

2022 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。
- 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。
- 任何題目內提到的「一行」，皆代表以換行字元「`\n`」結尾的字串。
- 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「相對或絕對誤差不超過 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。

Problem	Problem Name	Time Limit	Memory Limit
A	雅量	1 s	1024 MB
B	選巫醫巫醫	1 s	1024 MB
C	巴乙己分組問題	2 s	1024 MB
D	教授分球球	1 s	1024 MB
E	真島與莉可麗絲	1 s	1024 MB
F	城市規劃	1 s	1024 MB
G	魔族與訓練	3 s	1024 MB
H	買餅乾	1 s	1024 MB

2022 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. 雅量

Problem ID: generosity

朋友出了一場模擬比賽，沒用的範例帶冗長的敘述。當她拿給我們看時，一位對手速十分感興趣的同學說：

「啊，好像 Codeforces 似的。」

「我看倒有點像唬爛大賽。」我說。

「真像一題題 IMO。」一位外號叫「方格王」的同學緊接著說。

我們不禁哄堂大笑，同樣的一場比賽，每個人卻有不同的感覺。那位朋友連忙把題本用紙袋裝好，她覺得模擬比賽就是模擬比賽，不是 Codeforces，也不是唬爛大賽，更不是 IMO。

現在，給你兩個正整數 N, M ，請求出 $\sum_{1 \leq i \leq N} (\lfloor \log_2 i \rfloor \cdot \frac{N!}{i})$ 除以 M 的餘數。

Input

輸入第一行包含一個正整數 T 。接下來有 T 行，每行有兩個以空白隔開的正整數 N, M 。

- $1 \leq N \leq 10^{18}$
- $1 \leq M \leq 10^6$
- $1 \leq T \leq 30000$
- $\sum N \leq 10^{18}$
- $\sum M \leq 10^6$
- 所有的輸入都是正整數。

Output

對於每一組 N, M ，請輸出一行包含一個整數，代表 $\sum_{1 \leq i \leq N} (\lfloor \log_2 i \rfloor \cdot \frac{N!}{i})$ 除以 M 的餘數。

Sample Input 1	Sample Output 1
3 3 2 5 4 9982 10007	1 0 4798

B. 選巫醫巫醫

Problem ID: absolute

你知道嘛，巫醫巫醫為了選繼承人，每四年就會舉辦一次巫醫巫醫爭奪戰，要讓最強衛冕者拿到傳奇巫醫巫醫的頭銜。

總共有 N 名參賽者，編號從 1 到 N ，其中編號為 i 的參賽者的巫力為 a_i ，他們即將在象大山的女神競技場打架。

比賽的方法如下：

每次由巫醫巫醫指定兩個尚未被打敗的參賽者去打架，並淘汰掉打輸的參賽者。

當兩名參賽者打架，擁有比較多巫力的參賽者獲勝（如果兩個參賽者巫力一樣，編號比較小的參賽者獲勝）。令 x, y 分別是勝者及敗者的編號，在打架完畢後 x 的巫力會被 y 給消耗。更精確的來說 a_x 會變成 $a_x - a_y$ 。

經過 $N - 1$ 場打架之後，沒有輸過的參賽者會成為新一代巫醫巫醫，但是因為經過很多消耗，這位參賽者的巫力可能大不如前，現在巫醫巫醫希望新一代巫醫巫醫能夠擁有最大的巫力，請你告訴巫醫巫醫怎麼樣安排比賽最好。

Input

輸入的第一行有一個正整數 N 。

接下來一行有 N 個正整數 a_i 以一個空白隔開，代表參賽者們的巫力。

- $1 < N \leq 100$
- $1 \leq a_i \leq 10^5$

Output

第一行請輸出一個整數代表新一代巫醫巫醫的最後所剩下的巫力。接下來 $N - 1$ 行，在第 i 行請輸出兩個整數以空白隔開，代表參加第 i 場架的兩位參賽者在打該場架前所剩下的巫力。

