New Zealand Programming Contest (NZPC)

8th August 2026

Contest Problems

Problem set contains 16 problems over 36 pages

In Memoriam

Phillip John Robbins

(1948 - 2026)

For 22 years, Phil served in the voluntary role of Contest Director, until ill health forced his resignation in May this year. Sadly, Phil died recently, on 22 July 2026. With quiet competency he ensured the entire multi-site contest ran smoothly year after year. We give sincere thanks for his professionalism and generosity, and for his unwavering belief in the value the contest experience gives students.

New Zealand Programming Contest

8 August 2026

Contest Instructions

Please read the following instructions carefully. They apply to every problem in this set unless a problem statement explicitly states otherwise.

Input and output

- Read all input from standard input, i.e. use stdin, `System.in`, `cin`, `input`, `Console.ReadLine` or equivalent. Input will be redirected from a file to form the input to your submission.
- Write all output to standard output, i.e. use stdout, `System.out`, `cout`, `print`, `Console.WriteLine` or equivalent. Do not write to standard error (stderr).
- **Do NOT prompt for input.** Any prompt text will appear in your output and cause the submission to be judged as wrong.
- Output is compared against the expected answer. Match the format shown in the problem exactly — extra spaces, blank lines, or missing/extra newlines can cause an otherwise correct solution to be judged wrong. A single trailing newline at the end of your output is accepted.
- Input files are sometimes terminated by a 'sentinel' line. This line should not be processed.
- A word is defined as a sequence of printable ASCII characters (*i.e.*, characters with ASCII values between 33 and 126). Unless stated otherwise in the problem statement, a single word or name has a maximum of 100 characters.
- You do not need to check that input data meets any specifications given in the problem statement, unless the problem specifically states that you should.
- Some problems are judged based on the relative or absolute error of your output (typically 10^{-6}). In these cases, suppose your answer is a, the jury's answer is b, and the tolerance is t, your answer must satisfy one of the following conditions to be considered correct:
  - $|a - b| \leq t$ (absolute error)
  - $|a - b| \leq t \cdot |b|$ (relative error)

Judging and scoring

- Submissions are judged automatically by DOMjudge. A submission is *Accepted* only if it produces the correct output on every test case within the time and memory limits.
- Common rejected verdicts are *WRONG-ANSWER*, *TIMELIMIT* (your program exceeded the time limit on a test case), *RUN-ERROR* (a crash or non-zero exit), and *COMPILER-ERROR*.
- Each problem has a time limit and memory limit specified on the problem. It is usually 1 second and 256 MB. The limit applies to each individual test run.
- Teams are ranked by the total number of points obtained, and ties are broken by total time. Time is the sum, over all solved problems, of the elapsed time from the start of the contest to the accepted submission, plus a penalty of 20 minutes for each rejected submission on a problem that is eventually solved. Rejected submissions on problems you never solve incur no penalty.

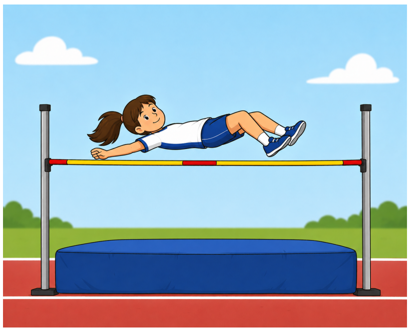
Submitting

- Submit your source file through DOMjudge. Domjudge automatically chooses the most matching langauge and problem for your submission. If you want to override this, you can select the language and problem manually.
- Any questions or clarifications about the problems should be asked through DOMjudge. This is the only official channel for clarifications. Questions asked through any other channels is prohibited.
- DOMjudge will reject a submitted file whose name contains spaces.

PROBLEM A - 3 Points

You're for the High Jump

Time Limit: 1.0 s | Memory Limit: 512 MiB



Every aspiring athlete dreams of competing at the New Zealand Physical Championship (NZPC) — the premier school sports event of the region. Before earning a place at the NZPC, however, students must first prove themselves at their local qualifying event.

At the school high jump event, all students have 3 attempts to qualify for the regional sports event high jump. The bar is set at a particular height and after all students have attempted that height, the bar is raised. This is repeated again to a 3rd height.

The 3 jump heights a student achieves are recorded against their name. Each name consists of a single word. If they balk (run, but don't jump), the height is recorded as 'NJ' for no jump. If they attempt the jump but fail to clear the bar, it is recorded as zero.

You are asked to write a program that names the students who have managed to qualify for the NZPC. A student qualifies if their best jump is higher or equal to the qualifying height.

Input

The first line contains a positive integer q ($1 \leq q \leq 10^3$), the qualifying height for the regional event in cm.

The second line contains a positive integer n ($1 \leq n \leq 10^4$), the number of students competing.

The following n lines each contain a student's name, then a space, followed by 3 space-separated jump values. Each value is either a zero (a failed jump) or a positive integer h ($1 \leq h \leq 10^3$) representing the height jumped in cm, or the letters `NJ` to represent no jump.

Output

Output the names of each student who has met the qualifying height on at least one of their 3 jumps, in the same order as they appear in the input. If no students met the qualifying height, the output should be the single line:

`Nobody qualifies!`

In the first sample below, Briggs jumped 150 in their last jump, higher than the 145 required, as did Santino.

