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

2025/26 HKGOI Mock

SCHEDULE

Date 19 / 20 November 2025

Duration 3 hours

PROBLEMS

	Problem	*Limit*		*Score*	
		Time	*Memory*	*Subtasks*	*Total*
TMG261	Rummikub	1s	256MB	19 | 21 | 25 | 17 | 18	100
TMG262	Little Helper	1s	256MB	6 | 14 | 21 | 13 | 19 | 16 | 11	100
TMG263	Girls and Boys	1s	256MB	8 | 12 | 16 | 20 | 14 | 19 | 11	100
TMG264	Cat Warn	1s	256MB	14 | 21 | 18 | 17 | 20 | 10	100
TMG265	Water	1s	256MB	3 | 4 | 4 | 4 | 10 | 12 | 13 | 11 | 14 | 17 | 8	100
TMG266	Son of Rainbow	1s	256MB	9 | 16 | 15 | 14 | 13 | 12 | 11 | 5 | 3 | 2	100
				Full Score	600

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.

TMG261 Rummikub

Score	Limit	
100	1s	256MB

Attending his first-ever Science lesson in secondary school, Alice was paying attention to the teacher.
*"Everyone, remember a polyatomic **molecule** is made up of two or more atoms…"*
She wondered, *"Isn't Rummikub (**魔力橋**) consists of some coloured tiles with numbers on them? That's weird."*
Jokes aside, in the game Rummikub, each normal tile consists of a number with a colour. The number ranges from 1 to 13 (inclusive), while the colour can be `Blue`, `Red`, `Orange` or `Black`.

Now, three tiles are given in ascending order of number printed on them. There are two types of valid sets:

Name	Requirement
Run	Three tiles of the same colour with consecutive numbers.
Group	Three tiles with the same number but different colours.

The following are some examples of valid and invalid sets:

Apart from the normal number-colour tiles, one of the three given tiles maybe a Joker tile. In forming sets, the Joker tile can be treated as any normal tile. For example, you can consider a Joker tile as a `Blue` 5, `Red` 9, etc.

You are asked to determine whether the given three tiles can form a valid set.

INPUT

The input consists of three lines, each representing a tile.

The i^{th} line first consists of a string S_i. If $S_i =$ `Joker`, the tile is a Joker. Otherwise, S_i represents the colour of the tile, and is followed by an integer N_i, representing the number on the tile.

OUTPUT

If the given three tiles can form a valid set, output `Valid`. Otherwise, output `Invalid`.

SAMPLE TESTS

	Input	Output

1	`Blue 5`
	`Red 5`
	`Black 5`
		`Valid`

2	`Blue 2`
	`Blue 3`
	`Blue 4`
		`Valid`

3	`Blue 7`
	`Red 8`
	`Black 8`
		`Invalid`

4	`Blue 7`
	`Joker`
	`Blue 9`
		`Valid`

The Joker tile can be considered as `Blue` 8.

5	`Red 5`
	`Joker`
	`Red 6`
		`Valid`

The Joker tile can be considered as `Red` 4 or `Red` 7.

6	`Joker`
	`Orange 1`
	`Orange 13`
		`Invalid`

CONSTRAINTS

$S_i =$ `Blue`, `Red`, `Orange`, `Black` or `Joker`

There is at most 1 `Joker` between S_1, S_2, S_3

$1 \le N_1 \le N_2 \le N_3 \le 13$

SUBTASKS

	Points	Constraints
1	19	$S_i \neq$ `Joker` $N_i = 5$
2	21	$S_i =$ `Blue`
3	25	$S_i \neq$ `Joker`
4	17	$S_3 =$ `Joker`
5	18	No additional constraints

TMG262 Little Helper

Score	Limit	
100	1s	256MB

After the mid-year exam, CPC once again needs to grade many papers. Since he still has to prepare next week's lecture, he has asked you to write a program called *CheckGPA* to help him evaluate the student's score.

The student is required to rewrite the sentence using a given tense T, which is either `present`, `past` or `continuous`. Both the original sentence Q and the student's answer A consist of N words, each consists of English letters (`A-Z`, `a-z`) only. For each word, it is a noun if its first letter is in uppercase (`A-Z`), otherwise is a verb if its first letter is in lowercase (`a-z`).