如果有多組解，任意一組都會被視為正確。

Sample Input 1	Sample Output 1
4 10 14 20 33	29 14 20 6 10 4 33

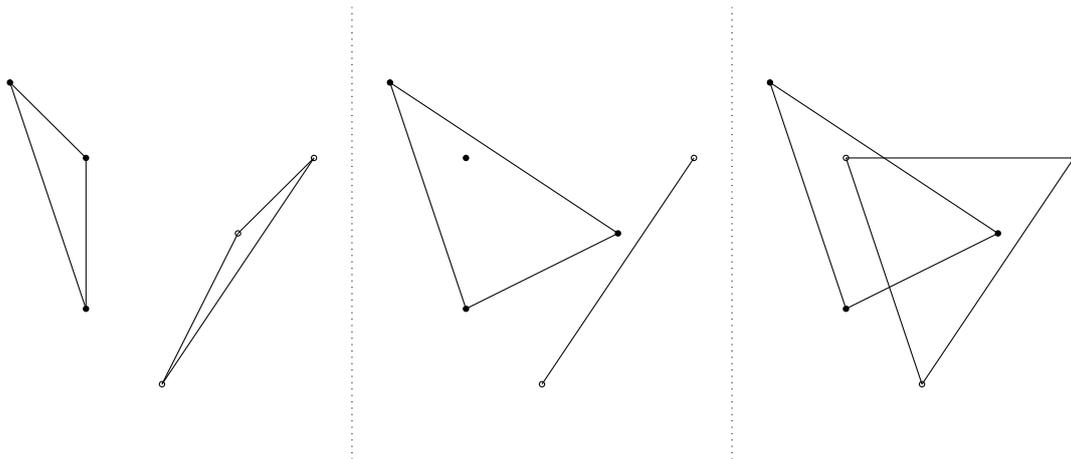
C. 巴乙己分組問題

Problem ID: grouping

巴乙己一到店，所有選手便對著他膜拜，有的叫道，「巴乙己，您 CF 又增上分數了！」他不回答，對櫃裏說，「備兩場 OI，要一單 Div. 1。」便排出九文大錢。他們又故意的高聲嚷道，「您一定又電爛人家的自尊了！」巴乙己睜大眼睛說，「你怎麼這樣憑空污人清白……」「什麼清白？我前天親眼見您海放 tourist 的題數，吊着打。」巴乙己便漲紅了臉，額上的青筋條條綻出，爭辯道，「做題不能算電……做題！……競程人的事，能算電麼？」接連便是難懂的話：

「假設現在二維平面上給定 N 個點，你們難道總是能把這些點分成非空的兩組，使得不存在一個實數點同時落在兩組點的兩個凸包內，還沒有任何給定的點嚴格落在任何一個凸包內部不成！？」

眾人一聽，愣了，百思不得其解。動輒上萬人給出了下列圖片做為示意圖，滿足左邊第一張圖是一個合法的分組方式，右邊的兩張圖則都不是合法的分組方式：



作為巴乙己粉絲的你，能給出這道問題的答案嗎？不過為了簡化問題，你只需要先解決「保證任三點不共線」的情況就可以了。

凸包

一組二維點的「凸包」是一個最小的簡單凸多邊形，滿足多邊形內任意兩點的連線皆不會經過多邊形外部，且所有給定的二維點也都不落在該多邊形外部。而在此題我們特別定義兩點形成的連線、單一點也可以被當作凸包。

Input

輸入第一行有一個正整數 N ，代表二維平面上點的數量。

接著 N 行，第 i 行包含兩個整數 x_i, y_i ，代表第 i 個點的座標為 (x_i, y_i) 。

- $2 \leq N \leq 2000$
- $-10^9 \leq x_i, y_i \leq 10^9$
- 不會有兩個點的座標相同
- 保證任三點不共線

Output

若不存在任何合法的分組，輸出 No 於一行。否則，輸出 Yes 於首行，次行輸出一個長度為 N 的 01 字串，第 i 個字元代表第 i 個點分到的組別，也就是所有字元為 0 的分在一組、字元為 1 的分在另一組。

任何一組滿足以下所有條件的輸出皆會獲得 Accepted，否則會獲得 Wrong Answer：

- 至少存在一個 0。
- 至少存在一個 1。
- 不存在一個實數點同時落在你分類好兩組點的兩個凸包內，且沒有任何給定的點嚴格落在任何一個凸包內部。

Sample Input 1

6	Yes
0 4	000111
1 3	
1 1	
3 2	
2 0	
4 3	

Sample Output 1

Sample Input 2	Sample Output 2
9 0 1 2 2 1 7 3 0 2 4 5 3 9 4 5 5 4 5	No

Sample Input 3	Sample Output 3
10 1 4 3 7 2 2 3 3 6 5 8 2 2 5 4 3 4 0 5 1	Yes 1011001010

Sample Input 4	Sample Output 4
3 0 0 0 1 1 0	Yes 011

This page is intentionally left blank.