Sample Input 1

```
145
5
Briggs 140 NJ 150
Emerson 140 143 0
Zhang 140 143 0
Santino 140 143 150
Cameron NJ NJ NJ
```

Sample Output 1

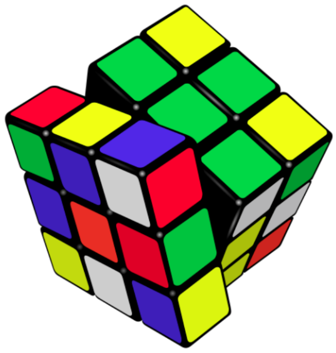
```
Briggs
Santino
```

Sample Input 2	Sample Output 2
145 3 Adams 130 0 NJ Brown 120 140 NJ Clark NJ NJ NJ	Nobody qualifies!

PROBLEM B - 3 Points

Speedcubing

Time Limit: 1.0 s | Memory Limit: 512 MiB



The National Zone Puzzle Championship (NZPC) is just around the corner, and Claire has never been more motivated to practise. As a dedicated speedcuber and ferociously competitive, she challenges her best friend Mai to a daily speedcubing competition every lunchtime in the school canteen. She wants to know what her longest win streak is over the last several games. The results are stored as two separate lists describing the number of seconds that it took Claire and Mai to complete their Rubik's cubes. The winner is the person who took the least amount of time to complete their Rubik's cube.

A streak is defined as a continuous run of wins. Streaks may start and end at any point in the recorded data and there may be more than one streak in the data. A streak ends with a loss or a draw.

Claire wants to know her longest winning streak over the last n days. Your task is to help her find it.

Input

The first line contains a single integer n ($1 \leq n \leq 50$), the number of days that Claire and Mai competed.

The second line contains n space-separated integers $c_1, c_2, \ldots, c_n$ ($1 \leq c_i \leq 10^4$), the number of seconds Claire took each day.

The third line contains n space-separated integers $m_1, m_2, \ldots, m_n$ ($1 \leq m_i \leq 10^4$), the number of seconds Mai took each day.

Output

Output a single integer: the length of Claire's longest winning streak.

For example, in the first sample, Claire was fastest in the second and third game but drew the fourth game making a streak of 2. She also won the 5th and 6th game, but that streak was not longer than the best previous streak.

Sample Input 1	Sample Output 1
6 12 11 14 15 16 12 10 14 19 15 21 14	2

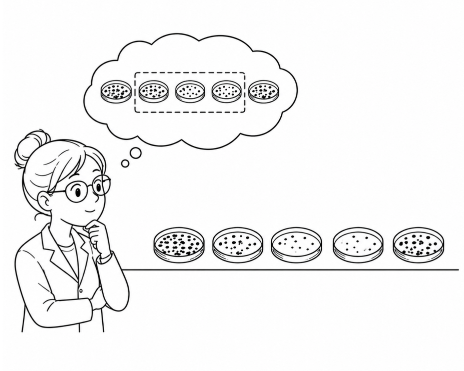
Sample Input 2	Sample Output 2
1 100 100	0

This page is intentionally left (almost) blank.

PROBLEM C - 3 Points

Germs

Time Limit: 1.0 s | Memory Limit: 512 MiB



The New Zealand Pathology Centre (NZPC) is home to some of the country's most dedicated scientific researchers — the kind who find a new strain of bacteria genuinely exciting.

Dr. Gemma works in the field of bacteriological research. As part of her work, she has to determine which range of temperatures helps to suppress the growth of a particularly nasty strain of the sniffles. Her experiments measure n petri dishes that are stored in a range of increasing temperatures. Dr Gemma is very keen to find a very specific range of temperatures where the bacteria is less frequent. Your task is to find which contiguous range of r dishes contains the lowest total sum of bacteria.

Input

The first line contains two space-separated integers, n $(0 < n \leq 10^3)$ and r $(0 < r \leq 100, r \leq n)$. The integer n represents the total number of petri dishes, and r represents the size of the contiguous range of dishes to consider.

The second line contains n space-separated integers, representing the amount of bacteria (b) present in each dish at the end of the experiment $(0 \leq b_i \leq 10^3)$.

Output

Output a single integer: the lowest total bacteria count in any contiguous range of exactly r dishes.

In the sample below, the values at index 2 and 3 sum to 9. No other contiguous range of values with 2 dishes sums to less than 9.

Sample Input 1	Sample Output 1
`6 2` `10 12 4 5 6 12`	`9`

This page is intentionally left (almost) blank.

PROBLEM D - 3 Points

Bisques

Time Limit: 1.0 s | Memory Limit: 512 MiB

The New Zealand Pastoral Classic (NZPC) is the most prestigious lawn sports tournament in the country, drawing teams from every region for a weekend of fierce but polite competition. With players of wildly different abilities competing together, keeping matches fair is crucial — and that is exactly what bisques are for.

In an association croquet (AC) handicap doubles event, each side may be awarded bisques. These are free turns given according to the handicaps of the 4 players so that weaker teams may compete with stronger teams on more even terms. In AC, handicaps range from -3 for the strongest players up to +24 for beginners.

To calculate bisques, each team calculates the total handicap of their two players. The difference between the two totals is then halved to give the number of bisques allocated to the weaker side, which is the team with the higher total handicap. This will be zero if the total handicaps are equal. Half bisques are allowed, but no points may be won when using a half bisque.

Given the a list of players and their handicaps, determine the number of bisques awarded and to which team.

Input

The input consists of 4 lines, each representing a player in a match. The first and second players form one team, the third and fourth the other.

Each line consists of a name consisting only of letters, followed by a space, followed by an integer representing that player's handicap (in the range -3 to +24 inclusive). + signs will not be used.

