分鐘。

第  $j$  條捷徑以兩個整數  $U_j, V_j$  表示。若  $U_j \neq 0$ ，表示這條捷徑已經確定連接地點  $U_j$  與地點  $V_j$ 。若  $U_j = 0$ ，表示這條捷徑目前只有一端確定連接地點  $V_j$ ，另一端尚未決定。

社團想嘗試  $N$  種不同的巡禮版本。對於每個  $i = 1, 2, \dots, N$ ，在第  $i$  種版本中，所有尚未決定端點的捷徑都會被接到地點  $i$ 。也就是說，所有滿足  $U_j = 0$  的捷徑，都會變成連接地點  $i$  與地點  $V_j$ 。

請你對每個  $i$ ，計算在第  $i$  種版本的地圖中，從地點 1 走到地點  $N$  最少需要幾分鐘。如果無法從地點 1 抵達地點  $N$ ，請輸出  $-1$ 。

注意，補上未定端點後，可能會出現一條捷徑連接某個地點到自己，也可能會有多條捷徑連接同一對地點。

### Input Format

第一行包含兩個整數  $N, M$ ，分別代表回憶地點數量與捷徑數量。

接下來  $M$  行，第  $j$  行包含兩個整數  $U_j, V_j$ ，代表第  $j$  條捷徑。

若  $U_j = 0$ ，表示這條捷徑有一端尚未決定，另一端固定連接地點  $V_j$ 。若  $U_j \neq 0$ ，表示這條捷徑確定連接地點  $U_j$  與地點  $V_j$ 。

### Output Format

請輸出  $N$  個整數，整數之間以一個空白分隔。

第  $i$  個整數代表：當所有未定端點的捷徑都接到地點  $i$  時，從地點 1 到地點  $N$  的最短時間。

若在該情況下無法從地點 1 抵達地點  $N$ ，則第  $i$  個整數應輸出  $-1$ 。

### Technical Specification

測試資料滿足以下限制：

- $2 \leq N \leq 3 \times 10^5$
- $1 \leq M \leq 3 \times 10^5$
- $0 \leq U_j < V_j \leq N$

- 對於任意  $a \neq b$ ，皆保證  $(U_a, V_a) \neq (U_b, V_b)$
- 輸入中的所有數值皆為整數

### Sample Input 1

```
3 2
0 2
1 2
```

### Sample Output 1

```
-1 -1 2
```

### Sample Input 2

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

### Sample Output 2

```
3 3 3 3 2
```

## Note

### Sample 1

當所有尚未決定端點的捷徑都接到地點 1 時，第 1 條捷徑與第 2 條捷徑都會連接地點 1 與地點 2。此時無法從地點 1 抵達地點 3。

當所有尚未決定端點的捷徑都接到地點 2 時，第 1 條捷徑會連接地點 2 與它自己，而第 2 條捷徑連接地點 1 與地點 2。此時仍然無法從地點 1 抵達地點 3。

當所有尚未決定端點的捷徑都接到地點 3 時，第 1 條捷徑會連接地點 3 與地點 2，而第 2 條捷徑連接地點 1 與地點 2。在這種情況下，可以用 2 分鐘從地點 1 走到地點 3。

- 使用第 2 條捷徑，從地點 1 走到地點 2。
- 使用第 1 條捷徑，從地點 2 走到地點 3。

因此，應依序輸出  $-1, -1, 2$ 。

注意，根據尚未決定端點的捷徑被接到哪個地點，可能會出現連接某個地點到自己的捷徑，也可能會有多條捷徑連接同一對地點。

## Problem J 最後一場比賽

Time limit: 2 seconds

Memory limit: 1024 megabytes

### Problem Description

畢業前，*DejaVu* 準備了一份最後的回憶題單。

題單共有  $N$  題，編號為  $1, 2, \dots, N$ 。一開始，只有第 1 題可以挑戰。

對於每個可以挑戰的第  $i$  題，其中  $1 \leq i \leq N - 1$ ，你可以選擇以下兩種解法之一：

- 花費  $A_i$  秒解出第  $i$  題，之後第  $i + 1$  題會變成可以挑戰。
- 花費  $B_i$  秒解出第  $i$  題，之後第  $X_i$  題會變成可以挑戰。