D. 教授分球球

Problem ID: professor

你在比完了全國爬山大賽 (National Pa Shan Contest, 簡稱 NPSC) 之後, 準備開心地返家, 但途中卻被多洛杜教授綁架了。你在昏迷一段時間後終於恢復意識, 映入眼簾的是多洛杜教授與一堆球。

多洛杜教授表示桌上總共有 N 顆球排成一列, 其中每顆球不是紅色就是白色, 他要求你算出有多少組正整數數對 (l, r) 滿足 $1 \leq l \leq r \leq N$ 且從左數來第 l 到第 r 球中, 紅色球的個數要和不在這範圍內的紅色球個數相同, 白色球的個數也要和不在這範圍內的白色球個數相同。

多洛杜教授宣稱你只要回答正確, 他就會放你離開, 同時他也告訴你, 他之所以綁架你的原因, 是因為他想用行動讓你體會到去比 NPSC 的人什麼都不會, 只會爬山。

經過一段時間的觀察, 你仍然沒有任何頭緒, 因此多洛杜教授決定大發慈悲的告訴你一個提示。他告訴你 M 個正整數 a_1, a_2, \dots, a_M , 這些數字所代表的意義為從左數來會先有連續 a_1 顆紅球, 再有 a_2 顆白球, 再有 a_3 顆紅球, 再有 a_4 顆白球, 以此類推, 而且 $N = a_1 + a_2 + \dots + a_M$ 。

他相信即便在獲得這個關鍵提示之後, 你仍然會做不出來。請利用這個提示, 正確回答多洛杜教授的問題, 讓多洛杜教授知道比 NPSC 的人並不是只會爬山, 還會解題。

Input

輸入的第一行包含一個正整數 M 。

接著第二行包含 M 個正整數 a_1, a_2, \dots, a_M 。

- $2 \leq M \leq 10^6$
- $1 \leq a_i \leq 10^9$

Output

輸出一行, 這行只有一個整數, 代表多洛杜教授問題的答案。

Hint

範例測資一中，所有滿足條件的 (l, r) 為 $(1, 5), (4, 8), (5, 9), (6, 10)$ 。

Sample Input 1

6 1 2 1 3 2 1	Sample Output 1 4
------------------	----------------------

Sample Input 2

5 4 8 7 6 3	Sample Output 2 7
----------------	----------------------

Sample Input 3

4 10 10 10 10	Sample Output 3 21
------------------	-----------------------

Sample Input 4

9 8 6 4 1 9 7 5 3 2	Sample Output 4 0
------------------------	----------------------

E. 真島與莉可麗絲

Problem ID: lycoris

NPSC 國在表面上看似很和平，但是這其實只是假象，事實上有一個專門無聲無息的將恐怖份子擺平的神祕機構，而機構內負責行動的那群人被統稱為莉可麗絲 (Lycoris)，而一直被莉可麗絲破壞恐攻計畫的恐怖份子首腦——真島，這次打算和他們一決死戰。

NPSC 國境內有 N 座城市，編號為 $1, 2, \dots, N$ ，總共恰有 $N - 1$ 條從城市到城市的雙向高速公路，而且任兩座城市之間恰有一條經過若干條高速公路的路徑連接彼此。

真島知道每座城市都有各自的莉可麗絲分部，因此真島大決戰計畫的第一部分，就是要炸毀每條高速公路，讓每個分部都不能互相支援，這部分真島可以用特製的炸彈完成。對於任兩座還互相連通的城市 X, Y ，一顆特製炸彈能同時炸毀連接城市 X, Y 的路徑中的所有高速公路。注意到一條高速公路被炸毀等同於這條高速公路未來不存在，因此原本能透過若干條高速公路連接的兩座城市有可能會變得不再連通。這些特製的炸彈還有一個大缺點，就是它們不能同時引爆，不然會有連真島都不能預測的後果。

