# The 18th Zhejiang Provincial Collegiate Programming Contest

April 17, 2021




## Problems

A League of Legends
B Restore Atlantis
C Cube
D Shortest Path Query
E Specially Super Rare
F Fair Distribution
G Wall Game
H Grammy and HearthStone
I Grammy and Ropes
J Grammy and Jewelry
K Grammy's Kingdom
L String Freshman
M Game Theory

*Do not open before the contest has started.*

Prepared by Zhejiang University

# Problem A. League of Legends

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

Grammy loves playing the worldwide famous game League of Legends.

In the latest version, there is a special rule that players can choose to play with. Under this rule, 10 players will be divided into Red team and Blue team. Each team consists of 5 players and every player has an initial HP value. If none of all 5 players from one team has positive HP value, this team will lose this game. In each turn, the team can choose a player and reduce his HP value by 1. Blue team will start first.

Now Grammy wants to know if both teams play in optimal strategy, which team will win the game.

## Input

The input contains only a single case.

There are 5 integers $B_1, B_2, B_3, B_4, B_5$ ($1 \le B_i \le 4 \times 10^8$) in the first line, indicating the HP value of each player in Blue team.

There are 5 integers $R_1, R_2, R_3, R_4, R_5$ ($1 \le R_i \le 4 \times 10^8$) in the second line, indicating the HP value of each player in Red team.

## Output

If Blue team will win, output "`Blue`" in one line. Otherwise, output "`Red`" in one line.

## Examples

| standard input | standard output |
|---|---|
| 1 1 2 3 4<br>2 4 1 5 3 | Red |
| 2 3 4 5 6<br>1 2 3 4 5 | Blue |

# Problem B. Restore Atlantis

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

There are $n$ ancient Greek maps describing the fabled islands Atlantis. The maps are labeled by $1, 2, \ldots, n$. The $i$-th map shows the rectangle area $R_i$ is a part of Atlantis. The sides of all rectangles are parallel to the axes. There may be multiple islands, and the rectangles may overlap.

Unfortunately, some maps are even unreliable so they will not be considered. You will be given $q$ queries. In the $i$-th query, you will be given two integers $s_i$ and $t_i$ ($1 \le s_i \le t_i \le n$). Please write a program to figure out the total area of Atlantis when all maps labeled by $k$ ($s_i \le k \le t_i$) are unreliable.

## Input

The input contains only a single case.

The first line of the input contains two integers $n$ and $q$ ($1 \le n, q \le 100\,000$), denoting the number of maps and the number of queries.

In the next $n$ lines, the $i$-th line contains four integers $xa_i$, $ya_i$, $xb_i$ and $yb_i$ ($0 \le xa_i < xb_i \le 2\,000$, $0 \le ya_i < yb_i \le 2\,000$), describing the $i$-th map $R_i$. $(xa_i, ya_i)$ is the southwest corner of $R_i$, and $(xb_i, yb_i)$ is the northeast corner of $R_i$.

In the next $q$ lines, the $i$-th line ($1 \le i \le q$) contains two integers $s_i$ and $t_i$ ($1 \le s_i \le t_i \le n$), describing the $i$-th query.

## Output

For each query, print a single line containing an integer, denoting the total area using the information of all reliable maps in this query.

## Example

| standard input | standard output |
|---|---|
| 3 6<br>10 10 20 20<br>12 12 22 22<br>15 15 25 25<br>1 1<br>2 2<br>3 3<br>1 2<br>2 3<br>1 3 | 151<br>175<br>136<br>100<br>100<br>0 |

# Problem C. Cube

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

You are given eight points in three-dimensional space, please check if they can form a cube.

A cube is a regular hexahedron, bounded by six square faces, with three meeting at each vertex.

## Input

The first line contains a single integer $T$ ($1 \leq T \leq 100$), denoting the number of test cases.

For each test case, each of the following eight lines containing three integers $x, y, z$ ($-100 \leq x, y, z \leq 100$), denoting the coordinates of the eight points, respectively.

## Output

For each test case, output a single line "`YES`" if the points can form a cube, or "`NO`" if they don't.