The i^{th} word in the original sentence Q and the student's answer A is denoted by Q_i and A_i respectively. The i^{th} word is considered correct if their parts of speech (noun / verb) of Q_i and A_i are the same and its corresponding check is passed:

Parts of Speech	*Check*
Noun	■ Case-sensitive (uppercase and lowercase are considered different). ■ A_i must exactly match Q_i.
Verb	■ Case-insensitive (uppercase and lowercase are considered the same). ■ If $T =$ `present`, A_i must exactly match Q_i. ■ If $T =$ `past`, A_i must equal Q_i followed by `ed` (or `Ed`, `ED`). ■ If $T =$ `continuous`, A_i must equal Q_i followed by `ing` (or `inG`, `iNg`, `InG`, etc.).

You are asked to determine the student's score for the question, which is the number of words that is correct.

INPUT

The first line consists of an integer N and a string T, representing the number of words and the required tense respectively.

The second line consists of N strings, the i^{th} string Q_i represents the i^{th} word in the original sentence.

The third line consists of N strings, the i^{th} string A_i represents the i^{th} word in the student's answer.

OUTPUT

Output an integer, the student's score.

SAMPLE TESTS

	Input	Output
1	2 continuous Adam orz Adam orZinG	2
2	4 past play IoI in SuMMer played IoI In SuMMER	2
3	5 present One Plus One Equal TWO One PluS ONE Eq TWOOOO	1

This sample satisfies the constraints of Subtask 2.

4	5 past a a a a a a as aed aing aingly	1

This sample satisfies the constraints of Subtask 4.

5	5 continuous a a a a a a as aed aing aingly	1

This sample satisfies the constraints of Subtask 4.

6	6 present i go to schOOl bY bUS i gO too shcool by buSes	3

This sample satisfies the constraints of Subtask 5.

CONSTRAINTS

$1 \le N \le 5000$

$T =$ `present`, `past` or `continuous`

Q_i, A_i consist of English letters (`A`-`Z`, `a`-`z`)

$1 \le$ length of Q_i, length of $A_i \le 20$

SUBTASKS

	Points	Constraints
1	6	$N = 1$ Q_1, A_1 starts with an uppercase English letter (`A`-`Z`)
2	14	Q_i, A_i starts with a uppercase English letter (`A`-`Z`)
3	21	$T =$ `present` length of $Q_i =$ length of A_i Q_i consists of `A` and `a` only Q_i, A_i starts with `a`
4	13	$Q_i =$ `a` A_i starts with `a` A_i consists of lowercase English letters (`a`-`z`) only
5	19	$T =$ `present` length of $Q_i =$ length of A_i Q_i, A_i starts with a lowercase English letter (`a`-`z`)
6	16	Q_i, A_i starts with a lowercase English letter (`a`-`z`)
7	11	No additional constraints

TMG263 Girls and Boys

Score	Limit	
100	1s	256MB

As the coach of the school's OI Team, you somehow forgot to take a group photo of the girls team members at HKGOI! Luckily, HKOI follows immediately, and all of them advanced to the Final Event! Therefore, you decide to gather all girls team members at HKOI Final Event for the photo.

"Girls and boys, let's take a group photo!" Not wanting your students to learn about your silly mistake of forgetting to take the photo at HKGOI, you decide to still let all team members *"participate"* in the photo.

There are N members lining up in a straight line. Their initial positions are represented by S, with S_i representing the gender of the i^{th} member from the left, which is either a boy (`B`) or a girl (`G`). To achieve a photo with girls only, you decide to ask some members to switch places. Specifically, in each step, you can only swap positions of two adjacent members (i.e. standing next to each other). At the end, all the girls should stand next to each other and all the boys should also stand next to each other, such that you can crop once and obtain a girls-only photo. For instance, `BBGGGG` and `GGBB` are valid final arrangements, while `BGBGGG` and `BGGB` are not.

"Yes, you swap with you. And you swap with you. And you…" Asking members to swap places are horribly time consuming! Therefore, you decide to find out the minimum number of steps needed to reach a valid final arrangement.

INPUT

The first line consists of an integer N, representing the number of members.

The second line consists of a string S, representing the initial arrangement.