理論上高速公路用特製炸彈隨便炸一炸就可以了，但是真島希望能堅持自己的原則，他的原則是要讓每個炸彈都恰好炸毀**質數條**高速公路。

你是真島的副手，因此他很理所當然的要你幫他想好一種使用炸彈的方法滿足他的原則，或是告訴他如果要秉持自己原則，則一定炸不完所有的高速公路。

連通

兩座城市互相連通，代表可以從其中一座城市經過若干個**未被炸毀**的高速公路到達另一座城市。

質數

一個正整數 P 是質數若且唯若 $P > 1$ 且 P 只能被 1 和自己整除。

Input

輸入的第一行包含一個正整數 T ，代表接下來測試資料的筆數。

每一筆測試資料會有數行輸入。

其中第一行輸入一個正整數 N ，代表有 N 座城市。

接下來輸入 $N - 1$ 行，其中第 i 行輸入兩個正整數 u_i, v_i ，代表有一條連接城市 u_i 和城市 v_i 的高速公路。

- $1 \leq T \leq 100$
- $2 \leq N \leq 2 \times 10^5$
- $1 \leq u_i, v_i \leq N$
- $u_i \neq v_i$
- 對於任兩個 $i, j (1 \leq i < j \leq N)$ 保證 $\{u_i, v_i\} \neq \{u_j, v_j\}$
- 保證輸入的 $N - 1$ 條高速公路滿足題目敘述
- 保證這 T 筆測試資料中， N 的總和不超過 2×10^5

Output

對於每一筆測試資料，如果不存在滿足真島的原則的炸彈用法，則輸出一行，這行輸出 -1 即可。否則第一行輸出一個正整數 $M (1 \leq M \leq N - 1)$ ，代表要使用 M 顆炸彈。

接下來輸出 M 行，第 i 行輸出兩個正整數 $a_i, b_i (1 \leq a_i, b_i \leq N, a_i \neq b_i)$ ，代表第 i 顆炸彈要炸毀城市 a_i 到城市 b_i 路徑上的所有高速公路。記得每次炸毀的高速公路個數必須是質數。

如果存在多種可行的炸彈使用方案，輸出任意一種即可。注意到你**不需要最小化**炸彈的使用量。

Sample Input 1	Sample Output 1
4	-1
2	2
1 2	1 3
5	3 5
1 2	-1
2 3	3
3 4	3 6
4 5	8 12
4	1 10
1 2	
1 3	
1 4	
12	
1 2	
2 3	
3 4	
4 5	
5 6	
3 7	
7 8	
7 9	
9 10	
7 11	
11 12	

This page is intentionally left blank.

F. 城市規劃

Problem ID: grid

NPSC 國打算興建一座名為方格市的新城市。身為方格市第一任市長的方格王，規劃新城市的任務自然地落到了他的手上。

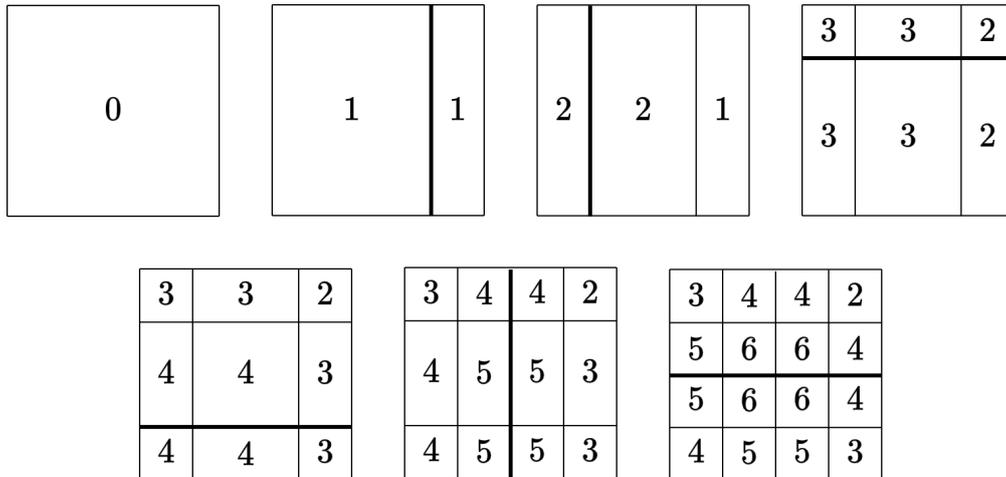
在方格王的構想當中，這座城市最後的樣貌應該正如其名，是由排成 N 個東西向橫列與 M 個南北向直行的 $N \times M$ 個方格組成。但是方格王認為直接建造 $N \times M$ 個一模一樣的方格太無聊了，他決定用有趣一點的方式建造：一開始城市是一個大方格，這個方格的「有趣度」是 0，且城市中沒有任何道路。接下來，他要建造 $N + M - 2$ 條道路，建造道路的方法分為以下兩種：