## Example

| standard input | standard output |
|---|---|
| 3<br>0 0 0<br>0 0 0<br>0 0 0<br>0 0 0<br>0 0 0<br>0 0 0<br>0 0 0<br>0 0 0<br>-4 -6 -2<br>5 9 10<br>12 -11 11<br>-9 4 8<br>-2 -16 9<br>7 -1 21<br>-7 -6 19<br>10 -1 0<br>0 0 0<br>2 2 1<br>0 0 1<br>0 2 0<br>2 0 0<br>0 2 1<br>2 0 1<br>2 2 0 | NO<br>YES<br>NO |

# Problem D. Shortest Path Query

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 512 megabytes |

There is an undirected graph with $n$ vertices and $m$ edges. The vertices are labelled by $1, 2, \ldots, n$. The $i$-th edge connects the $u_i$-th vertex and the $v_i$-th vertex, the length of which is $w_i$. Here, $u_i$'s binary representation is always a prefix of $v_i$'s binary representation. Both binary representations are considered without leading zeros. For example, $u_i = 2_{10} = \mathbf{10}_2$, $v_i = 5_{10} = \mathbf{10}1_2$.

You will be given $q$ queries. In the $i$-th query, you will be given two integers $s_i$ and $t_i$. Please write a program to figure out the length of the shortest path from the $s_i$-th vertex to the $t_i$-th vertex, or determine there is no path between them.

## Input

The input contains only a single case.

The first line of the input contains two integers $n$ and $m$ ($1 \le n \le 100\,000$, $1 \le m \le 200\,000$), denoting the number of vertices and the number of edges.

In the next $m$ lines, the $i$-th line ($1 \le i \le m$) contains three integers $u_i, v_i$ and $w_i$ ($1 \le u_i < v_i \le n$, $1 \le w_i \le 10^9$), describing the $i$-th edge. It is guaranteed that $u_i$'s binary representation is a prefix of $v_i$'s binary representation.

In the next line, there contains a single integer $q$ ($1 \le q \le 200\,000$), denoting the number of queries.

In the next $q$ lines, the $i$-th line ($1 \le i \le q$) contains two integers $s_i$ and $t_i$ ($1 \le s_i, t_i \le n$, $s_i \ne t_i$), describing the $i$-th query.

## Output

For each query, print a single line containing an integer, denoting the length of the shortest path. If there is no path, print "`-1`" instead.

## Examples

| standard input | standard output |
|---|---|
| 5 6<br>1 2 4<br>1 3 2<br>1 4 5<br>1 5 8<br>2 4 3<br>2 5 2<br>4<br>2 3<br>1 4<br>1 5<br>4 5 | 6<br>5<br>6<br>5 |
| 3 1<br>1 2 100<br>3<br>1 2<br>1 3<br>2 3 | 100<br>-1<br>-1 |

# Problem E. Specially Super Rare

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

A string is palindromic iff it reads the same right to left as left to right. For example "`abba`", "`zjcpcjz`" are palindromes. Chenjb loves palindromes, and he owns an extremely long palindrome string $S$.

Yesterday, RMB player Chenjb paid to open $m$ boxes in a popular mobile game. Every time he pays to open a box, he will get an item called "SSR" (Specially Super Rare) with the probability of 0.003%. Every time Chenjb got such an item, he would either delete a character or modify a character in his own string $S$ to celebrate his luck.

Today, Chenjb wants his string $S$ to be palindrome again by removing some characters from $S$. Please write a program to help him find the longest palindromic subsequence of the current string $S$.

## Input

The input contains only a single case.

The first line contains a string $S$ which consists of $n$ ($1 \leq n \leq 10^7$) lower-case English letters, denoting the current string.

The second line contains a single integer $m$ ($1 \leq m \leq 10^7$), denoting the number of boxes Chenjb opened yesterday.

## Output

Print a single line containing an integer, denoting the length of the longest subsequence of $S$.

## Example

| standard input | standard output |
|---|---|
| abadba<br>31274 | 5 |

# Problem F. Fair Distribution

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

There are $n$ robots and $m$ energy bars in the Dream Kingdom. DreamGrid, the king, is trying to make a fair distribution of the energy bars. A fair distribution exists if and only if the number of the energy bars is a multiple of the number of robots.

The only tool DreamGrid has is a powerful laser gun. Every time he turns on the laser gun, he can do exactly one of the two things:

- Create a new energy bar.
- Destroy a robot.

To avoid the extinction of robots, it's forbidden to destroy all the $n$ robots. It takes one dollar to turn on the laser gun once. You are asked to find the minimum cost of making a fair distribution.

