

香港電腦奧林匹克競賽 2025/26 高級組

題目總覽

編號	名稱	執行時間限制	記憶體限制	子任務
S261	部落佔領	1.000 s	256 MB	11 + 12 + 17 + 22 + 23 + 15
S262	節奏均衡	2.000 s	1024 MB	4 + 8 + 17 + 18 + 11 + 15 + 13 + 14
S263	神奇形色牌	1.000 s	256 MB	7 + 12 + 16 + 21 + 12 + 32
S264	三角領土	1.000 s	256 MB	6 + 7 + 18 + 11 + 15 + 24 + 12 + 7

注意:

除非特別注明，否則輸入輸出將依照以下格式:

- 同一行中，數字與數字或字元之間需有一個空格。
- 同一行中，字元與字元之間並無空格。
- 每個字串需放在獨立的行。
- 輸出將自動被修正如下：每行最尾的連續的空格會被刪除，及在輸出最後補上換行符(如沒有)。其他格式問題則不會修正。

C++ 使用者請注意 `cin` / `cout` 可能導致輸入輸出樽頸使程式執行變慢。

有些題目可能需要使用 64 位元整數。在 C++ 中它是 `long long` 而其 `scanf` / `printf` 代號是 `%lld`。

所有題目均有細分多個子任務，你需要通過該子任務中的所有測試數據才能得到分數。

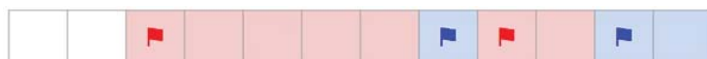
S261 - 部落佔領

執行時間限制: 1.000 s / 記憶體限制: 256 MB

紅、藍兩個部落正在一條長線上插旗以佔領領土。

這條線是一個有 M 個單元格的網格，從左到右以 1 至 M 編號。共有 N 面旗幟。第 i 面旗幟有一個顏色 C_i （紅或藍）和一個位置 P_i 。

放置所有旗幟後，部落開始按以下方式佔領網格單元格。對於每面旗幟，該旗幟的擁有者，紅或藍部落其一，開始佔領從旗幟所在單元格往右的領土，包括旗幟所在的單元格。一面旗幟的領土在下一面旗幟的前一格停止，而最後一面旗幟的領土則延伸到單元格 M 。一個部落的 總領土 是該部落所佔領的單元格總數。



你希望幫助紅部落獲得最多的領土。為此，你允許移動 **最多一面** 紅旗，方法是：選擇網格上現有的一面紅旗，將其移除，然後將該旗幟放置在 **當前為空** 的任何單元格（從 1 到 M ）中。你也可以選擇不移動任何旗幟。

在你做出最佳移動後，紅部落可能獲得的最大總領土是多少？

輸入

輸入的第一行包含兩個整數 N 和 M ，分別是旗幟的數量和網格的大小。

接下來的 N 行，每行包含一個字符 C_i 和一個整數 P_i ，分別是旗幟的顏色和位置。 $C_i = \text{R}$ 表示旗幟是紅色的， $C_i = \text{B}$ 表示旗幟是藍色的。

保證網格中的每個單元格最多包含一面旗幟，並且旗幟按位置的升序排列。保證至少有一面旗幟是紅色的。

輸出

輸出一個整數：紅部落可能獲得的最大總領土。

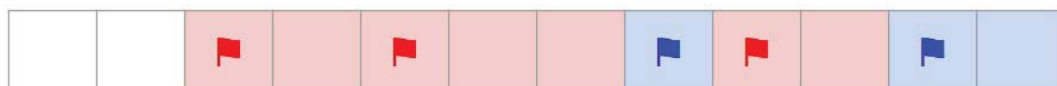
樣例

輸入

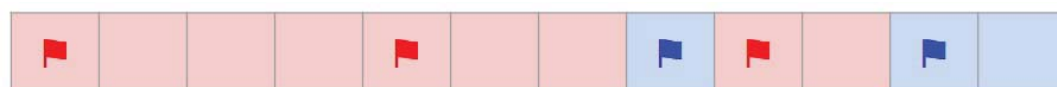
輸出

1	5 12 R 3 R 5 B 8 R 9 B 11	9
----------	--	---

兩個部落最初佔領的網格單元格如下圖所示。紅部落的總領土為 7。



最佳策略是將位於位置 3 的紅旗移動到位置 1。紅部落的總領土變為 9。



輸入

輸出

2

4 9 B 3 R 5 B 8 R 9	5
---------------------------------	---

3

1 10 R 6	10
-------------	----

你可以將該面紅旗移動到位置 1。

子任務

對於所有數據：
 $1 \leq N \leq 10^5$
 $1 \leq M \leq 10^9$
對於 $1 \leq i \leq N$ ， C_i 為 R 或 B
 $1 \leq P_1 < P_2 < \cdots < P_N \leq M$
至少有一面旗幟是紅色的。