OUTPUT

Output an integer, the minimum number of steps needed to result in a valid final arrangement.

SAMPLE TESTS

	Input	Output
1	7 GGGBGGG	3
2	6 BGBGBG	3
3	8 BBBBBBGG	0
4	8 GGGBGBGG	5

CONSTRAINTS

$1 \le N \le 5000$

S consists of `G` and `B`

SUBTASKS

	Points	Constraints
1	8	number of `B` in $S = 0$
2	12	$1 \le N \le 3$
3	16	number of `B` in $S \le 1$
4	20	number of `B` in $S \le 2$
5	14	N is even $S_i \neq S_{i+1}$
6	19	$1 \le N \le 10$
7	11	No additional constraints

TMG264 Cat Warn

Score	Limit	
100	1s	256MB

"Typhoon is approaching! It will be no. 10 tomorrow!"
"No way you can predict weather!"
"Nah, the day after is no. 11."
…

「內幕消息：聽日一定十號！」
「你唔好嗡得就嗡喎。」
「冇呃你，後日十一號。」
。。。

The typhoon season is approaching! With your amazing *"prediction"* skills, you decide to establish a typhoon tracking organisation called *"Cat Warn"*! In *Cat Warn*, you will use cats to predict future weather and provide *not necessarily accurate (but surely cute)* information to the public.

In Heung Shing, there are 8 weather stations located at different places, numbered from 1 to 8. The wind speed at each station is recorded every minute. The tropical cyclone will affect the city for N minutes. At the i^{th} minute, the wind speed recorded is $S_{i,1}, S_{i,2}, \ldots, S_{i,8}$ for station $1, 2, \ldots, 8$ respectively. During this period, Warning Signals will be issued to inform citizens about the current situation.

The four Warning Signals, from low to high, are as follows:

Number	Name	Requirement
`1`	Standby	No requirement.
`3`	Strong Wind	At least 4 weather stations observe wind speeds higher than or equal to 41.
`8`	Gale or Storm	At least 4 weather stations observe wind speeds higher than or equal to 63.
`10`	Hurricane	At least 1 weather station observes wind speed higher than or equal to 118.

A Warning Signal should be issued when its requirement is met. If there are multiple signals with their requirements met, the highest one will be in force. To prevent frequently switching between signals, unless replaced by a higher signal, a signal should be in force until its requirement is last met, even if there exists some time that the requirement is not met in between (see Sample Tests for examples).

Furthermore, the sequence of the in-force Warning Signals can only go from low to high, then back to low for at most once. For example, `1` → `3` → `10` → `8` → `3` is a valid sequence, while `3` → `8` → `3` → `10` is not. Following are more examples of valid and invalid sequences:

Although the tropical cyclone has not arrived yet, you have already used your cats to predict the wind speeds of the eight stations for the whole duration. You decide to use these *meowvellous* data to determine the total in-force duration of each signal.

INPUT

The first line consists of an integer N, representing the number of minutes.

The i^{th} line of the next N lines consists of eight integers $S_{i,1}, S_{i,2}, \ldots, S_{i,8}$, representing the wind speeds of the eight stations at the i^{th} minute.

OUTPUT

Output four integers on the first and only line, separate each by a space, the number of minutes that Warning Signal `1`, `3`, `8` and `10` is in force respectively.

SAMPLE TESTS

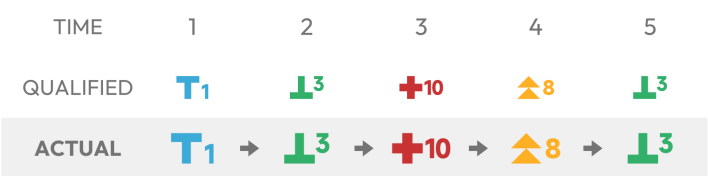
	Input	Output
1	1 62 46 124 42 85 51 89 102	0 0 0 1

Super Typhoon Ragasa (2025)