Output

Output a single line showing the number of bisques awarded and to which team.

If one team receives bisques, the line will say

`<B> bisques are awarded to <Player 1> and <Player 2>.`

where `<B>` is the number of free turns, and `<Player 1>` and `<Player 2>` are the appropriate team members in the order they appear in the input.

However, if `<B>` is 1, the line must instead say

`1 bisque is awarded to <Player 1> and <Player 2>.`

If the total handicaps are equal, the line will say:

`No bisques are awarded.`

Sample Input 1	**Sample Output 1**

`Albert 8`
`Bernice 4`
`Chen 3`
`Dimitri 12`
	`1.5 bisques are awarded to Chen and Dimitri.`

Sample Input 2	**Sample Output 2**

`Eddie -1`
`Freda 13`
`Garfield 5`
`Hettie 7`
	`No bisques are awarded.`

Sample Input 3	**Sample Output 3**

`Bob 0`
`Alice 2`
`Carol 1`
`Dave 3`
	`1 bisque is awarded to Carol and Dave.`

PROBLEM E - 10 Points

Best Takeaways in Town

Time Limit: 1.0 s | Memory Limit: 512 MiB

Every year, the New Zealand Palate Competition (NZPC) recognises the finest takeaway stores across the country. This year, the Super Grub chain is determined to earn that recognition. They have set up a monthly internal challenge in each area which has at least 3 of their stores. Evaluation-panel members place orders at each store at least 4 times each month, and award establishments according to their findings.

The chain's management have come up with the following attributes they want to measure:

- Food taste
- Food quality
- Portion size
- Store hygiene
- Quality of service
- Price

Each feature is graded A, B or C on each visit. At the end of the month, the grades are used to calculate a score for each store. An A is worth 5 points, a B is worth 2 points and a C is worth 0 points. The store with the highest average score across all visits is given the Store of the Month Award. The panel scores takeaway to guarantee there are no ties for the award.

Input

The first line contains the month of the survey, given as a single word with the full name of the month.

The second line contains the name of the town in which the stores are located (one or two words).

The next line contains a single integer N ($1 \leq N \leq 10^5$), the number of lines of data on the stores.

The following N lines each contain the name of one store with the 6 grades for one visit. The name may be up to 4 words and is followed by a comma. The grades are 6 space-separated letters, each being A, B, or C, representing the score for a different attribute from the list above.

Each store will appear at least 4 times in the list.

Output

Output a single line in the format:

```
<Store> was Store of the Month for <Town> in <Month>.
```

<Store> is the name of the store with the highest calculated score.

<Town> and <Month> are as shown in the input.

The line ends with a full stop.

In first sample below:

- Don’s Super Grub scores 6 As and 15 Bs, totalling 30 + 30 = 60 points. Average 15.
- South Street Super Grub scores 1 A and 15 Bs, 5 + 30 = 35, average 8.75.
- Elite Fish n Chips scores 8 As and 16 Bs, 40 + 32 = 72, average 18, the top score.

Sample Input 1

```
July
Masterton
12
Don’s Super Grub, A C B B A C
South Street Super Grub, B C B B C C
Elite Fish n Chips, A B B A B B
Don’s Super Grub, B B B B A B
South Street Super Grub, B B B A C B
Don’s Super Grub, A B B B A B
Elite Fish n Chips, A B A B B B
Elite Fish n Chips, B B B A B B
South Street Super Grub, B B B B C B
Don’s Super Grub, B B B B A C
South Street Super Grub, B B B C C C
Elite Fish n Chips, A B A A B B
```

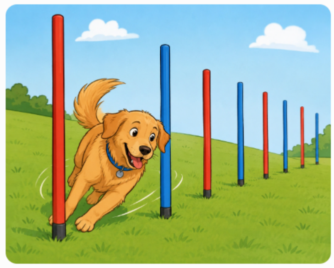
Sample Output 1

```
Elite Fish n Chips was Store of the Month for Masterton in July.
```

PROBLEM F - 10 Points

Puppies and Posts

Time Limit: 1.0 s | Memory Limit: 512 MiB



The New Zealand Puppy Championship (NZPC) is the nation's most beloved canine agility competition, open to dogs of every breed from svelte Greyhounds to stocky Labradors. Sarah, the owner of a puppy school, has entered several of her clients' dogs this year and wants to find out which of them are ready to compete.

At her school, Sarah has set up a practice course of a series of posts that are of varying widths apart. When the puppies arrive for training on a Saturday morning, part of their routine is to walk between the posts, in sequence. They weave from one side to the other as in a ski slalom.

Unfortunately Sarah has set the course based on her own very slim rescue Greyhound and not given thought to the Beagles and Labradors.

Write a program that outputs the name of the puppy who fits between the most posts. If at any point a dog can't fit between two posts, it tries the next pair of posts, and so on until it finds a gap it can fit through.

If there is more than one dog that achieves the same maximum number of gaps, the names should be printed in alphabetical order.

Input

The first line contains a sequence of at most 10^4 space-separated integers. Each integer represents the width in cm between 2 adjacent posts ($1 \leq w \leq 10^4$).

The second line contains an integer, n ($1 \leq n \leq 10$), representing the number of puppies.

The following n lines each begins with a single word describing a puppy's name, followed by a space and an integer p ($1 \leq p \leq 10^4$) which is the width in cm of the puppy at their widest part.

