



1

色塗り (Grid Coloring)

K 理事長は縦 N 行, 横 N 列のマス目で表される模様を作ろうとしている. そのために, 各マスに整数の番号で表される色を塗ることにした. 以降, 上から i 行目 ($1 \leq i \leq N$), 左から j 列目 ($1 \leq j \leq N$) のマスをマス (i, j) と呼ぶことにする.

現時点で, 1 列目と 1 行目のマスには既に色が塗られている. 具体的には, マス $(i, 1)$ ($1 \leq i \leq N$) は色 A_i で, マス $(1, j)$ ($1 \leq j \leq N$) は色 B_j で塗られている. ここで $A_1 = B_1$ である.

残りのマスについて, K 理事長は以下の手順で色を塗っていく.

- $i = 2, 3, \dots, N$ の順に, 以下の手順で i 行目のマスに色を塗る.
 - $j = 2, 3, \dots, N$ の順に, マス (i, j) を
 - * マス $(i-1, j)$ に塗られている色
 - * マス $(i, j-1)$ に塗られている色のうち番号が大きい方の色で塗る. 番号が同じ場合は, その色で塗る.

K 理事長は, 最終的に N^2 個のマスすべてに色が塗られたとき, 最も多くのマスに塗られた色の番号, およびその色が塗られているマスの個数を求めたい.

マス目の大きさおよび 1 列目と 1 行目のマスの情報が与えられたとき, 最も多くのマスに塗られた色の番号とその色が塗られているマスの個数を求めるプログラムを作成せよ. 最も多くのマスに塗られた色の番号が複数存在する場合, そのうち最も番号の大きいものを求め出力すること.

入力

入力は以下の形式で標準入力から与えられる.

```
N
A1 A2 ⋯ AN
B1 B2 ⋯ BN
```

出力

標準出力に, 最も多くのマスに塗られた色の番号と, その色が塗られているマスの個数を空白区切りで 1 行に出力せよ. 最も多くのマスに塗られた色の番号が複数存在する場合, そのうち最も番号の大きいものを出力すること.



制約

- $2 \leq N \leq 200\,000$.
- $1 \leq A_i \leq 10^9$ ($1 \leq i \leq N$).
- $1 \leq B_j \leq 10^9$ ($1 \leq j \leq N$).
- $A_1 = B_1$.
- 入力される値はすべて整数である.

小課題

1. (15 点) $N \leq 500$, $A_i \leq 100\,000$ ($1 \leq i \leq N$), $B_j \leq 100\,000$ ($1 \leq j \leq N$).
2. (10 点) $N \leq 500$.
3. (20 点) $A_i \leq 2$ ($1 \leq i \leq N$), $B_j \leq 2$ ($1 \leq j \leq N$).
4. (25 点) $A_i < A_{i+1}$ ($1 \leq i \leq N - 1$), $B_j < B_{j+1}$ ($1 \leq j \leq N - 1$).
5. (30 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
3 5 2 5 5 3 1	5 4

最終的に各マスに塗られる色の番号は以下ようになる.

5	3	1
2	3	3
5	5	5

最も多くのマスに塗られた色の番号は 5 であり, これは 4 個のマスに塗られているため, 5 と 4 を空白区切りで出力する.

この入力例は小課題 1, 2, 5 の制約を満たす.



入力例 2	出力例 2
3 1 7 8 1 3 5	8 3

最終的に各マスに塗られる色の番号は以下のようになる。

1	3	5
7	7	7
8	8	8

最も多くのマスに塗られた色の番号は 7 と 8 であり、それぞれ 3 個のマスに塗られている。2 つの色のうち最も番号が大きい色は 8 であるため、8 と 3 を空白区切りで出力する。

この入力例は小課題 1, 2, 4, 5 の制約を満たす。

入力例 3	出力例 3
4 2 1 2 1 2 1 1 2	2 10

この入力例は小課題 1, 2, 3, 5 の制約を満たす。



2

勇者ビ太郎 2 (Bitaro the Brave 2)