	Input	Output
2	1 65 56 114 42 52 43 79 82	0 0 1 0

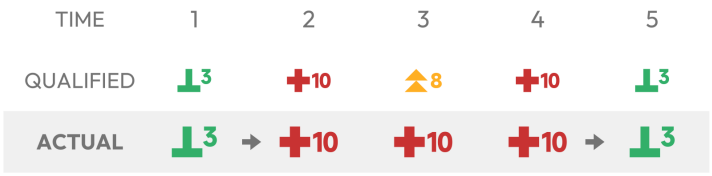
Typhoon Prapiroon (2006)

	Input	Output
3	5 0 0 0 0 0 0 0 0 41 41 41 41 41 41 41 41 118 118 118 118 118 118 118 118 63 63 63 63 63 63 63 63 41 41 41 41 41 41 41 41	1 2 1 1



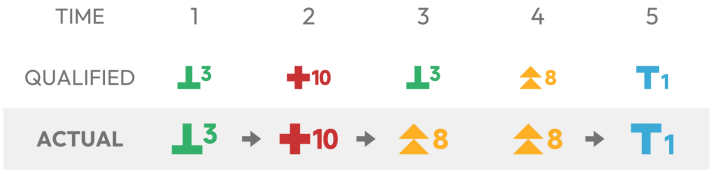
4

5 41 41 41 41 41 41 41 41 118 118 118 118 118 118 118 118 63 63 63 63 63 63 63 63 118 118 118 118 118 118 118 118 41 41 41 41 41 41 41 41	0 2 0 3



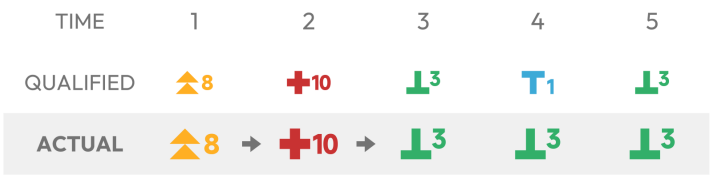
5

5 41 41 41 41 41 41 41 41 118 118 118 118 118 118 118 118 41 41 41 41 41 41 41 41 63 63 63 63 63 63 63 63 0 0 0 0 0 0 0 0	1 1 2 1



6

5 63 63 63 63 41 41 41 41 118 0 0 0 0 0 0 0 63 63 41 41 41 41 41 0 0 0 0 0 0 0 0 0 63 41 41 41 41 0 0 0	0 3 1 1



CONSTRAINTS

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

$0 \le S_{i,1}, S_{i,2}, \ldots, S_{i,8} \le 256$

SUBTASKS

	Points	Constraints
1	14	$N = 1$ $S_{1,1} = S_{1,2} = \cdots = S_{1,8}$
2	21	$N = 1$
3	18	$40 \le S_{i,1} = S_{i,2} = \cdots = S_{i,8} \le 41$
4	17	$S_{i,1} = S_{i,2} = \cdots = S_{i,8}$ $S_{N,1} = S_{N,2} = \cdots = S_{N,8} = 118$
5	20	$S_{i,1} = S_{i,2} = \cdots = S_{i,8}$
6	10	No additional constraints

TMG265 Water

Score	Limit	
100	1s	256MB

"You know what? About 60% of the human body is water. So you should be able to 'water' 60% of the scores in the competition!"

Here comes your opportunity to water!

There is a pool with N rows and M columns. Let (r, c) be the cell on the r^{th} row and the c^{th} column. A cell (x, y) is a neighbour of (r, c) if and only if $|x - r| + |y - c| = 1$. In the other words, $(r - 1, c)$, $(r + 1, c)$, $(r, c - 1)$, $(r, c + 1)$ are neighbours of (r, c). If at least 2 neighbour cells of (r, c) are filled with water, (r, c) will also become filled with water.

As a loyal supporter of water, Jack would like to choose x cells to manually fill with water, so that water spreads according to the above rule and fills the entire pool. Moreover, there are K ($0 \le K \le 1$) cells initially filled with water. If $K = 1$, the cell (R, C) is initially filled with water and Jack does not need to fill it himself. Being lazy, he would like to minimise x so he can pay minimum effort.

Jack asked you to help him to calculate the minimum number of cells he need to manually fill with water x, so that the water fills the whole pool, and to provide the corresponding positions of those cells.

INPUT

The first line consists of three integers N, M and K, representing the number of rows, number of columns and the number of cells initially filled with water respectively.