除了實際解題花費的時間以外，其他時間皆忽略不計。

請問最少需要花費幾秒，才能讓第  $N$  題變成可以挑戰？

### Input Format

第一行包含一個整數  $N$ ，代表題單中的題目數量。

接下來  $N - 1$  行，第  $i$  行包含三個整數  $A_i, B_i, X_i$ 。

$A_i$  代表使用第一種解法解出第  $i$  題所需的秒數，並會解鎖第  $i + 1$  題。

$B_i$  代表使用第二種解法解出第  $i$  題所需的秒數，並會解鎖第  $X_i$  題。

### Output Format

輸出一個整數，代表讓第  $N$  題變成可以挑戰所需的最少秒數。

### Technical Specification

測試資料滿足以下限制：

- $2 \leq N \leq 2 \times 10^5$
- $1 \leq A_i, B_i \leq 10^9$
- $1 \leq X_i \leq N$
- 輸入中的所有數值皆為整數

### Sample Input 1

```
5
100 200 3
50 10 1
100 200 5
150 1 2
```

### Sample Output 1

```
350
```

### Sample Input 2

```
10
1000 10 9
1000 10 10
1000 10 2
1000 10 3
1000 10 4
1000 10 5
1000 10 6
1000 10 7
1000 10 8
```

### Sample Output 2

```
90
```

### Sample Input 3

```
6
1000000000 1000000000 1
1000000000 1000000000 1
1000000000 1000000000 1
1000000000 1000000000 1
1000000000 1000000000 1
```

### Sample Output 3

```
5000000000
```

## Note

範例一

- 花費 100 秒解出第 1 題，之後第 2 題會變成可以挑戰。
- 花費 50 秒解出第 2 題，之後第 3 題會變成可以挑戰。
- 花費 200 秒解出第 3 題，之後第 5 題會變成可以挑戰。

## Problem K 戀愛學分

Time limit: 2 seconds

Memory limit: 1024 megabytes

### Problem Description

網路上流傳著一句話: 大學四年, 要是戀愛學分沒修到, 就要畢業了。

身為競程的一員, 四年來把所有時間都拿去刷題, AC 數突破三千, 戀愛學分卻還是 0。眼看畢業在即, 你決定發起最後的衝刺。

校園裡有  $N$  位令你心動的對象, 編號為  $1, 2, \dots, N$ 。從第 1 天開始, 每天你最多能鼓起勇氣寄出一封情書, 每位對象至多寄一次。

若在第  $d$  天寄情書給第  $i$  位, 對方會考慮  $A_i$  天, 並在第  $d + A_i$  天回覆, 使你獲得  $B_i$  點心動值。

然而, 第  $i$  位將在第  $T_i$  天離開校園: 只有在第  $T_i$  天 (含) 以前送達的回覆才有效; 太晚寄出的話, 這段緣分就錯過了, 什麼也得不到。當然, 你也可以選擇不寄給某些對象。

據說心動值累積得夠多, 就能補修戀愛學分。請求出你能獲得的心動值總和的最大值。

### Input Format

第一行包含一個整數  $N$ , 代表心動對象的數量。

接下來  $N$  行, 第  $i$  行包含三個整數  $A_i, B_i, T_i$ 。

$A_i$  代表第  $i$  位需要考慮的天數,  $B_i$  代表獲得的心動值,  $T_i$  代表第  $i$  位離開校園的日子。

### Output Format

輸出一個整數, 代表心動值總和的最大值。

### Technical Specification

測試資料滿足以下限制:

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq A_i, B_i, T_i \leq 10^9$
- 輸入中的所有數值皆為整數

### Sample Input 1

```
3
1 5 2
1 8 2
2 3 4
```

### Sample Output 1

```
11
```

### Sample Input 2

```
4
1 1000000000 1000000000
2 1000000000 1000000000
3 1000000000 1000000000
1000000000 1000000000 999999999
```

### Sample Output 2

```
3000000000
```

### Sample Input 3

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

### Sample Output 3

```
26
```

## Note

範例一

第 1 天只能寄一封，因此可以選擇寄給第 2 位對象，得到 8 分。

第 2 天再寄給第 3 位對象，會在第 4 天收到回覆，仍然有效，得到 3 分。

總分為  $8 + 3 = 11$ 。

*P.S.* 在你遇到真的很有感覺，很喜歡的人之前，不在一起，就是不交到也沒關係。

## Problem L 好想當勝利組

Time limit: 1 second

Memory limit: 1024 megabytes

### Problem Description

「我覺得那位叫夜白的學長看起來很厲害，希望有機會能多向他學習。」

在夜黑風高的夜晚裡，夜白學長就像螢火蟲一般閃耀，總能在迷茫之中指引大家走向光明、幸福、美滿的初音未來。

爲了在畢業前不留下遺憾，你決定從現在開始努力提升自己的見識與能力，朝著成爲像夜白學長一樣的勝利組邁進。無論是在 ICPC 賽事中大有斬獲、在課業上學得更加充實，還是再次投稿一篇論文，這些努力都會一點一滴累積成你的「勝利指數」。

根據傳說中的「螢火蟲發光定律」，第  $n$  天的勝利指數  $F_n$  定義如下：

$$F_1 = x_1$$

$$F_2 = x_2$$

當  $n \geq 3$  時，

$$F_n = a \times F_{n-2} + b \times F_{n-1}$$

不過在多天累積後實力可能超級強，所以勝利指數需要對 1000000007 取模，現在給定  $x_1, x_2, a, b, n$ ，請你計算第  $n$  天的勝利指數  $x_n$ 。

### Input Format

輸入只有一行，有五個正整數，依序爲  $x_1, x_2, a, b, n$ ，數值間以空白隔開。

### Output Format

由於  $x_n$  的數值可能很大，請輸出  $x_n$  除以 1000000007 的餘數。

### Technical Specification

- $0 \leq x_1, x_2, a, b \leq 10^9$
- $3 \leq n \leq 10^9$

#### Sample Input 1

0 1 1 1 5
-----------

#### Sample Output 1

3
---

#### Sample Input 2

0 1 1 1 50
------------

#### Sample Output 2

778742000
-----------

This page is intentionally left blank.

## Problem M

### 那些年錯過的愛情

Time limit: 2 seconds

Memory limit: 512 megabytes

#### Problem Description

世界上最難過的莫過於明明互相喜歡卻擦肩而過，就像那幾年在黑板上一起討論的競程題目，你很認真的在解題，我欲言又止，也許能跟喜歡的人一起解題目也是一種浪漫。

題目如下：

有一個正整數  $x$  將會以長度為  $n$  的二進位字串形式給你看，你可以對這個數字進行兩種操作：

1. 將  $x$  替換為  $\lfloor \frac{x}{2} \rfloor$ ，其中  $\lfloor \frac{x}{2} \rfloor$  代表小於或等於  $\frac{x}{2}$  的最大整數（下取整）。
2. 將  $x$  替換為  $\lceil \frac{x}{2} \rceil$ ，其中  $\lceil \frac{x}{2} \rceil$  代表大於或等於  $\frac{x}{2}$  的最小整數（上取整）。

我們會持續進行這兩種類型的操作，直到  $x$  變為 1 為止，每一次操作時，他都會以  $\frac{1}{2}$  的機率獨立且隨機地選擇執行第一種操作或第二種操作，希望你可以計算將  $x$  變為 1 所需執行的操作次數期望值對  $10^9 + 7$  取模後的結果。