勇者のビ太郎は、モンスターを討伐しに冒険に出ることになった。

ビ太郎は強さという値を持っている。ビ太郎の強さの初期値を x とする。モンスターは N 体存在し、1 から N までの番号が付けられている。モンスター i ($1 \leq i \leq N$) を倒すには強さが A_i 以上である必要がある。モンスター i を倒すと強さが B_i 増える。

ビ太郎は冒険において次のような行動をとることですべてのモンスターを倒したい。

- ある j ($1 \leq j \leq N$) から始めて、モンスター $j, j+1, \dots, N$ を順に倒す。
- 次に、 $j \geq 2$ なら、モンスター $1, 2, \dots, j-1$ を順に倒す。

モンスターの情報が与えられたとき、すべてのモンスターを倒すために必要な強さの初期値 x の最小値を求めるプログラムを作成せよ。

入力

入力は以下の形式で標準入力から与えられる。

```
N
A1 A2 ... AN
B1 B2 ... BN
```

出力

標準出力に、すべてのモンスターを倒すために必要な強さの初期値の最小値を 1 行で出力せよ。

制約

- $2 \leq N \leq 500\,000$.
- $0 \leq A_i \leq 10^9$ ($1 \leq i \leq N$).
- $0 \leq B_i \leq 10^9$ ($1 \leq i \leq N$).
- 入力される値はすべて整数である。



小課題

1. (10 点) $N \leq 2000$, 必要な強さの初期値の最小値は 10 以下である.
2. (21 点) $N \leq 2000$.
3. (19 点) 必要な強さの初期値の最小値は 10 以下である.
4. (22 点) $B_i = 1$ ($1 \leq i \leq N$).
5. (28 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
5 1 3 2 8 6 4 3 1 1 2	1

強さの初期値が 1 であるとき, たとえば次のような順番ですべてのモンスターを倒すことができる.

- 強さの初期値を 1 とする.
- モンスター 1 を倒す. 強さが 4 増えて, 強さは 5 になる.
- モンスター 2 を倒す. 強さが 3 増えて, 強さは 8 になる.
- モンスター 3 を倒す. 強さが 1 増えて, 強さは 9 になる.
- モンスター 4 を倒す. 強さが 1 増えて, 強さは 10 になる.
- モンスター 5 を倒す. 強さが 2 増えて, 強さは 12 になる.

強さの初期値が 0 以下ですべてのモンスターを倒す方法は存在しないため, 1 を出力する.

この入出力例は小課題 1, 2, 3, 5 の制約を満たす.

入力例 2	出力例 2
5 1 6 3 3 2 1 2 1 0 1	3

強さの初期値が 3 であるとき, たとえば次のような順番ですべてのモンスターを倒すことができる.

- 強さの初期値を 3 とする.
- モンスター 3 を倒す. 強さが 1 増えて, 強さは 4 になる.



- モンスター 4 を倒す。強さが 0 増えて、強さは 4 になる。
- モンスター 5 を倒す。強さが 1 増えて、強さは 5 になる。
- モンスター 1 を倒す。強さが 1 増えて、強さは 6 になる。
- モンスター 2 を倒す。強さが 2 増えて、強さは 8 になる。

強さの初期値が 2 以下ですべてのモンスターを倒す方法は存在しないため、3 を出力する。
この入出力例は小課題 1, 2, 3, 5 の制約を満たす。

入力例 3	出力例 3
10 11 9 8 12 7 7 8 12 9 10 1 1 1 1 1 1 1 1 1 1	9

この入出力例は小課題すべての制約を満たす。

入力例 4	出力例 4
7 1125 638 0 37 737 820 1202 23 984 558 350 52 345 580	0

この入出力例は小課題 1, 2, 3, 5 の制約を満たす。



3

ミ・テレフェリコ (Mi Teleférico)