If $K = 1$, a line consists of two integers R and C follows, representing the position of the cell initially filled with water.

OUTPUT

On the first line, output an integer x, the number of cells Jack needs to manually fill with water.

On the i^{th} line of the next x lines, output two integers r_i and c_i, separated by a space, representing the position of the i^{th} cell that needs to be manually filled with water.

You may output the cells in any order. If there are multiple solutions, you can output any of them.

SAMPLE TESTS

	Input	Output
1	2 2 0	2 1 1 2 2

	Input	Output
2	4 2 1 3 1	2 1 1 4 2

	Input	Output
3	3 4 1 2 3	3 1 2 1 4 3 1

CONSTRAINTS

$1 \le N, M \le 100$

$0 \le K \le 1$

$1 \le R \le N$

$1 \le C \le M$

SUBTASKS

	Points	Constraints
1	3	$N = 2$ $M = 4$ $K = 0$
2	4	$N = 2$ $M = 5$ $K = 0$
3	4	$N = 2$ $M = 6$ $K = 0$
4	4	$N = M = 3$ $K = 0$
5	10	$N = 1$ $K = 0$
6	12	$N = 2$ $K = 0$
7	13	$K = 0$
8	11	$N = 1$
9	14	$N = 2$
10	17	N is even
11	8	No additional constraints

TMG266 Son of Rainbow

Score	Limit
100	1s | 256MB

Once upon a time, there is a *Son of Rainbow*. Son of Rainbow follows the footsteps of *Son of Weather*, where he creates rainbows based on weather of previous days. The weather of N consecutive days are given by a string W. The i^{th} character W_i represents the weather on the i^{th} day with `S` representing a sunny day, `R` representing a rainy day and `C` representing a cloudy day.

The appearance of rainbows is only determined by the number of sunny days and the number of rainy days in the past few days. If the number of sunny day $-$ the number of rainy day between the A^{th} day and the B^{th} day (inclusive) is exactly K, where $A \leq B$, a rainbow of brightness $B - A + 1$ will appear on the B^{th} day.

As the Son of Rainbow is extremely unstable, the exact value of K is still unknown, perhaps dynamically changing. Therefore, you have listed out Q questions. For the i^{th} question, you would like to find out the number of rainbows with brightness between l_i and r_i (inclusive) on the x_i^{th} day when $K = k_i$.

As the creator of the weather prediction platform *Cat Warn*, you would like to find out the answers to all questions to show that the Son of Rainbow is actually predictable.

INPUT

The first line consists of two integers N and Q, representing the number of days and number of questions respectively.

The second line consists of a string W, representing the weather of the N consecutive days.

The i^{th} line of the next Q lines consists of four integers k_i, x_i, l_i and r_i, representing the criteria of rainbow appearing, the target day and the range of the required rainbow brightness respectively.

OUTPUT

Output Q lines.

On the i^{th} line, output an integer, the answer to the i^{th} question.

SAMPLE TESTS

1

Input	Output
5 3 RCSSS 1 4 1 5 -1 2 1 2 3 5 4 5	2 1 1

For the first query, there is a rainbow of brightness 1 and a rainbow of brightness 4 on the 4^{th} day.
For the second query, there is a rainbow of brightness 2 on the 2^{nd} day.
For the third query, there is a rainbow of brightness 3 and a rainbow of brightness 4 on the 3^{rd} day.

2

7 6 RSRSRSR 1 7 1 7 0 7 2 4 0 7 2 7 0 7 1 5 -1 7 3 3 -1 7 4 4	0 2 3 2 1 0

CONSTRAINTS

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

W consists of `S`, `R` and `C`

$-N \le k_i \le N$

$1 \le x_i \le N$

$1 \le l_i \le r_i \le N$

SUBTASKS

	Points	Constraints
1	9	W consists of `S` only
2	16	$1 \le N, Q \le 5000$ $l_i = 1$ $r_i = N$
3	15	$1 \le N, Q \le 5000$
4	14	$l_i = r_i$
5	13	$x_i = N$
6	12	W consists of `S` and `C` only $k_1 = k_2 = \cdots = k_Q$
7	11	W consists of `S` and `C` only
8	5	$r_i = N$
9	3	$l_i = 1$
10	2	No additional constraints