佔分

約束條件

1	11	$N = 2$ $C_1 = \text{R}$ $C_2 = \text{B}$
2	12	$N \geq 2$ $C_1 = \text{R}$ $C_2 = \cdots = C_N = \text{B}$
3	17	$N \geq 2$ $C_1 = \text{R}$ $C_2 = \text{R}$
4	22	$N \leq 100$ $M \leq 500$
5	23	$N \leq 2000$
6	15	無額外約束

S262 - 節奏均衡

執行時間限制: 2.000 s / 記憶體限制: 1024 MB

鮑伯的樂隊正在創作一首新的複節奏作品，而他需要設計相應的鼓點節奏。

這首作品由 N 個小節組成，第 i 個小節的時長為 A_i 毫秒。

為了保持穩定的節拍，鮑伯要求在任何單一小節內的鼓點必須在時間上等距。對於每個小節 i ，他必須選擇一個間隔 X_i ，使得 X_i 是時長 A_i 的一個 **正因數**。因此，在第 i 個小節中演奏的音符數量恰好為 $\frac{A_i}{X_i}$ 。

然而，為了避免演出時過度疲勞，鮑伯希望整首作品最多演奏 $K \geq N$ 個音符。這意味着所選的間隔必須滿足條件 $\frac{A_1}{X_1} + \frac{A_2}{X_2} + \dots + \frac{A_N}{X_N} \leq K$ 。

最後，為了讓作品聽起來更具凝聚力，不同小節之間的間隔不該有太大的變化。鮑伯將一種選擇間隔的方法 $X_1, X_2, \dots, X_N$ 的 **不穩定度** 定義為所有小節所用間隔中的最大值與最小值之差，即 $\max(X_i) - \min(X_i)$ 。鮑伯希望在滿足以上條件的前提下，將不穩定度最小化。

鮑伯是位很棒的鼓手，但他不太擅長編寫程式。你能幫他設計這首作品嗎？若有多種鼓點間隔能達到最小的不穩定度，你可以輸出任意一種。注意，在本題的限制條件下，總是存在至少一種符合條件的鼓點間隔。

輸入

第一行包含兩個整數 N 和 K ，代表小節數量與音符數量上限。

第二行包含 N 個整數 $A_1, A_2, \dots, A_N$ ，表示每個小節的時長，以毫秒作為單位。

輸出

第一行輸出一個整數，表示你能達到的最小不穩定度。

第二行輸出 N 個整數 $X_1, X_2, \dots, X_N$ ，表示所設計作品的間隔。

若有多種鼓點間隔能達到最小的不穩定度，你可以輸出任意一種。

樣例

輸入

輸出

1

3 10	2
10 9 12	5 3 4

音符總數為 $\frac{10}{5} + \frac{9}{3} + \frac{12}{4} = 8 \leq 10$ ，而不穩定度為 $5 - 3 = 2$ 。