1. 選擇一橫列的方格，建造一條東西向道路穿越這一系列的所有方格，讓其中的每個方格分為上下兩個方格。如果一個方格本來的有趣度是 x ，那麼分隔出的兩個方格有趣度則是 $x + 1$ 。
2. 選擇一直行的方格，建造一條南北向道路穿越這一行中的所有方格，讓其中的每個方格分為左右兩個方格。如果一個方格本來的有趣度是 x ，那麼分隔出的兩個方格有趣度則是 $x + 1$ 。

顯而易見地，方格王會使用第一種方法恰好 $N - 1$ 次，以及使用第二種方法恰好 $M - 1$ 次。

方格王開心地規劃了建造道路的方法和順序，並把結果的 $N \times M$ 個方格有趣度記錄下來，然後就去睡覺了。隔天早上醒來，他完全不記得建造道路的順序應該要是什麼。這件事讓負責建造城市的鯉魚王非常困擾，為了避免方格王生氣，請你幫鯉魚王找出一種建造道路的方式，使得得出的所有方格的有趣度都和方格王規劃的一樣。

舉例來說，下圖是範例輸出一描述的建造過程中，方格市與其中每個方格的有趣度變化，粗線是新建造的道路：



Input

輸入的第一行有兩個整數 N, M ，表示方格王規劃的方格市有 N 列 M 行。

接下來有 N 行，其中第 i 行包含 M 個非負整數 $a_{i,1}, a_{i,2}, \dots, a_{i,M}$ ， $a_{i,j}$ 表示第 i 列第 j 行的方格的有趣度。

- $2 \leq N \times M \leq 10^6$
- $0 \leq a_{i,j} \leq 2 \times 10^6$
- 保證方格王不會出錯，一定存在至少一種滿足鯉魚王需求的建造道路的方式

Output

輸出 $N + M - 2$ 行，其中第 i 行表示第 i 條要建造的道路是哪一條。一條道路以 $R \ x$ (x 是 $[1, N - 1]$ 內的整數) 或 $C \ y$ (y 是 $[1, M - 1]$ 內的整數) 表示， $R \ x$ 表示最後規劃中第 x 列和第 $x+1$ 列方格之間的東西向道路、 $C \ y$ 表示最後規劃中第 y 行和第 $y+1$ 行方格之間的南北向道路。

Sample Input 1	Sample Output 1
4 4	C 3
3 4 4 2	C 1
5 6 6 4	R 1
5 6 6 4	R 3
4 5 5 3	C 2
	R 2

Sample Input 2	Sample Output 2
1 8	C 1
1 4 4 3 4 5 5 3	C 4
	C 3
	C 2
	C 7
	C 5
	C 6

Sample Input 3	Sample Output 3
4 3	R 2
3 4 4	C 1
3 4 4	R 1
3 4 4	C 2
3 4 4	R 3

This page is intentionally left blank.

