Tsuen Wan Government Secondary School Olympiad in Informatics Team
St. Mark's School Programming Team
Ying Wa Girls' School Programming Team

2025/26 HKOI Final Mock I

SCHEDULE

Date	26 / 28 November 2025
Duration	3 hours

PROBLEMS

	Problem	Limit: Time	Limit: Memory	Score: Subtasks	Score: Total
TMF261	Sword and Shield	1s	256MB	10 15 19 11 16 14 8 7	100
TMF262	New Horizons	1s	256MB	11 12 13 14 17 25 8	100
TMF263	Breath of the Wild	1s	256MB	8 10 11 13 10 6 14 15 13	100
TMF264	Air Ride	1s	256MB	9 6 10 20 14 11 12 18	100
				Full Score	400

REMINDERS

- All problems are divided into subtasks. All test cases in a subtask must be passed to get points.
- You may attempt the problems using C++20 or Python 3 (secondary language). It is not guaranteed that the problems are solvable using secondary language.
- Unless otherwise specified, your output will be automatically fixed as follows:
  - Trailing spaces in each line will be removed.
  - An end-of-line character will be added to the end of the output if not present.
- 64-bit integers may be required in some problems, use `long long` in C++ if needed.

TMF261 Sword and Shield

Score	Limit	
100	1s	256MB

Victor and Gloria are now playing the game *Sword and Shield*, an interesting competitive game between two players. As the name suggests, there are two opposite factions — Sword and Shield. The Sword faction believes that swords can break through any kind of defense with their sharp edge while the Shield faction believes that shields can protect from any kind of attack with their tough defense. The contradicting beliefs caused them to fight with each other again and again in order to prove their truths.

And the reality is always the same, none of the factions can defeat their opponents.

In this new era, Victor and Gloria are leaders of the factions respectively. They have developed a completely new technique which allows them to use a sword/shield multiple times without cooldown. The improvement of reducing intervals between each action completely changes the battle from the turn-based to real-time. Shields can be stacked up to protect from a strong attack while swords can be used to break through multiple layers of defense.

More specifically, the strength of a sword and a shield can be described by an integer. Suppose there is an attack by a sword of strength x_i to a shield of strength y_i.

- If $x_i < y_i$, the shield successfully protects and the attack will **end** while the strength of the shield will **remain unchanged**.
- If $x_i \geq y_i$, the shield will be broken and the attack will **continue** and try to break through other shields with strength $x_i - y_i$.

Notice that when multiple shields are stacked up, the attack **must** break through them from the **most recently** made unbroken shield to the oldest unbroken shield.

Now, there is a new battle between Victor and Gloria. It can be described by Q operations in **chronological order**. There are two types of operations, Sword and Shield.

t_i	Name	Operation
1	Sword	n_i swords with strength x_i are created. Each sword begins its attack only after the previous one has finished. Note that any unused swords will be discarded after the operation ends, i.e. will not be carried into the future operations.
2	Shield	m_i shields with strength y_i are built in front of the existing shields.

Victor, the leader of the Sword faction, would like to compare the power of swords and shields. Therefore, he would like to know the number of shields **broken** after each **Sword operation**. Can you help him calculate the number of shields broken in each Sword operation?

INPUT

The first line consists of an integer Q.

The i^{th} line of the next Q lines starts with an integer t_i, representing the type of the i^{th} operation.

If $t_i = 1$, a Sword operation is performed, two integers n_i and x_i follow, representing n_i swords with strength x_i are created.

If $t_i = 2$, a Shield operation is performed, two integers m_i and y_i follow, representing m_i shields with strength y_i are built.

OUTPUT

For each Sword operation ($t_i = 1$), output a line consisting of an integer, the number of shields broken in this operation.

SAMPLE TESTS

	Input	Output
1	2 2 5 3 1 1 8	2

- The first operation builds 5 shields of strength 3.
- The second operation creates 1 sword of strength 8.
  That sword first breaks a shield of strength 3, reducing its strength to 5.
  Then it continues to break the second shield and further reduces its strength to 2.
  Finally, as $2 < 3$, it is stopped by the third shield.

	Input	Output
2	3 2 5 3 1 1 8 1 1 8	2 2

Note that the second operation will not reduce the strength of the third shield.

	Input	Output
3	2 2 5 3 1 2 8	4

The second operation is equivalent to the second and third operations in Sample Test 2.

	Input	Output
4	3 2 5 3 2 1 6 1 2 8	3

The attack must first break the shield created in the second operation as it is the most recently made.

	Input	Output
5	2 2 5 3 1 10 8	5

All shields are broken.

	Input	Output
6	2 2 5 1000 1 100 8	0

No shield is broken.