## Input

There are multiple test cases. The first line of the input contains an integer $T$ ($1 \le T \le 1\,000$), indicating the number of test cases. For each test case:

The only line contains two integers $n$ and $m$ ($1 \le n, m \le 10^8$), indicating the initial number of robots and energy bars.

## Output

For each test case output one line containing an integer, indicating the minimum cost to get a fair distribution.

## Example

| standard input | standard output |
|---|---|
| 3<br>3 12<br>10 6<br>8 20 | 0<br>4<br>2 |

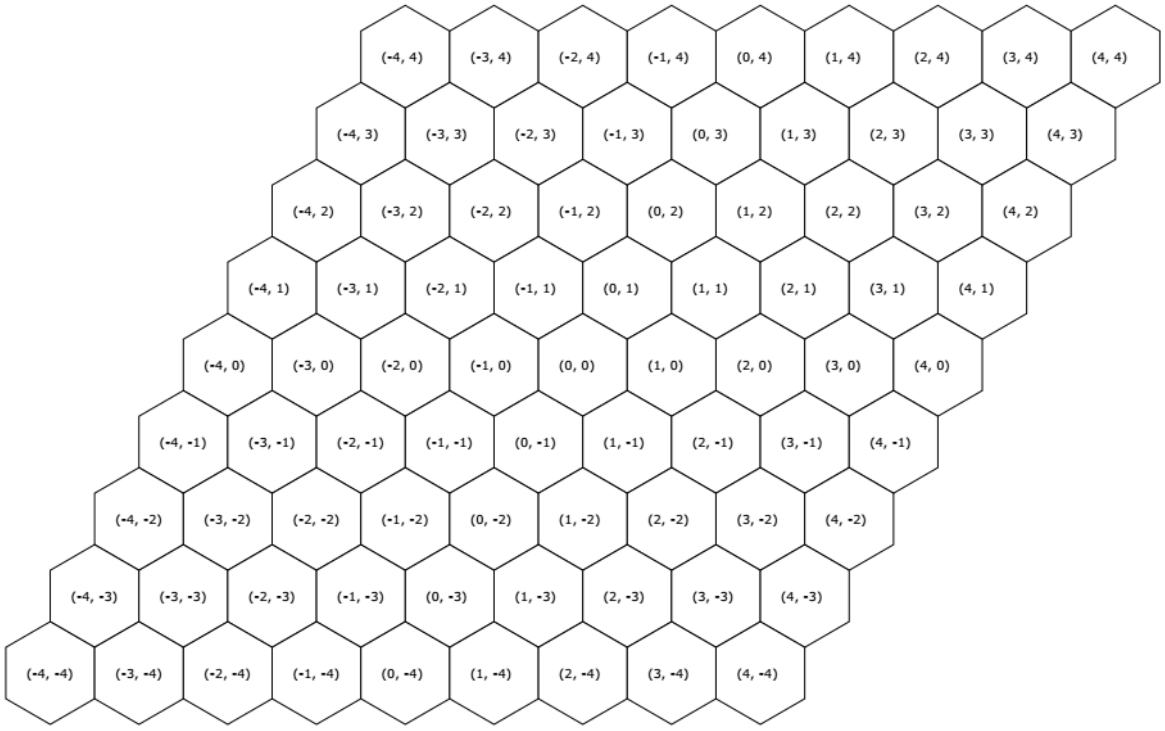
## Note

For the third sample, the best way is to destroy a robot and create an energy bar. After that, we have 7 robots and 21 energy bars, which leads to a fair distribution.

# Problem G. Wall Game

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 4 seconds |
| Memory limit: | 512 megabytes |

Recently, Grammy is playing a boring game called Wall Game.




There is a plane in the game, which is tiled with regular hexagons. There are honeycombs on this plane, which are oriented in such a manner that there are hexagon nodes below and above, and there are edges to the left and right which the honeycomb shares with its adjacent honeycombs in the row. Every consequent row is shifted relative to the previous row by half a honeycomb. The $Ox$ axis goes from left to right along the horizontal row of honeycombs. The $Oy$ axis is inclined 60 degrees relative to the $Ox$ axis. The axes intersect at the honeycomb with coordinates $(0, 0)$.