ボリビアの首都であるラパスは観光地であるとともに、ミ・テレフェリコ (Mi Teleférico) というロープウェイ路線網でも有名である。あなたはラパスに観光に来ており、できるだけ多くの場所を観光したいと思っている。ここで、現実を単純化した次のような状況設定を考えたい。

ラパスには N 個のロープウェイ駅があり、標高が低い順に 1 から N までの番号が付けられている。また、 M 個の一方通行の路線があり、1 から M までの番号が付けられている。さらに、 P 個のロープウェイ会社があり、1 から P までの番号が付けられている。各路線は 1 つの会社によって管理されている。路線 i ($1 \leq i \leq M$) は駅 A_i から駅 B_i に向かって運行しており、会社 C_i によって管理されている。ここで、路線は必ず標高の低い駅から標高の高い駅に向かって運行している。すなわち、 $A_i < B_i$ が成立している。

利便性のために、ラパスの交通局はフリーパスを発行した。それぞれのフリーパスには $1 \leq l \leq r \leq P$ を満たす 2 つの整数 l, r が書かれており、会社 $l, l+1, \dots, r$ によって管理されている路線に乗ることができる。すなわち、 $1 \leq i \leq M$ を満たす整数 i について $l \leq C_i \leq r$ を満たすならば路線 i に乗ることができる。ここで、1 つのフリーパスを複数の路線で使うことも可能である。このフリーパスをフリーパス (l, r) とする。

さて、ラパスに 1 から Q までの番号が付けられた Q 人の観光客が訪れた。観光客 j ($1 \leq j \leq Q$) はフリーパス (L_j, R_j) と、現金 X_j ボリビアーノを持っている。

観光客の目標は、持っているフリーパスを使って乗ることができる路線のみを用いて、駅 1 から移動できない駅がないようにすることである。そのために、観光客 j ($1 \leq j \leq Q$) は以下の手順で表される交換を行うことができる。ただし、各観光客について、交換は高々 1 回しか行うことができない。

1. $1 \leq l' \leq r' \leq P$ を満たす 2 つの整数 l', r' を決める。
2. フリーパス (L_j, R_j) とフリーパス (l', r') を交換する。手数料として $|L_j - l'| + |R_j - r'|$ ボリビアーノを要する。

あなたの目的は、それぞれの観光客について、持っている現金の範囲内で目標を達成することができるかを判定することである。

路線と観光客の情報が与えられたとき、それぞれの観光客について、持っている現金の範囲内で目標を達成することができるかを判定するプログラムを作成せよ。



入力

入力は以下の形式で標準入力から与えられる。

```
N M P
A1 B1 C1
A2 B2 C2
⋮
AM BM CM
Q
L1 R1 X1
L2 R2 X2
⋮
LQ RQ XQ
```

出力

標準出力に Q 行で出力せよ。 j 行目 ($1 \leq j \leq Q$) には、観光客 j が目標を達成することができる場合は Yes を、そうでない場合は No を出力せよ。



制約

- $2 \leq N \leq 300\,000$.
- $1 \leq M \leq 300\,000$.
- $1 \leq P \leq 10^9$.
- $1 \leq A_i < B_i \leq N$ ($1 \leq i \leq M$).
- $1 \leq C_i \leq P$ ($1 \leq i \leq M$).
- $1 \leq Q \leq 400\,000$.
- $1 \leq L_j \leq R_j \leq P$ ($1 \leq j \leq Q$).
- $0 \leq X_j \leq 10^9$ ($1 \leq j \leq Q$).
- 入力される値はすべて整数である.

小課題

1. (7 点) $N \leq 50$, $M \leq 50$, $Q \leq 50$, $X_j = 0$ ($1 \leq j \leq Q$).
2. (8 点) $P \leq 10$.
3. (11 点) $P \leq 100$.
4. (23 点) $P \leq 300\,000$, $X_j = 0$ ($1 \leq j \leq Q$).
5. (9 点) $P \leq 300\,000$.
6. (22 点) $N \leq 8\,000$, $M \leq 8\,000$.
7. (20 点) 追加の制約はない.