Output

Output the name of the dog who fits between the most posts. If there is more than one dog that fits between the most posts, output the names, one per line, in alphabetical order. If no dog fits through any of the gaps, output the single line

```
No dogs fit!
```

Sample Input 1	**Sample Output 1**
23 35 41 27 55 33 8 Freddie 43 Mercury 38 Ringo 33 Star 34 Germain 40 Bret 30 Clement 55 Mackenzie 35	Bret Ringo

Sample Input 2	**Sample Output 2**
12 14 16 18 1 Alpha 19	No dogs fit!

PROBLEM G - 10 Points

Arcane Rivalry

Time Limit: 1.0 s | Memory Limit: 512 MiB

You are an operative for the Natural Zodiac Patrol Corps (NZPC), the intelligence agency for arcane and cryptographic threats. Your latest assignment: two rival alchemists have been exchanging encrypted messages, and your superiors want them decoded. You have intercepted one such message — intelligence reports indicate they are using a modified Caesar cipher to obscure their notes.

At great cost, our spies discovered that only 27 unique characters are ever used in the alchemists' messages: the lowercase characters and the underscore character. This encryption process assigns each character a value, where $a = 1, b = 2, ..., z = 26, \_ = 27$, and then increases each character's value by some integer offset. If a the value exceeds 27, it wraps around to 1 and continues.

For example, the letter 't' is encoded as 20. With a shift of 5, it becomes a 'y' (which is the character encoded as 25).

You do not know the secret offset value, but your spies have intercepted a decrypted keyword that is guaranteed to appear at least once as a contiguous substring within the original plaintext message for all possible offsets.

Write a program to fully decrypt the alchemist's secret. There will be exactly one valid offset that yields a plaintext containing the keyword.

Input

The first line contains a string W with $4 \leq |W| \leq 30$, representing the unencrypted mystery keyword discovered by your spies.

The second line contains a string S with $|W| \leq |S| \leq 10^3$, representing the encrypted ciphertext message.

Both strings will contain only lower case letters from a-z and the underscore character.

Output

Output a single line containing the completely decrypted plaintext string.

Explanation

In the first sample below, shifting the entire ciphertext forward by 22 results in the plaintext string

`this_is_a_string_of_gold`

No other shift results in the string `gold` appearing in the plaintext string.

Sample Input 1	Sample Output 1
`gold` `ymnxenxefexywnsletkeltqi`	`this_is_a_string_of_gold`

This page is intentionally left (almost) blank.

New Zealand Programming Contest

8 August 2026

PROBLEM H - 10 Points
Order N

Time Limit: 1.0 s | Memory Limit: 512 MiB

```
####   ####  ###########
#####  ####  ##########
###### ####       ####
###########      ####
###########     ####
#### ######    ####
####  #####   #########
####   ####  ##########

#########      #######
####  ####   #########
####  ####   ####  ###
##########   ####
########     ####
####         ####  ###
####         #########
####          #######
```

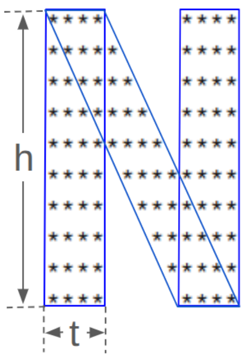
It is the day of the New Zealand Programming Contest (NZPC) that you are participating in right now, and your site coordinator has had an inspired idea. Between setting up the judging server and ordering sufficient quantities of pizza, they have decided that what the campus truly needs is large ASCII art signs spelling out the letters N, Z, P, and C. Because representing letters in various sizes with ASCII isn't quite as easy as it sounds, four separate programmers have each been given a single letter to program. Your letter is 'N'.

An ASCII art N is defined by a character c to be used for filling the letter and just two integers:

1. the height, h, of the letter (*i.e.*, the number of lines of text it occupies), and
2. a thickness parameter, t, which defines the width of the lines used to compose the letter N. Note that this includes both the vertical and diagonal components.

The figure below illustrates the meanings of h and t for a letter N of height 10 and thickness 4, drawn with asterisks ('*'s) as the character c. Only the asterisks are the output of the program; the blue lines are added to show the underlying geometry.

Note that the middle segment has a slope of 1, in the sense that the t asterisks it contains move one print position to the right on each successive row of output.



Input

The input consists of a single line, containing a non-space, printable ASCII character c and two integers, h and t ($5 \leq h \leq 20, 1 \leq t \leq h - 4$).

Output

Output h lines containing only spaces and the character c, displaying the ASCII art letter N as defined above.

Sample Input 1

```
* 10 4
```

Sample Output 1

```
****     ****
*****    ****
******   ****
*******  ****
******** ****
**** ********
****  *******
****   ******
****    *****
****     ****
```

Sample Input 2

```
$ 5 1
```

Sample Output 2

```
$   $
$$  $
$ $ $
$  $$
$   $
```

Sample Input 3

```
# 9 5
```

Sample Output 3

```
#####   #####
######  #####
####### #####
#############
#############
#############
##### #######
#####  ######
#####   #####
```

PROBLEM I - 30 Points

Raising the Flag

Time Limit: 1.0 s | Memory Limit: 256 MiB

For centuries, the Kingdom of Numeria has relied on an extensive network of watchtowers to guard its borders. Each tower flies a signal flag that can be seen from neighbouring towers, allowing urgent messages to travel across the kingdom in a matter of minutes. Although messengers have repeatedly suggested that carrier pigeons would be cheaper, the Royal Treasury insists that watchtowers are much more impressive.

The watchtowers along the roads of Numeria.