G. 魔族與訓練

Problem ID: mazoku

在經過幾場的挑戰後，魔法少女小桃認為魔族夏美子太弱了，沒有好好運用魔族的力量來戰鬥，因此她決定要來訓練夏美子。透過各種戰鬥，能夠讓魔族更有效率的利用魔力，使她的戰鬥能力變強。

魔族的魔力值可以用一個非負整數 x 來表示，由於夏美子是個新手魔族，她的魔力值有一個正整數 M 作為上限，即代表她的魔力值超過 M 的部分會直接消失。

為了有效率的訓練夏美子，小桃準備了 N 個關卡，其中第 i 個關卡可用一個整數 a_i 作為它的難易度。根據難易度的不同，可將關卡分為以下兩類：

- 若 $a_i > 0$ ，則代表這個關卡有一隻強度為 a_i 的怪物，夏美子的魔力值需至少為 a_i 才能完成此關卡，且戰鬥完後夏美子的魔力值會減少 a_i 。
- 若 $a_i \leq 0$ ，則代表這個關卡是一個休息關卡，夏美子的魔力值會增加 $|a_i|$ ，然而若增加後的魔力值超過上限 M ，超過的部分還是會消失。

一場訓練會包含許多活動，活動一共有兩種類型：

- 為了讓訓練的效果更好，小桃將第 p 個關卡的難易度調整為 x 。這種類型叫做**調整**。
- 在夏美子魔力值為 y 的時候，小桃會告訴夏美子一個關卡區間 $[l, r]$ ，代表她需要依序完成第 $l, l+1, \dots, r$ 個關卡。這種類型叫做**闖關**。

然而小桃發現在闖關的過程中，夏美子的魔力值可能會不夠導致無法完成。為了避免這種情況發生，她為夏美子準備了許多可以隨時恢復魔力值的藥丸，每吃 1 顆藥丸她的魔力值就會增加 1，增加後的魔力值超過上限 M 的部分依然會消失。

做為小桃的助手，小桃告訴你今天的訓練一共會有 Q 個活動。然而因為藥丸價格不便宜，對於每個闖關活動，小桃想要知道至少需要幾個藥丸才能保證夏美子能夠完成闖關。同時為了事先了解每個闖關活動的成效，她也想知道在使用最少藥丸的情況下，完成闖關後的夏美子魔力值是多少。請幫助小桃解決這個問題。

Input

首先輸入第一行包含三個正整數 N, Q, M ，代表小桃準備的關卡數量、今天訓練的活動數量以及夏美子的魔力值上限。

接著輸入第二行包含 N 個整數 a_1, a_2, \dots, a_N ，代表第 i 個關卡的難易度。

再來第三行到第 $Q + 2$ 行，每一行會代表一個訓練的活動。此行首先會有一個整數 t ，若 $t = 1$ 則代表此活動的類型為調整，且接下來會有兩個整數 p, x 。若 $t = 2$ 則代表此活動的類型為闖關，且接下來會有三個整數 l, r, y 。變數 p, l, r, x, y 的定義與題目敘述相同。

- $1 \leq N, Q \leq 3 \times 10^5$
- $1 \leq M \leq 10^9$
- $-M \leq a_i \leq M$
- $t \in \{1, 2\}$
- $1 \leq p \leq N$
- $-M \leq x \leq M$
- $1 \leq l \leq r \leq N$
- $0 \leq y \leq M$

Output

對於每一個闖關活動，輸出一行，該行包含兩個整數，第一個為最少所需準備的藥丸數量，第二個為在使用最少藥丸的情況下，完成闖關後夏美子的魔力值。

Notes

在範例測資一中，以下為第 2 個闖關活動的其中一個最佳策略：

1. 完成第 1 個關卡，魔力值變成 10。
2. 完成第 2 個關卡，魔力值變成 2。
3. 吃下 2 顆藥丸，魔力值變成 4。
4. 完成第 3 個關卡，魔力值變成 4。
5. 完成第 4 個關卡，魔力值變成 0。

因此答案為 2 0。

Sample Input 1

```
5 6 10
-4 8 0 4 -2
2 3 5 3
2 1 4 8
1 3 -1
2 3 5 3
2 1 4 8
2 1 5 0
```

Sample Output 1

```
1 2
2 0
0 2
1 0
7 2
```

Sample Input 2

```
5 1 1000000000
1000000000 -123456789 987654321 1000000000 -487638763
2 1 5 0
```