入出力例

入力例 1	出力例 1
4 6 10	Yes
1 2 3	No
2 4 7	No
1 2 6	Yes
2 3 5	
3 4 2	
3 4 8	
4	
3 7 0	
5 6 0	
3 4 0	
1 9 0	

まず、観光客 1 については、最初フリーパス (3, 7) と現金 0 ポリビアーノを持っている。この観光客は、フリーパスの交換を行わないことで目標を達成できる。なぜなら、フリーパス (3, 7) により乗ることができる路線は 1, 2, 3, 4 の 4 つであり、この 4 つの路線を用いて、以下のように駅 1 から各駅に移動することができるからである。

- 路線 3 を用いることで、駅 1 → 2 と移動することができる。
- 路線 1, 4 をこの順に用いることで、駅 1 → 2 → 3 と移動することができる。
- 路線 3, 2 をこの順に用いることで、駅 1 → 2 → 4 と移動することができる。

したがって、1 行目に **Yes** を出力する。

次に、観光客 2 については、最初フリーパス (5, 6) と現金 0 ポリビアーノを持っているが、この観光客は目標を達成することができない。なぜなら、フリーパス (5, 6) により乗ることができる路線は路線 3, 4 の 2 つのみであり、この 2 つの路線を用いて、駅 1 から駅 4 へ移動することができないからである。また、現金を 0 ポリビアーノしか持っていないため、異なるフリーパスと交換することができないからである。

したがって、2 行目に **No** を出力する。

さらに、観光客 3 については目標を達成することができず、観光客 4 については目標を達成することができるため、3 行目に **No** を出力し、4 行目に **Yes** を出力する。

この入力例はすべての小課題の制約を満たす。



入力例 2	出力例 2
4 6 10	Yes
1 2 3	No
2 4 7	Yes
1 2 6	
2 3 5	
3 4 2	
3 4 8	
3	
5 6 10	
3 4 1	
7 8 3	

路線の情報は入力例 1 と同じである。

まず、観光客 1 については最初フリーパス (5, 6) と現金 10 ポリビアーノを持っている。この観光客は、フリーパスの交換を次のように行うことで目標を達成できる。

1. $1 \leq l' \leq r' \leq P$ を満たす 2 つの整数として $l' = 1, r' = 5$ と決める。
2. フリーパス (5, 6) とフリーパス (1, 5) を交換する。手数料として $|5 - 1| + |6 - 5| = 5$ ポリビアーノを要する。

したがって、1 行目に **Yes** を出力する。

次に、観光客 2 については最初フリーパス (3, 4) と現金 1 ポリビアーノを持っている。この観光客はどのようなフリーパスの交換を行っても、目標を達成することができない。

したがって、2 行目に **No** を出力する。

さらに、観光客 3 については目標を達成することができるため、3 行目に **Yes** を出力する。

この入力例は小課題 2, 3, 5, 6, 7 の制約を満たす。

入力例 3	出力例 3
3 1 10000000000	No
1 2 6	
1	
1 10000000000 10000000000	

この路線において、駅 1 から駅 3 へ移動することができない。したがって、フリーパスによらず、観光客は目標を達成することはできない。

この入力例は小課題 6, 7 の制約を満たす。



第 24 回日本情報オリンピック (JOI 2024/2025) 本選
2025 年 2 月 2 日 (オンライン開催)

入力例 4	出力例 4
5 9 2000	Yes
2 3 1814	Yes
2 3 457	Yes
1 2 1226	Yes
3 4 1354	No
1 5 1050	
1 2 1725	
2 3 1383	
1 5 1626	
1 4 1795	
5	
850 1872 128	
82 428 1217	
487 924 573	
1639 1926 202	
202 420 25	

この入力例は小課題 5, 6, 7 の制約を満たす。



4

長いだけのネクタイ 2 (Just Long Neckties 2)