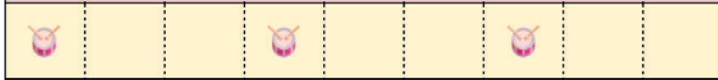
Bar 1

第一小節



Bar 2

第二小節



Bar 3

第三小節



除此之外， $X = \{5, 3, 3\}$ 也是一個有效的答案。

輸入 輸出

2	2 3	0
	1 2	1 1

音符總數為 $\frac{1}{1} + \frac{2}{1} = 3 \leq 3$ ，而不穩定度為 $1 - 1 = 0$ 。

3	6 20	10
	2 3 5 7 11 13	2 3 5 7 11 1

4	6 20	6
	1 3 5 7 9 11	1 3 5 7 3 1

子任務

對於所有數據：

$$2 \leq N \leq 2 \times 10^5$$

$$1 \leq K \leq 10^{11}$$

$$1 \leq A_i \leq 5 \times 10^5$$

$$N \leq K \leq A_1 + A_2 + \cdots + A_N$$

佔分 約束條件

1	4	$N = 2$ $A_1 = 1$ $A_2 = 2$
2	8	$N = 2$ $A_1 = 1$
3	17	$N \leq 2000$ 對於 $1 \leq i \leq N$ ， $A_i \leq 2000$ $A_1 \leq A_2 \leq \cdots \leq A_N$ 對於 $1 \leq i \leq N$ ， A_i 為質數
4	18	$N \leq 100$ 對於 $1 \leq i \leq N$ ， $A_i \leq 100$
5	11	$N \leq 2000$ 對於 $1 \leq i \leq N$ ， $A_i \leq 2000$
6	15	$N \leq 5 \times 10^4$ 對於 $1 \leq i \leq N$ ， $A_i \leq 10^5$ $A_1 = 1$
7	13	$N \leq 5 \times 10^4$ 對於 $1 \leq i \leq N$ ， $A_i \leq 10^5$
8	14	無額外約束

S263 - 神奇形色牌

執行時間限制: 1.000 s / 記憶體限制: 256 MB

你有沒有玩過 神奇形色牌 這款卡牌遊戲？在 神奇形色牌 中，一張牌由 L 種不同 特徵 所組成，例如紋路、圖案數量等，其中每種特徵有 三種可能性。舉例來說，紋路的可能性可以是空心、條紋或實心，而每張牌上可以有一個、兩個或三個圖案。

具體來說，我們將特徵以 1 到 L 作編號，並將一個特徵之可能性標記為 $\boxed{A}$ 、 $\boxed{B}$ 及 $\boxed{C}$ 。每張牌以一個長度為 L ，並由 $\boxed{A}$ 、 $\boxed{B}$ 及 $\boxed{C}$ 組成的字串所表示，其中第 j 個 ($1 \leq j \leq L$) 字符表示牌的第 j 個特徵。

一副 完整牌組 由 3^L 張牌所組成，其中有着獨一無二特徵組合的每張牌剛好出現一次。例如一副 $L = 2$ 完整牌組有 $3^2 = 9$ 張牌，分別可以由 $\boxed{AA}$ 、 $\boxed{AB}$ 、 $\boxed{AC}$ 、 $\boxed{BA}$ 、 $\boxed{BB}$ 、 $\boxed{BC}$ 、 $\boxed{CA}$ 、 $\boxed{CB}$ 及 $\boxed{CC}$ 所表示。

一場 神奇形色牌 遊戲中，一些牌被放在桌上，然後玩家爭先尋找 有效 的三牌集合。一個三牌集合為 有效 當且僅當對於每個特徵，三張牌或是完全相同，或是完全不同。舉例來說，

- 牌集合 $\boxed{AB}$ 、 $\boxed{BB}$ 及 $\boxed{CB}$ 是有效的，因為他們的特徵 1 完全不同，且特徵 2 完全相同。
- 牌集合 $\boxed{AA}$ 、 $\boxed{AB}$ 及 $\boxed{BC}$ 是無效的，因為他們的特徵 1 既非完全相同，亦非完全不同。
- 三張相同牌組成的牌集合總是有效的，因為他們的 L 個特徵都完全相同。

鮑伯十分喜歡玩 神奇形色牌，但由於他太過厲害，沒有人想跟他對戰！所以，他決定發明 神奇形色牌 的單人模式，玩法如下：

- 首先，抽取 N 張牌（其中 N 是 3 的倍數）並將它們放在桌上。抽取的牌中可能有相同的牌。
- 然後，可以打開 任意數量（可能為零）的完整牌組，並將完整牌組內的所有牌放到桌上。
- 接着，可以不停選擇組成 有效 三牌集合的任意三張牌，並將它們從桌上移除。
- 當桌上的所有牌都被移除時，即為獲勝。

鮑伯剛剛勝出一場單人模式的 神奇形色牌 —— 並決定要挑戰你！給定鮑伯抽取的 N 張起始牌，你能否勝出遊戲？由於你需要在完結後清理桌面，你並不想打開太多完整牌組（請參照輸出部分）。