#### Input Format

輸入的第一行包含一個正整數  $t$  ( $1 \leq t \leq 10^5$ )，代表接下來有幾筆測資，每組測資包含兩行：

第一行包含一個正整數  $n$  ( $1 \leq n \leq 10^5$ )，代表整數  $x$  的二進位表示法的長度。

第二行包含一個長度為  $n$  的二進位字串：代表整數  $x$  的二進位表示，保證該字串的最高有效位元必定為 1。

#### Output Format

對於每組測試資料，輸出一個整數，代表我們將  $x$  變為 1 所需執行操作次數的期望值對  $10^9 + 7$  取模後的結果。

#### Technical Specification

- $1 \leq t \leq 10^5$
- $1 \leq n \leq 10^5$
- $\sum n \leq 10^5$
- 設  $M = 10^9 + 7$ ，可以證明最終的精確答案一定可以表示為一個最簡分數  $\frac{p}{q}$ ，其中  $p$  和  $q$  皆為整數且  $q \not\equiv 0 \pmod{M}$ ，請輸出一個整數  $x$ ，滿足  $0 \leq x < M$  且  $x \cdot q \equiv p \pmod{M}$ ，即輸出  $p \cdot q^{-1} \pmod{M}$ 。

---

**Sample Input 1**

```
4
3
110
3
100
10
1101001011
31
1000111110000101101000110100100
```

**Sample Output 1**

```
500000006
2
193359386
5201314
```

## Problem N 心動的訊號

Time limit: 1 second

Memory limit: 1024 megabytes

### Problem Description

世界上最難過的莫過於明明互相喜歡卻擦肩而過，就像那幾年在黑板上一起討論的競程題目，你很認真的在解題，我欲言又止，也許能跟喜歡的人一起解題目也是一種浪漫。

在中正競程有個傳說，每人身上都有一些特徵值，傳說若兩人特徵值完全相同，即可成為所有人羨慕的情侶，而所有特徵值相異或相同的排列，則會構成各種不同的關係，例如朋友、隊友，愛情的部分，有些人這輩子可以長長久久，而有些人注定有緣無分，或許能有在競程共同解題的回憶，已是命運給的最大寬容，謝謝妳，謝謝中正競程，讓我平凡的人生短暫的璀璨過。

而友情的部分則沒有那麼嚴格的條件，只要兩人有任一特徵值相同兩人就是朋友，幸運的是朋友的朋友也算是朋友，這讓大學的生活更加精采，眾多好友的陪伴或許也是大學不枯燥的關鍵，當有人問你某兩人是否為朋友時，希望你能清楚的分辨兩人關係。

### Input Format

輸入的第一行包含一個正整數  $n$  代表人數。

接著對於每人第一行有一個數字  $num$ ，代表該人特徵數量。對於每人第二行有  $num$  個數字  $f$ ，代表此人特徵。

接下來有一個數字  $q$ ，代表詢問數量。接下來有  $q$  行輸入，每行有兩個正整數  $a, b$ ，代表詢問的兩人。

### Output Format

對於每筆詢問，輸出 Yes 或 No，代表兩人是否為朋友。

### Technical Specification

- $1 \leq n \leq 10^6$
- $1 \leq num \leq 10^6$
- $\sum n \times num \leq 10^6$
- $1 \leq f \leq 10^6$
- $1 \leq q \leq 10^6$
- $1 \leq a, b \leq n$

### Sample Input 1

```
5
3
1 3 5
3
2 3 6
1
6
2
10 11
2
11 13
5
1 3
1 4
4 5
2 4
2 3
```

### Sample Output 1

```
Yes
No
Yes
No
Yes
```

### Hint

本題需使用 io 加速

```
ios_base::sync_with_stdio(0);
```

```
cin.tie(0);
```