Just Odd Inventions 社は、「ただ奇妙な発明 (just odd inventions)」をすることで知られている会社である。ここでは略して JOI 社と呼ぶ。

JOI 社は、看板商品である「長いだけのネクタイ」発売 5 周年を記念し、新しく「長くなるだけのネクタイ」を開発した。この新型ネクタイの特徴は、その名の通り長さをいくらでも伸ばせることである。

JOI 社は、新型ネクタイの宣伝を目的とした披露会の開催を決定し、その司会にあなたを抜擢した。披露会ではまず、新型ネクタイを着用した何人かのモデルが舞台上に登壇する。最初、各モデルが着用しているネクタイの長さはすべて 1 である。

その後、あなたはネクタイの長さを伸ばせる機能を観客に実感してもらうためのパフォーマンスを N 回行う。各パフォーマンスは以下のように行われる。

- まず、観客に好きな数を 1 つ唱えてもらう。ここで観客が唱えた数を x とおく。
- 次に、司会のあなたはこれに反応するか無視するかを選ぶ。
 - 反応することを選んだ場合、あなたは登壇しているモデルのうち着用しているネクタイの長さが x 以下であるような者を 1 人選び、そのモデルのネクタイの長さを x にする（着用しているネクタイの長さが元々 x であるようなモデルも選ぶことができる点に注意せよ）。ただし、選ぶことのできるモデルが 1 人も存在しない場合、披露会は失敗に終わる。
 - 無視することを選んだ場合、何もしない。

ただし、観客の唱えた数を 2 回以上連続で無視してしまうと、観客が機嫌を損ね、披露会は失敗に終わる。

舞台上に登壇させるモデルの人数 k ($k \geq 1$) はまだ決まっていないが、モデルを雇うにはお金がかかるため、 k の値はできるだけ小さい方が望ましい。披露会が失敗に終わらないために必要なモデルの人数は各パフォーマンスで観客が唱える数に依存するが、あなたはその予知能力により、 i 回目 ($1 \leq i \leq N$) のパフォーマンスで観客が唱える数が A_i であることを予見した。

披露会で観客が唱える数の情報が与えられたとき、披露会が失敗に終わらないために必要なモデルの人数 k の最小値を求めるプログラムを作成せよ。



入力

入力は以下の形式で標準入力から与えられる。

N
 $A_1 A_2 \cdots A_N$

出力

標準出力に、披露会が失敗に終わらないために必要なモデルの人数 k の最小値を 1 行で出力せよ。

制約

- $2 \leq N \leq 5\,000\,000$.
- $1 \leq A_i \leq 21$ ($1 \leq i \leq N$).
- 入力される値はすべて整数である。

小課題

1. (10 点) $N \leq 15$.
2. (6 点) $N \leq 500$, $A_i \leq 2$ ($1 \leq i \leq N$).
3. (12 点) $N \leq 500$, $A_i \leq 5$ ($1 \leq i \leq N$).
4. (18 点) $N \leq 500$, $A_i \leq 15$ ($1 \leq i \leq N$).
5. (26 点) $N \leq 500\,000$, $A_i \leq 15$ ($1 \leq i \leq N$).
6. (10 点) $N \leq 500\,000$.
7. (18 点) 追加の制約はない。



入出力例

入力例 1	出力例 1
5 5 3 4 2 1	2

$k = 2$ のとき、例えば以下のように披露会を行うことができる。

- まず、新型ネクタイを着用した 2 人のモデルが舞台上に登壇する。最初、各モデルのネクタイの長さはいずれも 1 である。
- 1 回目のパフォーマンスでは、観客は 5 を唱える。あなたはこれを無視する。
- 2 回目のパフォーマンスでは、観客は 3 を唱える。あなたはこれに反応して、1 人目のモデルを選び、そのネクタイの長さを 3 にする。各モデルのネクタイの長さはそれぞれ 3, 1 になる。
- 3 回目のパフォーマンスでは、観客は 4 を唱える。あなたはこれに反応して、1 人目のモデルを選び、そのネクタイの長さを 4 にする。各モデルのネクタイの長さはそれぞれ 4, 1 になる。
- 4 回目のパフォーマンスでは、観客は 2 を唱える。あなたはこれに反応して、2 人目のモデルを選び、そのネクタイの長さを 2 にする。各モデルのネクタイの長さはそれぞれ 4, 2 になる。
- 5 回目のパフォーマンスでは、観客は 1 を唱える。あなたはこれを無視する。