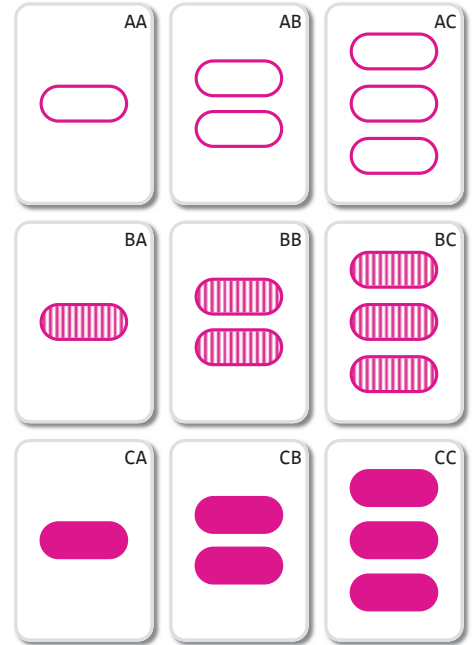


圖 1：一副 $L = 2$ 完整牌組。

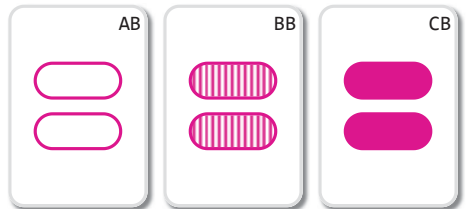


圖 2a：一個有效的三牌集合。

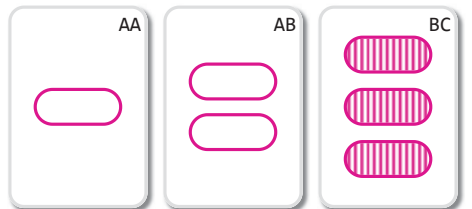


圖 2b：一個無效的三牌集合。

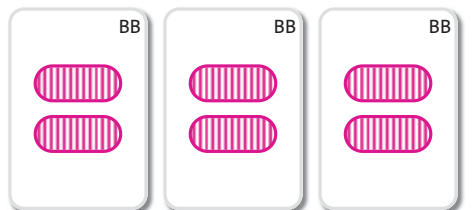


圖 2c：三張相同牌組成的集合總是有效的。

注意

由於鮑伯不清楚是否所有人都從 HKOI 初賽中得知一個相關（**可能有用，可能無用**）的事實，出於公平性考慮，他讓你知道如果我們將一個特徵的三種可能性 $\boxed{A}$ 、 $\boxed{B}$ 及 $\boxed{C}$ 表示成 1、2 及 3，那麼三張牌在該特徵中屬完全相同或完全不同，當且僅當總和被 3 整除。

這是由於特徵完全相同或完全不同的情況為 $\boxed{AAA}$ 、 $\boxed{BBB}$ 、 $\boxed{CCC}$ 、 $\boxed{ABC}$ 、 $\boxed{ACB}$ 、 $\boxed{BAC}$ 、 $\boxed{BCA}$ 、 $\boxed{CAB}$ 及 $\boxed{CBA}$ ，剛好覆蓋所有總和被 3 整除的情況。例如考慮情況 $\boxed{CCC}$ ，對應的總和為 $3 + 3 + 3 = 9$ ，被 3 所整除。

輸入

第一行包含兩個整數 N 和 L ，分別為牌的數量和每張牌上特徵的數量。

接下來的 N 行各包含一個長度為 L 、只由 $\boxed{A}$ 、 $\boxed{B}$ 及 $\boxed{C}$ 所組成的字串 S_i ($1 \leq i \leq N$)，描述第 i 張牌。其中 S_i 的第 j 個 ($1 \leq j \leq L$) 字符表示第 i 張牌的第 j 個特徵，而 $\boxed{A}$ 、 $\boxed{B}$ 及 $\boxed{C}$ 表示該特徵的三種可能性。

輸出

在第一行，輸出兩個非負整數 M 和 K ，分別為你使用的額外完整牌組數量和你所組成的集合數量。

接下來的 K 行，各輸出一個有效的三牌集合。每張牌應該以一個長度為 L 、只由 $\boxed{A}$ 、 $\boxed{B}$ 及 $\boxed{C}$ 所組成的字串表示。每張牌必須使用剛好一次。