Sample Output 2

```
2864197532 487638763
```

Sample Input 3

```
12 12 12
0 9 -2 8 0 -3 2 5 -1 -1 0 -3
2 6 12 4
2 1 1 9
2 8 11 0
2 4 8 3
1 1 -2
2 5 6 10
1 4 1
2 4 7 2
2 8 9 1
1 8 9
1 2 -4
2 1 12 0
```

Sample Output 3

```
0 5
0 9
5 2
9 0
0 12
0 2
4 1
1 5
```

H. 買餅乾

Problem ID: cookie

小董跟小薺是感情很好的一對姐妹，為了慶祝她們的四歲生日，她們決定到小橘開設的「黑色薔薇」烘焙坊購買一些手工餅乾。手工餅乾每 M 片的售價為 72.7 元。在結帳時，若袋中的餅乾數量並非 M 的倍數，則小橘會視情況放入或取出一些餅乾，讓餅乾數量變為 M 的倍數。

具體來說，小橘會設定一個整數 K ，假設某袋餅乾一共有 x 塊，而 x 除以 M 的餘數為 r ，若 $r \leq K$ ，則小橘會取出 r 塊餅乾；若 $r > K$ ，則小橘會放入 $M - r$ 塊餅乾，使得該袋餅乾的數量變為 M 的倍數。

身為姐姐的小董，因為她沒有朋友，她打算把所有要買的餅乾放入同一個袋子中；而妹妹小薺為了與朋友們分享餅乾，她打算把所有要買的餅乾分裝在 N 個袋子，並且每個袋子裡至少要有 1 塊餅乾。

在結帳前，聰明的小薺發現，即使兩人拿的餅乾總數相同，結帳時小橘調整餅乾數量的方式也可能讓最後得到的餅乾數量有很大的差異！可愛的小董想請你幫她們算算，若兩人拿的餅乾總數相同，在怎樣的分裝方式下結帳後小董得到的餅乾數量會比小薺多最多，而又是在怎樣的分裝方式下結帳後小薺得到的餅乾數量會比小董多最多呢？

以範例測資一為例，若小薺的三袋餅乾數量分別為 9, 10, 13 塊，經過小橘調整後三袋餅乾的數量分別將會變成 7, 7, 14 塊，一共有 28 塊；而小董的袋子中則裝了 $9 + 10 + 13 = 32$ 塊餅乾，經過小橘調整後將變為 35 塊，小董的餅乾數量比小薺多了 $35 - 28 = 7$ 塊。可以證明無論小薺怎麼拿餅乾，結帳後小董的餅乾數量不可能比小薺多出 8 塊或更多。

若小薺的三袋餅乾數量分別為 5, 11, 22 塊，經過小橘調整後三袋餅乾的數量分別將會變成 7, 14, 21 塊，一共有 42 塊；而小董的袋子中則裝了 $5 + 11 + 22 = 38$ 塊餅乾，經過小橘調整後將變為 35 塊，小薺的餅乾數量比小董多了 $42 - 35 = 7$ 塊。可以證明無論小薺怎麼拿餅乾，結帳後小薺的餅乾數量不可能比小董多出 8 塊或更多。

Input

輸入僅一行，包含三個整數 N, M, K ，分別以一個空白隔開。

- $2 \leq N \leq 10^5$
- $1 \leq M \leq 10^9$
- $0 \leq K < M$

Output

輸出兩行，每行包含 N 個正整數，第 i 個整數表示小齊的第 i 個袋子中的餅乾數量。第一行請輸出結帳後小董的餅乾數量比小齊多最多的分裝方式；第二行請輸出結帳後小齊的餅乾數量比小董多最多的分裝方式。

由於小橘只烤了 10^{18} 塊餅乾，每種分裝方式的餅乾數量總和不能超過 10^{18} 。如果有多種分裝方式，你可以輸出任意一種。

請注意，同一行兩個整數間請以一個空白隔開，並且每一行最後一個整數後不得有空白。

Sample Input 1	Sample Output 1
3 7 3	9 10 13 5 11 22
Sample Input 2	Sample Output 2
8 9 6	14 14 15 24 24 31 41 42 53 61 70 70 88 89 89 89