$k = 1$ のとき、披露会は必ず失敗に終わる。例えば、上記の進行例のように 2, 3, 4 回目のパフォーマンスで観客の唱えた数に反応した場合、3 回目のパフォーマンスが終了した時点で登壇している唯一のモデルのネクタイの長さが 4 になっているため、4 回目のパフォーマンスにおいて、着用しているネクタイの長さが 2 以下であるようなモデルを選ぶことができず、披露会は失敗に終わる。

よって、披露会が失敗に終わらないために必要なモデルの人数 k の最小値は 2 であるため、2 を出力する。この入力例は小課題 1, 3, 4, 5, 6, 7 の制約を満たす。



入力例 2	出力例 2
6 2 1 1 2 2 1	1

$k = 1$ のとき、例えば以下のように披露会を行うことができる。

- まず、新型ネクタイを着用した 1 人のモデルが舞台上に登壇する。最初、モデルのネクタイの長さは 1 である。
- 1 回目のパフォーマンスでは、観客は 2 を唱える。あなたはこれを無視する。
- 2 回目のパフォーマンスでは、観客は 1 を唱える。あなたはこれに反応して、登壇している唯一のモデルを選び、そのネクタイの長さを 1 にする。
- 3 回目のパフォーマンスでは、観客は 1 を唱える。あなたはこれに反応して、登壇している唯一のモデルを選び、そのネクタイの長さを 1 にする。
- 4 回目のパフォーマンスでは、観客は 2 を唱える。あなたはこれに反応して、登壇している唯一のモデルを選び、そのネクタイの長さを 2 にする。
- 5 回目のパフォーマンスでは、観客は 2 を唱える。あなたはこれに反応して、登壇している唯一のモデルを選び、そのネクタイの長さを 2 にする。
- 6 回目のパフォーマンスでは、観客は 1 を唱える。あなたはこれを無視する。

なお、上記の進行例における 2,3 回目のパフォーマンスでは、元々ネクタイの長さが 1 であるようなモデルを選んでそのネクタイの長さを 1 にしているが、そのように、ネクタイの長さが変化しないようなモデルの選び方も許されていることに注意せよ。

よって、披露会が失敗に終わらないために必要なモデルの人数 k の最小値は 1 であるため、1 を出力する。この入力例はすべての小課題の制約を満たす。

入力例 3	出力例 3
10 2 4 6 7 4 5 5 3 4 1	3

この入力例は小課題 1,4,5,6,7 の制約を満たす。



5

郵便局 (Post Office)

JOI 国には N 個の郵便局があり、それぞれ 1 から N までの番号が付けられている。各郵便局には「送り先」が 1 つだけ指定されており、郵便局 i の送り先は郵便局 P_i である。ただし $P_i = i$ である可能性もある。もし時刻 t に郵便局 i から荷物を一つ発送した場合、時刻 $t + 1$ に郵便局 P_i にその荷物が到着する。ただし、荷物を発送している間はその郵便局から別の荷物を発送することができない。また、各郵便局には個数の制限なく荷物を保管しておくことができる。

さて、これから JOI 国では M 個の荷物を届けることになっている。 j 個目の荷物は時刻 0 に郵便局 A_j に到着し、最終的に指定の郵便局 B_j に届けなければならない。郵便局と荷物の情報が与えられたとき、すべての荷物を指定の郵便局に届けられるかを判定し、もし可能ならば最後に荷物が指定の郵便局に届く時刻として考えられる最も小さな値を求めるプログラムを作成せよ。