留意 K 不能超過 2×10^5 。換句話說，牌的總數量 $N + M \times 3^L = 3K$ 不能超過 6×10^5 。

如果有數種方式贏得遊戲，你可以輸出任意一個。請留意只要 $K \leq 2 \times 10^5$ 即可，毋須使 K 最小。

保證 **給定的測試數據中**，遊戲皆可被獲勝。可以證明在題目的約束下，如果一場遊戲可被獲勝，存在一種方式獲勝使得 K 不超過 2×10^5 。

樣例

	輸入	輸出
1	6 2 AB AB AC AC BB CC	1 5 AA BB CC AB AB AB AC AC AC BA BB BC CA CB CC

我們可以打開一副完整牌組，其由 $3^2 = 9$ 張牌所組成，並分別以 $\boxed{AA}$ 、 $\boxed{AB}$ 、 $\boxed{AC}$ 、 $\boxed{BA}$ 、 $\boxed{BB}$ 、 $\boxed{BC}$ 、 $\boxed{CA}$ 、 $\boxed{CB}$ 和 $\boxed{CC}$ 表示。然後，我們可以將 $6 + 9 = 15$ 張牌分成 5 個有效三牌集合。

	輸入	輸出
2	6 1 A B A C A A	0 2 A A A A B C
3	6 1 A B A C A A	1 3 A A A A B C A B C

毋須打開最小數量的完整牌組。

子任務

對於所有數據：
 $3 \leq N \leq 60000$
 N 是 3 的倍數
 $1 \leq L \leq 9$
在所有測試數據中，遊戲皆可被獲勝

	佔分	約束條件
1	7	$N = 3^L$ 起始卡牌組成一副完整牌組
2	12	$L = 1$
3	16	$L = 2$ 剛好有 1 張牌以 B 作為開頭 剛好有 1 張牌以 C 作為開頭
4	21	$L = 2$
5	12	$L \leq 4$
6	32	無額外約束

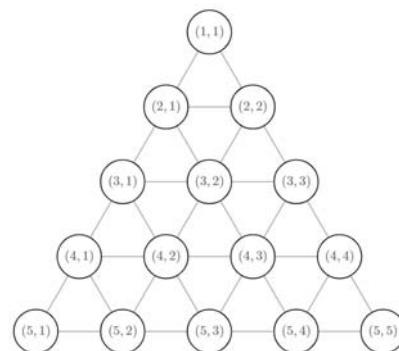
S264 - 三角領土

執行時間限制: 1.000 s / 記憶體限制: 256 MB

紅、藍兩個部落再次插旗以佔領領土。但這次他們不是在一條長線上，而是在一個三角形網格上佔領領土。

該三角形網格有 N 行。行以 1 至 N 從上到下表示，所以頂端的第一行是第 1 行，而底端的第一行是第 N 行。第 i 行有剛好 i 個單元格，以 1 至 i 從左至右表示。我們以 (i, j) 表示第 i 行中的第 j 個單元格。

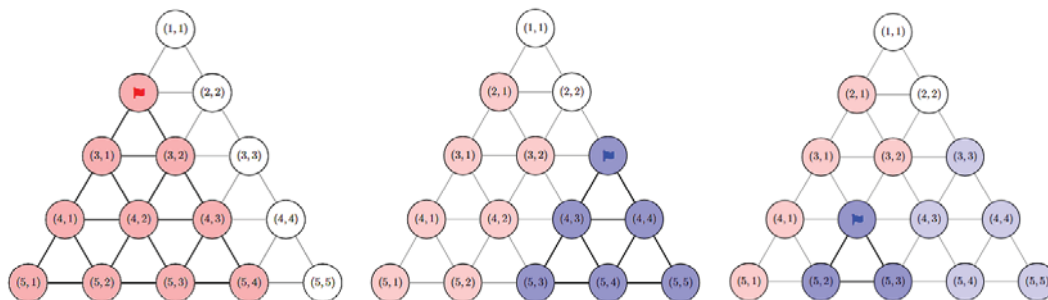
每個部落按以下方式佔領單元格：在三角形網格上的任何單元格上插上該部落顏色的旗幟（紅色或藍色其一）。然後，從旗幟所在的單元格開始，到最底行結束的子三角形上的所有單元格都被該部落佔領。請注意，如果子三角形上的一個單元格先前已被另一個部落佔領，則該單元格將被新插旗的部落佔領，並且不再被先前的部落佔領。正式地，一旦在單元格 (u, v) 上插上旗幟，所有滿足 $u \leq i \leq N$ 和 $v \leq j \leq v + (i - u)$ 的單元格 (i, j) 都將被旗幟的擁有者佔領，無論單元格 (i, j) 之前是否被任何部落佔領。請注意，旗幟可以插在已被其他部落佔領的單元格上，並且可以在同一個單元格上插上多個旗幟（顏色可以相同或不同）。