Every year, during the New Zenith Peace Celebration (NZPC), the Royal Signal Society performs an ancient ceremonial ritual. According to tradition, the ritual dates back to the reign of Queen Euleria II, who united Numeria after years of civil unrest. Historians argue endlessly over whether the ritual was originally intended as a military drill, a religious ceremony, or simply an excuse for the guards to avoid paperwork. Regardless of its origins, nobody has dared change it in over five hundred years.

The watchtowers are positioned along the kingdom's main road and are numbered from 1 to n. Before the ceremony begins, each tower's flag is either raised or lowered. During ritual round i, the performers visit every tower whose number is divisible by i and toggle its flag: a raised flag is lowered, and a lowered flag is raised.

For example, during the first round, every flag is toggled. During the second round, the flags at towers 2, 4, 6, $\ldots$ are toggled. During the third round, the flags at towers 3, 6, 9, $\ldots$ are toggled, and so on, until the n-th and final round.

Legend claims that the number of raised flags at the end of the ritual predicts the kingdom's fortunes for the coming year. Whether this is true is irrelevant, because the current King has become thoroughly convinced of it. After consulting with one particularly persuasive goat, the King has decided that the only acceptable outcome is exactly k raised flags.

As the kingdom's chief mathematician, you know that arguing with the King is considerably more difficult than solving mathematical problems. Instead, your task is to determine the initial positions of the flags so that, after the ritual has finished, exactly k flags are in a raised state.

Input

The input consists of one line with two integers:

- n ($1 \leq n \leq 10^6$), the number of watchtowers;
- k ($0 \leq k \leq n$), the King's favourite number of raised flags after the ritual.

Output

Output a single string of length n consisting only of the characters `0` and `1`. The i-th character represents the initial state of the flag at tower i, where `1` indicates raised and `0` indicates lowered.

If multiple valid initial configurations exist, output any of them.

Sample Input 1	Sample Output 1
`10 4`	`0100000000`

Sample Input 2	Sample Output 2
`20 1`	`00010000100000010000`

PROBLEM J - 30 Points

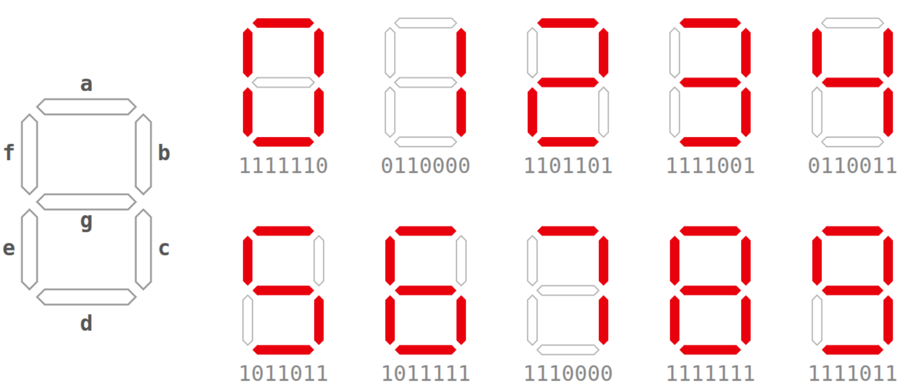
Segment Fault

Time Limit: 5.0 s | Memory Limit: 512 MiB

Angus thought it would be fun to have a large physical display by each NZPC team, displaying their current score. He decided to use a multi-digit 7-segment display.

However, Angus seems to have made some sort of hardware or software error. Angus suspects that in each digit, one segment is faulty and never lights up. The faulty segment may be in a different position from digit to digit, and some digits may have no faulty segment at all. This makes it difficult to tell what each team's true score was.

The segments of a single digit 7-segment display are labelled 'a' through 'g', as in the following figure. The figure also shows how the digits 0 through 9 are represented when all segments are functional. Underneath each digit is the representation of the display state using a 7-character string of zeros or ones, with the first character representing the 'a' segment, the second the 'b' segment and so on.



Because a faulty segment can only stay dark, an observed digit is some real digit with *at most one* of its lit segments switched off. For example, the pattern `1111110` normally shows a `0`; but it could also be an `8` (`1111111`) whose segment 'g' is faulty. So an observed `0` might really have been a `0` or an `8`.

Angus have come to you for help. He wants to know what the k-th largest possible original score could have been, for several values of k. The scores are compared as n-digit numbers, so the greatest possible value is considered the 1st largest, the second greatest value is the 2nd largest, and so on.

Input

The first line contains two integers n and q ($1 \leq n \leq 10^5$, $1 \leq q \leq 6$), the number of digits in the displayed score and the number of queries.

The next n lines each contain a 7-character binary string describing the observed state of one digit. The first character corresponds to segment 'a', the second to 'b', and so on.

The next q lines each contain a single integer k ($1 \leq k \leq 10^6$), asking for the k-th largest possible original score, given the observed digits.

Output

While Angus hopes there is only 1 segment being faulty in each digit, this is not guranteed. If some displayed digit cannot be produced by any real digit with at most one faulty segment, then Angus will throw the scoreboard away and you should output

```
ERROR
```

and terminate.

Otherwise, let m be the number of possible original scores.

For each query:

- If $k > m$, output

  ```
  NONEXISTENT
  ```

- Otherwise, output the k-th largest possible original score as an n-digit string, with leading zeros where appropriate.