There are 2 operations attack and query. Grammy can conquer a honeycomb located at $(x_i, y_i)$ by an attack operation. For a query operation, Grammy wonders if she builds walls between the honeycombs she has conquered and the honeycombs she hasn't conquered, how many walls she can touch if she set out from honeycomb $(x_i, y_i)$ **in her territory** without crossing any walls.

Note: A person in honeycomb $A$ can go to honeycomb $B$ if and only if $A$ and $B$ share a common edge or there exists a honeycomb $C$ that $A$ can go to $C$ and $C$ can also go to $B$.

At the beginning of the game, no honeycombs are conquered. Please write a program to handle $n$ operations.

## Input

The input contains only a single case.

The first line contains a single integer $n$ ($1 \le n \le 500\,000$), representing the number of operations.

The following $n$ lines describe $n$ operations. Each line contains 3 integers $op_i, x_i$ and $y_i$ ($1 \le op_i \le 2$, $-10^9 \le x_i, y_i \le 10^9$), representing the type of operation and the coordinate of the honeycomb.

- $op_i = 1$ means an attack operation. Grammy will conquer honeycomb $(x_i, y_i)$.
- $op_i = 2$ means a query operation. In this operation, Grammy asks you how many walls she can touch if she transfers from honeycomb $(x_i, y_i)$ without crossing any walls.

**It is guaranteed that $(x_i, y_i)$ has not been conquered if $op_i = 1$, and $(x_i, y_i)$ has been conquered if $op_i = 2$.**

## Output

For each query operation, you should output an integer in a line representing the answer to Grammy's query.

## Example

| standard input | standard output |
|---|---|
| 8<br>1 0 0<br>2 0 0<br>1 0 2<br>1 1 2<br>1 0 3<br>2 0 3<br>1 0 1<br>2 0 0 | 6<br>12<br>20 |

# Problem H. Grammy and HearthStone

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

"HearthStone" is an online digital collectible card game developed and published by Blizzard Entertainment.

In the game, players have to summon minions and cast spells to attack their opponent.

The following 5 paragraphs illustrate the rules that you need to know in order to solve this problem.

Each minion has its attack value and its hitpoint value.

When a minion receives damage, its hitpoint value decreases by the damage value.

Whenever a minion has a non-positive hitpoint value, it dies.

Minions can have "Deathrattle($x/y$)" properties, that is, when this minion dies, another minion with $x$ attack and $y$ hitpoint will be summoned instantly. Additionally, the summoned minion does not have any "Deathrattle" property.

"Defile" is a spell. When cast, the spell will deal 1 damage to all minions, and if any minion dies, the spell will be automatically cast again.


Grammy is a famous player who ranked top $10^9$ among all "HearthStone" players. Nonetheless, she is unhappy because she does not know how to use "defile" just like the textbook. She wants to do some practice.

$n$ minions have been placed on the battlefield. The $i$-th of them have 0 attack, $10^9$ hitpoint, and "Deathrattle($i/i$)". Grammy can modify each minion's hitpoint to another positive value before she cast the last "Defile" in her hand, can you help her to find out a way to modify each minion's hitpoint so that the recasting effect of the spell will be triggered $2n$ times? In other words, the spell is cast $2n + 1$ times in total, manually or automatically.

## Input

The input contains only a single case.

The only line of the input contains a positive integer $n$ ($1 \le n \le 1\,000$), indicating the total number of minions on the battlefield.

## Output

If the solution exists, output $n$ integers in one line, the $i$-th one represents the hitpoint value of the $i$-th minion after Grammy's modification.

Otherwise output "-1" in one line.

If there are multiple solutions, output any.

## Examples

| standard input | standard output |
|---|---|
| 3 | -1 |
| 8 | 8 1 13 11 2 4 5 6 |

## Note

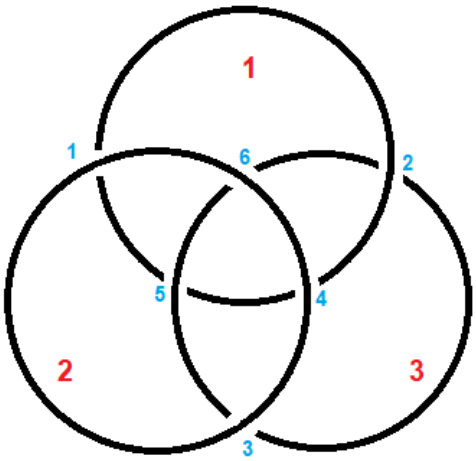
The rules of "HearthStone" in this problem might be different from the original game, so please read the statement carefully.