(有 $N = 5$ 行的三角形網格)

考慮以下在大小為 $N = 5$ 的三角形網格上的一系列插旗的範例。每次插旗後網格的結構如下所示。

1. 在單元格 $(2, 1)$ 上插上紅旗；
2. 在單元格 $(3, 3)$ 上插上藍旗；
3. 在單元格 $(4, 2)$ 上插上藍旗。



上述一系列插旗的結果是紅部落佔領了 5 個單元格，藍部落佔領了 8 個單元格。

在過去幾個月中，部落一直在插旗劃分領土。目前，網格上恰好有 X 個單元格屬於紅部落，恰好有 Y 個單元格屬於藍部落。記錄顯示，紅部落最多插了 A 面紅旗，而藍部落最多插了 B 面藍旗。然而，旗幟的位置以及插旗的順序都是未知的。

你的任務是復原網格上的一系列插旗序列，使得該序列最多包含 A 面紅旗和最多 B 面藍旗，並且其結果是紅部落在三角形網格上恰好佔領 X 個單元格，藍部落恰好佔領 Y 個單元格。記錄也有可能是錯誤的並且不可能構建這樣的序列，請報告此情況。

輸入

輸入的唯一一行包含五個整數： N 、 A 、 B 、 X 、 Y 。

輸出

如果不可能構建滿足約束的序列，輸出 -1 。

否則，輸出的第一行包含一個整數 K ，即插旗序列的長度。 K 必須滿足 $0 \leq K \leq A + B$ 。

輸出的接下來 K 行，每行描述一次插旗。這些行中的第 i 行應包含一個字符 c_i 和兩個整數 x_i 、 y_i ，表示插下的第 i 面旗幟的顏色為 c_i ，插在單元格 (x_i, y_i) 上。 c_i 必須是 R （紅色）或 B （藍色）。 (x_i, y_i) 必須是網格內的單元格，即必須滿足 $1 \leq y_i \leq x_i \leq N$ 。

滿足 $c_i = R$ 的 i 的數量應最多為 A ，而滿足 $c_i = B$ 的 i 的數量應最多為 B 。

插旗序列應使得紅部落恰好佔領 X 個單元格，藍部落恰好佔領 Y 個單元格。

如果有多個滿足上述限制的序列，你可以輸出其中任何一個。請注意，你不必最小化序列的長度。

樣例

輸入

輸出

1	5 2 2 5 8	3 R 2 1 B 3 3 B 4 2
---	-----------	------------------------------

這個輸出對應於問題陳述中所示的插旗序列。
該樣本測試滿足子任務 5、7 和 8 的約束條件。

2	5 2 2 5 8	4 R 3 3 R 2 1 B 4 3 B 3 1
---	-----------	---------------------------------------

這是樣本測試 1 的另一個有效的插旗序列。

3	6 1 3 6 5	4 B 6 5 R 3 1 B 6 3 B 5 1
---	-----------	---------------------------------------

該樣本測試滿足子任務 6、7 和 8 的約束條件。

輸入

輸出

4	6 9 4 2 0	4 B 6 2 B 6 1 R 6 2 R 6 1
---	-----------	---------------------------------------

該樣本測試滿足子任務 5、7 和 8 的約束條件。

5	4 1 1 4 5	-1
---	-----------	----

該樣本測試滿足子任務 4、7 和 8 的約束條件。

子任務

對於所有數據:

$$1 \leq N \leq 500$$

$$0 \leq A, B, X, Y \leq \frac{N(N+1)}{2}$$

$$1 \leq A + B \leq \frac{N(N+1)}{2}$$

$$1 \leq X + Y \leq \frac{N(N+1)}{2}$$

估分 約束條件

1	6	$A = 0$ $B = 1$
2	7	$A = 0$ $B = \frac{N(N+1)}{2}$
3	18	$A = 0$ $B \geq 2$
4	11	$A = B = 1$
5	15	$A, B \geq 2$
6	24	$A = 1$ $B \geq 3$
7	12	$N \leq 100$
8	7	無額外約束