Sample Input 1	Sample Output 1
2 4 1111110 0110000 1 2 5 3	87 81 NONEXISTENT 07

Sample Input 2	Sample Output 2
1 1 0000001 1	ERROR

Sample Input 3	Sample Output 3
3 2 1111111 0110011 1101101 1 2	842 NONEXISTENT

PROBLEM K - 30 Points

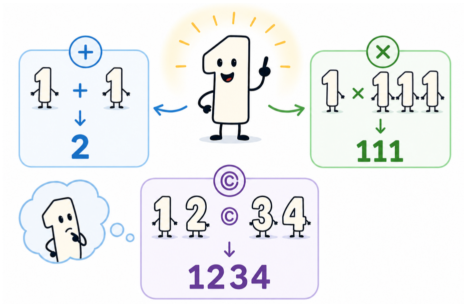
1's For All

Time Limit: 10 s | Memory Limit: 512 MiB

The *complexity* of an integer is the minimum number of 1's needed to represent it using only addition, multiplication and parentheses. For example, the complexity of 2 is 2 (writing 2 as $1 + 1$) and the complexity of 12 is 7 (writing 12 as $(1 + 1 + 1) \times (1 + 1 + 1 + 1)$).

To make the problem more interesting and more efficient, we'll add a new operation called Number Zipping via Piecewise Concatenation (NZPC). This operation (which we'll represent using ©) takes two integers and "glues" them together, so $12 © 34$ becomes the four digit number 1234. Using this operation, the complexity of 12 is now 3 (writing it either as $(1 © 1) + 1$ or $1 © (1 + 1)$). Note that for the NZPC operation you may not treat numbers as having leading zeros at all: $1 © 01$ does not result in 101 but results in 11.

The goal is to find the minimum number of 1's needed to represent the given integer using the revised definition.



Input

Each test case consists of a single line containing an integer n, where $1 < n \leq 10^5$.

Output

Output the complexity of the number, using the revised definition above.

Sample Input 1	Sample Output 1
2	2

Sample Input 2	Sample Output 2
12	3

Sample Input 3	Sample Output 3
100000	12

This page is intentionally left (almost) blank.

PROBLEM L - 30 Points

Alchemist's Numbers

Time Limit: 3.0 s | Memory Limit: 512 MiB

The Alchemists of Zeland have discovered a collection of mysterious crystals, known as nubreccia. According to Noble Zelandic Prophet's Chronicles (NZPC), the nubreccia will reveal a hidden message only when they are all synchronised—that is, when every crystal carries the same weight.

A sequence of 5 Nubreccia

Fortunately, the alchemists possess two magical operations.

- Increase the weight of a single nubreccium by multiplying it by a factor of p, where p is any prime number.
- Decrease the weight of a single nubreccium by dividing it by any prime number p (but only if the weight is divisible by p).

Each use of either operation costs exactly one unit of magical energy.

The Royal Alchemist wants to synchronise all crystals while using as little energy as possible.

Given the weights currently engraved on the crystals, determine the minimum number of operations required to make all weights equal.

Input

The first line contains an integer $1 \leq n \leq 10^5$, the number of crystals.

The second line contains n positive integers $w_1, w_2, \ldots, w_n$, where $1 \leq w_i \leq 10^6$ is the weight of the i-th crystal.

Output

Output a single integer: the minimum amount of magical energy required to make all weights equal.

Sample Input 1	Sample Output 1
5 10 6 35 105 42	8

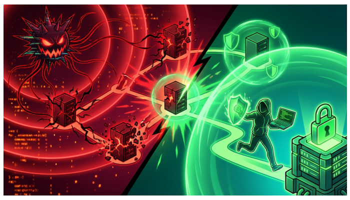
Sample Input 2	Sample Output 2
4 12 24 58 99	8

This page is intentionally left (almost) blank.

PROBLEM M - 100 Points

Network Protection

Time Limit: 1.0 s | *Memory Limit: 512 MiB*


Every year, analysts have to secure servers from malware.

The National Zero-Day Protection Challenge (NZPC) is an annual cyber-security exercise in which participants study how attackers and defenders move through computer networks. Every year, security analysts build a network consisting of n servers and m bidirectional communication links. The servers are numbered from 1 to n, and server i contains data of size c_i gigabytes. For each communication link m_i, there is a latency w_i associated with it, indicating the amount of time it takes to travel between the two connected servers. Note that multiple communication links can exist between the same pair of servers.

Suppose a malware outbreak has been detected at server a. Once compromised, the malware spreads through the network along communication links. Meanwhile, a security analyst maintains a secure backup server located at server b. For the analyst, a server u is considered **recoverable** if there exists a route from u to the backup server b such that for every server x on that route (including both u and b), the earliest possible time at which the malware can reach server x is **strictly greater** than the time required for the analyst to reach server x along the chosen route.

In other words, the analyst can always plan a route from recoverable server to the backup server such that the analyst arrives strictly before the malware along the chosen route.

To avoid any risk of data loss, it is decided that only recoverable servers can store data. The analyst wants to know the **maximum** amount of storage that can be safely protected by the backup server.

Input

The first line contains two integers $3 \leq n \leq 10^5$ and $1 \leq m \leq 10^5$, the number of servers and communication links.

The second line contains two integers a and b, representing the malware source server and the backup server. It is guaranteed that $a \neq b$ and $1 \leq a, b \leq n$.

The third line contains n integers $c_1, c_2, \ldots, c_n$, where c_i ($1 \leq c_i \leq 10^9$) is the size of data stored on server i in gigabytes.

The next m lines each contain three integers u_i, v_i, and w_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i, 1 \leq w_i \leq 10^9$), representing a bidirectional communication link between servers u_i and v_i with latency w_i milliseconds.

Output

Output a single integer: the total value of data stored on all recoverable servers.