7	5 2 10 2 2 7 5 1 4 14 2 3 4 1 5 5	11 7

CONSTRAINTS

$1 \leq Q \leq 2 \times 10^5$

$1 \leq t_i \leq 2$

$1 \leq n_i, x_i, m_i, y_i \leq 10^4$

SUBTASKS

	Points	Constraints
1	10	$x_1 = y_1 = x_2 = y_2 = \cdots = x_Q = y_Q$
2	15	$Q = 2$ $t_1 = 2$ $t_2 = 1$
3	19	$n_i = m_i = 1$
4	11	$y_i = 1$
5	16	$t_i \geq t_j$ for $i < j$
6	14	$n_i = 1$
7	8	$m_i = 1$
8	7	No additional constraints

TMF262 New Horizons

Score	Limit	
100	1s	256MB

On your *Animal Crossing* island, Rootland, villagers communicate with the outside world using parrots. However, the parrots may sometimes deliver messages wrong!

As research go beyond, the villagers soon discover how the parrots actually change the message. They conclude that a received message can be converted back to the original message in two steps:

Step	Name	Operation
1	Replace	Replace the r^{th} character in the message with letter c, which can be the same as the original letter.
2	Swap	Swap the x^{th} character with the y^{th} character in the message, where $x \neq y$ but the characters are not necessarily be distinct.

Note that both steps have to be carried out following the order.

Now, another parrot has arrived the island with message R. With that, villagers start to have some guesses on the original message. To speed up the decryption process, you are tasked with writing a program to determine whether a guess G can be a possible original message and determine the corresponding steps if so.

INPUT

The first line consists of an integer N, representing the length of the message.

The second line consists of a string R of length N, representing the message received.

The third line consists of a string G of length N, representing the guess villagers made.

OUTPUT

If it is impossible to retrieve G from R, output `IMPOSSIBLE` on the first and only line.

Otherwise, if it is possible to retrieve G from R, output two lines.

The first line should consist of an integer r and a character c, separate them by a space, representing the index of the character chosen and its new letter respectively in step 1 (Replace).

The second line should consist of two integers x and y ($x \neq y$), separate them by a space, representing the indies of the characters chosen in step 2 (Swap).

If there are multiple valid sets of steps, you may output any of them.

SAMPLE TESTS

	Input	Output
1	8 iloveict ilovesit	7 s 6 7
2	8 iloveict ihateict	IMPOSSIBLE
3	3 ioi noi	3 n 1 3
4	4 hkoi hsoi	IMPOSSIBLE
5	5 twgss tmgss	2 m 4 5

CONSTRAINTS

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

R, G consist of lowercase English letters (`a` - `z`)

SUBTASKS

	Points	Constraints
1	11	$3 \leq N \leq 2 \times 10^5$ R, G consist of `a` and `b` only Each R, G contains exactly 1 `b`
2	12	$N = 2$
3	13	$R = G$
4	14	$R_i \neq R_j$ for $i \neq j$ $G_i \neq G_j$ for $i \neq j$
5	17	$2 \leq N \leq 100$
6	25	It is possible to retrieve G from R
7	8	No additional constraints

TMF263 Breath of the Wild

Score	Limit	
100	1s	256MB

Once upon a time, there was a great wilderness. The wilderness is long but narrow, which can be regarded as an $1 \times L$ rectangular grid. The i^{th} cell can be denoted by $(1, i)$.

There are N magical shrines in the wilderness, where the i^{th} shrine is located at $(1, P_i)$. These shrines contain magic power that allows creatures to teleport between any two shrines in just a second without any cost, acting as waypoints in this vast land.

The view of the wilderness is fascinating and the existence of the shrines are truly amazing. Link would like to see the entire view of the wilderness. However, his sight is fairly limited. If he is at $(1, x)$, he can only see cells from $(1, \max(x - K, 1))$ to $(1, \min(x + K, L))$ (inclusive). Therefore, he is going to design a trip across the wilderness such that he can see every cell in the wild during the trip.

As the trip may be quite long, Link decides to wisely use the power of the magical shrines. Each second, he can either walk left or right to an adjacent cell, or teleport between any two shrines. Viewing the cells within his sight would not consume any time.

Initially, Link is at $(1, S)$. As he still needs to save *Princess Zelda*, he decides to minimise the time spent on this trip. You are asked to help him figure out the minimum time required for him to see all cells. Note that Link may end at any position at the end of the trip.

INPUT

The first line consists of four integers L, S, N and K, representing the length of the wilderness, Link's starting position, the number of shrines and Link's sight range respectively.

The second line consists of N integers, the i^{th} integer P_i represents the position of the i^{th} shrine.

OUTPUT

Output an integer, the minimum time required for Link to see every cell in the wilderness.

SAMPLE TESTS

1

Input	Output
10 1 2 9 1 10	0

Link can see all cells without moving.

2

Input	Output
10 1 2 4 1 10	1

Link can see all cells after spending a second teleporting from the first shrine to the second shrine.

3

Input	Output
11 1 2 4 1 11	2

Link can see all cells after spending a second teleporting from the first shrine to the second shrine and then spend another second moving left by a cell.