入力

入力は以下の形式で標準入力から与えられる。

```
 $N$   
 $P_1 P_2 \cdots P_N$   
 $M$   
 $A_1 B_1$   
 $A_2 B_2$   
 $\vdots$   
 $A_M B_M$ 
```

出力

標準出力に 1 行で出力せよ。すべての荷物を指定の郵便局に届けられる場合は、最後に荷物が指定の郵便局に届く時刻として考えられる最も小さな値を出力せよ。そうでない場合は、代わりに -1 を出力せよ。



制約

- $2 \leq N \leq 200\,000$.
- $1 \leq M \leq 200\,000$.
- $1 \leq P_i \leq N$ ($1 \leq i \leq N$).
- $1 \leq A_j, B_j \leq N$ ($1 \leq j \leq M$).
- $A_j \neq B_j$ ($1 \leq j \leq M$).
- 入力はすべて整数である.

小課題

1. (3 点) $N \leq 3\,000$, $M = 1$.
2. (9 点) $N \leq 3\,000$, $M \leq 3\,000$.
3. (13 点) $P = (1, 1, 2, \dots, N - 1)$. また, $\max(B_1, B_2, \dots, B_M) < \min(A_1, A_2, \dots, A_M)$ である.
4. (25 点) $P = (1, 1, 2, \dots, N - 1)$.
5. (11 点) $P = (N, 1, 2, \dots, N - 1)$.
6. (25 点) $P_1 = 1$, $P_i < i$ ($2 \leq i \leq N$).
7. (14 点) 追加の制約はない.

入出力例

入力例 1	出力例 1
5 1 1 2 3 4 3 3 2 3 1 3 1	3

例えば, 以下のような送り方をすることによって時刻 3 までにすべての荷物を指定の郵便局に届けることができる.

- 時刻 0 には, 郵便局 3 に荷物 1, 2, 3 がある. このうち荷物 2 を郵便局 2 へ発送する.
- 時刻 1 には, 郵便局 2 に荷物 2, 郵便局 3 に荷物 1, 3 がある. 郵便局 2 からは荷物 2 を郵便局 1 へ



発送し、郵便局 3 からは荷物 3 を郵便局 2 へ発送する。

- 時刻 2 には、郵便局 1 に荷物 2, 郵便局 2 に荷物 3, 郵便局 3 に荷物 1 がある。郵便局 2 からは荷物 3 を郵便局 1 へ発送し、郵便局 3 からは荷物 1 を郵便局 2 へ発送する。
- 時刻 3 には、郵便局 1 に荷物 2, 3, 郵便局 2 に荷物 1 がある。このとき、すべての荷物が送り先に届いている。

時刻 2 までにすべての荷物を指定の郵便局に届けることはできないため、3 を出力する。
この入力例は小課題 2, 3, 4, 6, 7 の制約を満たす。

入力例 2	出力例 2
3 2 1 3 1 1 3	-1

どのような送り方をしても郵便局 1 から郵便局 3 へ荷物を運ぶことはできないため、-1 を出力する。
この入力例は小課題 1, 2, 7 の制約を満たす。

入力例 3	出力例 3
7 1 1 2 3 4 5 6 6 4 2 5 1 5 3 6 2 7 3 7 6	5

この入力例は小課題 2, 4, 6, 7 の制約を満たす。



入力例 4	出力例 4
4 4 1 2 3 4 4 1 4 1 2 3 2 3	4

この入力例は小課題 2,5,7 の制約を満たす。

入力例 5	出力例 5
7 1 1 1 3 3 4 4 5 6 1 6 3 7 1 5 1 5 1	5

この入力例は小課題 2,6,7 の制約を満たす。

入力例 6	出力例 6
11 3 1 2 5 6 7 8 4 4 5 10 6 2 1 9 8 11 8 10 4 5 6 5 7	6

この入力例は小課題 2,7 の制約を満たす。