Sample Input 1	Sample Output 1
5 5 1 2 1 2 4 8 16 1 2 4 2 3 3 3 4 1 2 5 2 3 1 8	22

Sample Input 2	Sample Output 2
6 10 3 4 1 1 1 1 1 1 1 2 6 2 3 3 3 1 4 3 4 5 4 5 8 5 6 2 6 4 1 3 2 4 5 4 4 3 5 6	3

PROBLEM N - 100 Points
Downsizing

Renowned biologist Nathaniel Zacharias Patrice Chalmers has found a solution to pest control in Kapiti island. Instead of trying to keep predators outside of the island, he is going to relocate all predators into Kapiti island! More precisely, he has found a way to teleport regions of land (both bounded and unbounded) to a bounded set of points inside a finite circular cell. He explains his process as follows.

Assume a circular cell of radius r is centred at a point O. Let P be any point not in the interior of the cell, and suppose the line $\overline{OP}$ intersects the cell's boundary at the point B. Then point P is teleported to a point P' lying on this line, where length$(\overline{OP}) \cdot$ length$(\overline{OP'}) = r^2$. (Points along the cell boundary are teleported to themselves.) Figure N.1 shows how a pentagonal enclosure is teleported to the shaded area within the circle; sample points O, P, P', and B are identified in the figure to illustrate the teleportation rule.

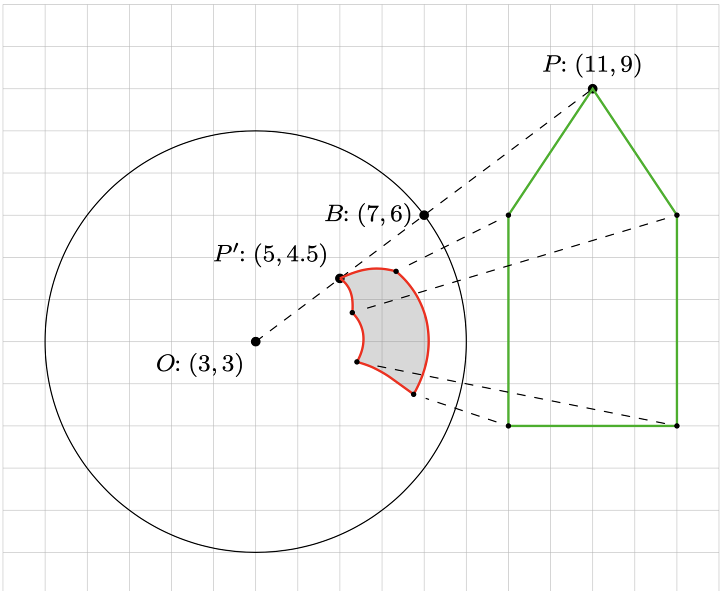


Figure N.1: Sample input 1, illustrating the downsizing process

Given a convex polygonal shape not containing any interior point of the cell, Nathaniel would like to know the area of the corresponding teleported shape. Can you help?

Input

The first line contains three integers x_0, y_0, and r ($-10^4 \le x_0, y_0 \le 10^4$, $1 \le r \le 10^4$), specifying the centre of the circular cell and its radius. (The entire circular cell lies within the specified bounds.)

The second line contains a positive integer n, $3 \le n \le 100$, followed by n pairs of integers x_1, y_1, x_2, y_2, $\ldots$, x_n, y_n, $-10^4 \le x_i, y_i \le 10^4$, giving the vertices of a convex polygon with n vertices in counterclockwise order.

Output

Output the area of the region of the cell corresponding to the polygonal region in the input. Your answer should be correct to a relative or absolute error of 10^{-6}. Refer to the contest instructions if you do not know what this means.

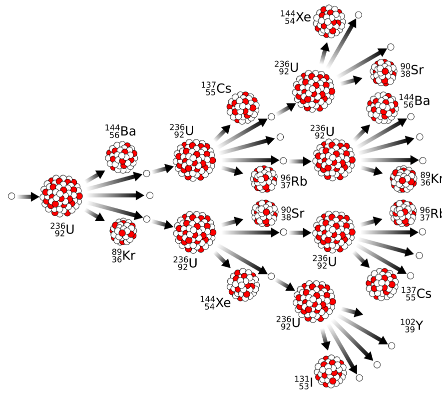
Sample Input 1	Sample Output 1
3 3 5 5 11 9 9 6 9 1 13 1 13 6	4.112904

PROBLEM O - 100 Points

Chain Reaction

Time Limit: 5 s | Memory Limit: 512 MiB

Professor X is the kind of scientist who treats the warning “please do not mix random chemicals” in the same way as C programmers treats compiler warnings. His laboratory is cluttered with bubbling flasks, and a collection of machines of questionable safety. Today, he is testing his latest breakthrough theory.



Nuclear fission of Uranium, a chain reaction.

According to X, all chemicals have a special phase number v associated with them. When two chemicals with identical phase numbers are brought into the same reaction line, they trigger a spectacular phenomenon known as the *Near-Zero Phase Cancellation* reaction, where all chemicals between the two matching chemicals are vaporised.

For instance, suppose X has a sequence of chemicals with phase numbers $1, 3, 1, 3$. If he places the chemicals from left to right, the third chemical will trigger a NZPC reaction with the first chemical and vaporise the first, second and third chemicals. The remaining chemical with phase number 3 will not trigger any further reactions, so it will be the only chemical that remains.