4

Input	Output
11 3 2 4 1 11	3

Link can spend two seconds walking to the first shrine and then teleport to the second shrine using a second.

5

Input	Output
11 1 2 1 3 9	8

6

Input	Output
11 1 3 5 1 5 9	1

Link can directly teleport to the third shrine using a second.

CONSTRAINTS

$1 \le S \le L \le 10^9$

$1 \le N \le 2 \times 10^5$

$0 \le K \le 10^9$

$1 \le P_1 < P_2 < \cdots < P_N \le L$

SUBTASKS

	Points	Constraints
1	8	$N = 1$
2	10	$S = 1$ $N = 2$ $P_1 = 1$ $P_2 = L$
3	11	$N = 2$ $P_1 = 1$ $P_2 = L$
4	13	$S = 1$ $N = 2$ $P_1 = 1$
5	10	$N = 2$ $P_1 = 1$
6	6	$N = 2$
7	14	$S = 1$ $N = 3$
8	15	$S = 1$
9	13	No additional constraints

TMF264 Air Ride

Score	Limit	
100	1s	256MB

"Kirby's here! He's flying this way!"

Riding on his Warp Star, Kirby is moving forward to stop Meta Knight's evil plan. If Kirby successfully hit Meta Knight, he can save the planet from Meta Knight! Suddenly, some enemies spawn between Kirby and Meta Knight, trying to stop Kirby from directly reaching Meta Knight. Looks like that an intense battle is going to begin.

There are N enemies, numbered from 1 to N. The size of the i^{th} enemy is S_i. Initially, the speed of the Warp Star that Kirby is riding on is K. When Kirby encounters an enemy, if its size is **greater than** the speed of the Warp Star, the Warp Star will break and **stop**. Otherwise, Kirby will inhale the enemy, **increasing** the speed of the Warp Star.

Formally, when Kirby with Warp Star of speed v hits enemy i, if $v < S_i$, Kirby will be stopped and the battle between Kirby and Meta Knight will begin instantly; if $v \geq S_i$, Kirby will inhale the enemy and the speed of the Warp Star will be increased by A_i.

As you have watched all videos from *Masahiro Sakurai on Creating Games*, you learnt that a game cannot be too easy nor too difficult. Therefore, you decide to rearrange the order of the enemies that Kirby will encounter, such that **exactly** T enemies are left without being inhaled, making them part of the boss fight with Meta Knight. Can you find any arrangement or report that it is impossible?

INPUT

The first line consists of three integers N, T and K, representing the number of enemies, the target number of enemies to be left without being inhaled and the initial speed of Kirby's Warp Star respectively.

The second line consists of N integers, the i^{th} integer S_i represents the size of enemy i.

The third line consists of N integers, the i^{th} integer A_i represents the amount of speed Kirby's Warp Star will gain upon successfully inhaling enemy i.

OUTPUT

If it is impossible for T enemies to be left without being inhaled, output `IMPOSSIBLE` on the first and only line.

Otherwise, output `POSSIBLE` on the first line.

On the second line, output N integers, separate each by a space, with the i^{th} integer being the indices of the i^{th} enemy that Kirby will encounter. Each number from 1 to N should appear exactly once in the arrangement.

If there are multiple valid arrangements, you may output any of them.

SAMPLE TESTS

	Input	Output
1	5 2 1 1 1 3 4 5 2 0 1 0 0	POSSIBLE 2 1 3 5 4

- After inhaling the first enemy (enemy 2), the speed of the Warp Star remains unchanged at 1.
- After inhaling the second enemy (enemy 1), the speed of the Warp Star becomes 3.
- After inhaling the third enemy (enemy 3), the speed of the Warp Star becomes 4.
- As $4 < 5$, Kirby is stopped when facing the fourth enemy (enemy 5).

There are $T = 2$ enemies (enemy 5, 4) left at the end.

	Input	Output
2	5 1 1 1 1 3 4 5 2 0 1 0 0	POSSIBLE 1 2 3 4 5
3	5 1 1 1 1 3 4 5 2 0 2 0 0	IMPOSSIBLE
4	5 3 1 1 1 3 4 5 2 0 2 0 0	POSSIBLE 1 2 5 4 3
5	5 4 1 1 2 3 4 5 1 1 1 1 1	POSSIBLE 1 3 2 4 5

CONSTRAINTS

$0 \le T \le N \le 2 \times 10^5$

$1 \le K, S_i \le 10^9$

$0 \le A_i \le 2$

SUBTASKS

	Points	Constraints
1	9	$A_i = 0$
2	6	$A_1 = A_2 = \cdots = A_N$
3	10	$T = 0$
4	20	$0 \le A_i \le 1$
5	14	$0 \le N \le 10$
6	11	$0 \le N \le 500$
7	12	$0 \le N \le 5000$
8	18	No additional constraints