X is given a sequence A of n chemicals, where the phase number of the i-th chemical is a_i ($1 \leq i \leq n$). He then asks q queries. In the j-th query ($1 \leq j \leq q$), you are given an interval $[l_j, r_j]$. For each query, X places the chemicals in the interval $[l_j, r_j]$ from left to right. After all the NZPC reactions have occured, determine how many chemicals remain in the sequence.

Input

The first line contains a positive integer $1 \leq T \leq 5$, the number of test cases.

For each test case:

The first line contains a positive integer $1 \leq n \leq 10^5$, the length of the chemical sequence A.

The second line contains n positive integers $a_1, a_2, \ldots, a_n$, where $1 \leq a_i \leq 50$ is the phase number on the i-th chemical.

The third line contains a positive integer $1 \leq q \leq 10^5$, the number of queries.

The next q lines each contain two integers l_j and r_j ($1 \leq l_j \leq r_j \leq n$), describing a query.

Output

For each test case, output q lines.

The j-th line ($1 \leq j \leq q$) should contain a non-negative integer: the answer to the j-th query.

Explanation

For the first query of sample input 1, X has a sequence of chemicals with phase number $1, 2, 2$. Since the last two chemicals have the same phase number, they trigger a NZPC reaction and vaporise both of them. The remaining chemical has phase number 1, so the answer is 1.

For the second query, the NZPC reactions are as follows: $\{1\} \to \{1,2\} \to \{1,2,2\} \to \{1\} \to \{1,3\} \to \{1,3,1\} \to \{\} \to \{3\}$ So only the chemical with phase number 3 remains, and the answer is 1.

Sample Input 1	Sample Output 1
1 6 1 2 2 3 1 3 4 1 3 1 6 1 5 5 6	1 1 0 2

Sample Input 2	Sample Output 2
1 21 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 3 1 21 1 20 2 21	0 2 2

PROBLEM P - 100 Points
Terrain Forming

Time Limit: 2 s | Memory Limit: 512 MiB

Long ago, the Natural Zenith Preservation Council (NZPC) was entrusted with shaping a mountain ridge stretching from east to west. To ensure water flowed naturally across the landscape, the council constructed the ridge so that the elevations never decreased going from west to east. NZPC has placed various checkpoints along the ridge, and the recorded elevations at the checkpoints therefore satisfy

$$1 \leq a_1 \leq a_2 \leq \cdots \leq a_n.$$

Land forming

The council employed a peculiar terrain-forming technique. Rather than freely changing the height of a checkpoint, they could only reshape an *interior* checkpoint by reflecting its elevation about the average of its two neighbouring checkpoints. Specifically, for any $2 \leq i \leq n-1$, they could replace

$$a_i$$

with

$$a_{i-1} + a_{i+1} - a_i.$$

The first and last checkpoints are anchored to solid bedrock and cannot be modified.

NZPC wants to measure the smoothness of a terrain profile which is based on the variance of its elevations,

$$D = \frac{1}{n}\sum_{i=1}^{n}(a_i - \bar{a})^2,$$

where

$$\bar{a} = \frac{1}{n}\sum_{i=1}^{n} a_i$$

is the average elevation.

Since the variance may be small and difficult to compare, NZPC defines the smoothness as the variance D multiplied by n^2, where n is the number of checkpoints. You are tasked by NZPC to find the minimum smoothness after performing the terrain-forming operation any number of times.

Input

The first line contains a single integer n ($3 \leq n \leq 10^4$).

The second line contains n monotonically increasing integers $a_1, a_2, \ldots, a_n$, satisfying $1 \leq a_1 \leq a_2 \leq \cdots \leq a_n \leq 10^3$.

Output

Output a single integer: the minimum possible smoothness ($D \times n^2$) after performing the terrain-forming operation any number of times.

Explanation

For $(a_1, a_2, a_3, a_4) = (1, 2, 4, 6)$ (Sample 1), There are two additional sequences that can be obtained by performing the operation any number of times:

$$(1, 3, 4, 6). \quad \text{and} \quad (1, 3, 5, 6).$$

For $(a_1, a_2, a_3, a_4) = (1, 2, 4, 6)$, the mean is

$$\frac{13}{4},$$

and the variance is

$$\frac{1}{4}\left(\left(1 - \frac{13}{4}\right)^2 + \left(2 - \frac{13}{4}\right)^2 + \left(4 - \frac{13}{4}\right)^2 + \left(6 - \frac{13}{4}\right)^2\right) = \frac{59}{16}.$$

For $(a_1, a_2, a_3, a_4) = (1, 3, 4, 6)$, the mean is

$$\frac{7}{2},$$

and the variance is

$$\frac{1}{4}\left(\left(1 - \frac{7}{2}\right)^2 + \left(3 - \frac{7}{2}\right)^2 + \left(4 - \frac{7}{2}\right)^2 + \left(6 - \frac{7}{2}\right)^2\right) = \frac{13}{4}.$$

For $(a_1, a_2, a_3, a_4) = (1, 3, 5, 6)$, the mean is

$$\frac{15}{4},$$

and the variance is

$$\frac{1}{4}\left(\left(1 - \frac{15}{4}\right)^2 + \left(3 - \frac{15}{4}\right)^2 + \left(5 - \frac{15}{4}\right)^2 + \left(6 - \frac{15}{4}\right)^2\right) = \frac{59}{16}.$$

The minimum variance is $\frac{13}{4}$, and the minimum smoothness is $n^2 \times D = 16 \times \frac{13}{4} = 52$.

Sample Input 1	Sample Output 1
4 1 2 4 6	52