# Problem I. Grammy and Ropes

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

The original problem name is "Separation Judgement Problem of Rope-made 3 Cycle Venn Diagram with Overlapping Intersections", but it can't fit into the margin.

Three cyclic ropes are lying on the table, the projection of which forms a Venn Diagram.




As the illustration shows, these ropes (indexed from one to three) have six overlapping intersections, indexed from one to six respectively.

Grammy wants to pull the ropes apart, but it seems that they are tied together, so she needs to cut a subset of the three ropes with scissors. She is wondering how many different ways are there to choose the subset so that these ropes are separable afterward. Can you tell her the answer?

Note: Two subsets are different if and only if there is at least one rope being chosen in one of the subsets and not being chosen in the other.

## Input

The input contains only a single case.

The only line of the input consists of 6 boolean variables (either "`true`" or "`false`") in one line, the $i$-th one representing whether the larger-indexed rope is on top of the other or not at the $i$-th intersection.

## Output

One integer, the answer to the problem.

## Example

| standard input | standard output |
|---|---|
| `true false false true true false` | 6 |

# Problem J. Grammy and Jewelry

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 0.5 seconds |
| Memory limit: | 512 megabytes |

There is a connected undirected graph with $n$ vertices and $m$ edges. Vertices are indexed from 1 to $n$. There is infinite jewelry in vertex $i$ ($2 \le i \le n$), each valued $a_i$. Grammy starts at point 1. Going through each edge consumes 1 unit of time. She can pick up a piece of jewelry at vertex $i$ and put it down at vertex 1. Picking up and putting down a piece of jewelry can be done instantly. Additionally, she can carry at most 1 piece of jewelry at any time. When she put down a piece of jewelry valued $x$ at vertex 1, she obtains the value of it. Now, for each $k$ between 1 and $T$ (inclusive) she wonders what is the maximum value she can get in $k$ units of time.

## Input

The input contains only a single case.

The first line contains three integers $n$, $m$, and $T$ ($1 \le n, m, T \le 3\,000$).

The second line contains $n-1$ integers $a_2, a_3, \ldots, a_n$ ($1 \le a_i \le 3\,000$).

The following $m$ lines describe $m$ edges. Each line contains 2 integers $x_i$ and $y_i$ ($1 \le x_i, y_i \le n$), representing a bidirectional edge between vertex $x_i$ and vertex $y_i$.

Note that the graph may contain self-loops or duplicated edges.

## Output

Print $T$ integers in one line, the $k$-th ($1 \le k \le T$) of which denoting the maximum value she can get in $k$ units of time.

## Example

| standard input | standard output |
|---|---|
| 5 6 5<br>2 3 4 5<br>1 2<br>4 5<br>5 5<br>2 3<br>1 3<br>3 3 | 0 3 3 6 6 |

# Problem K. Grammy's Kingdom

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1.5 seconds |
| Memory limit: | 512 megabytes |

Grammy's kingdom is in danger. Some foreign invaders have intruded into her kingdom and they are destroying the traffic system. There are $n$ stations indexed from 1 to $n$ in the traffic system. Moreover, there are $m$ ($m \le n$) airports located in some of the stations. If you are in station $i$, you can transfer to station $i+1$ if $i$ and $i+1$ haven't been destroyed. If you are in station $i$ **with an airport**, you can fly to station $j$ **if** $i \le j$ **and station** $j$ **has an airport** and both of the stations haven't been destroyed. We define the stability $G$ of the system as the number of routes that still available.

Formally, $G = \sum_{1 \le i \le j \le n}$[a person in station $i$ can transfer to station $j$ via several stations or airports].

At each moment $i$ ($1 \le i \le n$), the invaders will randomly choose an undestroyed station $x$ and destroy it as well as the airport in it (If there exists an airport in it).

We define $E(x)$ as the expected value of $x$. Grammy wonders the value of $\sum_{i=1}^{n} E(G_i)$ modulo 998 244 353. Could you please help her? Here, $G_i$ denotes the value of the stability $G$ after the invaders destroying the $i$-th chosen station at the $i$-th moment.

## Input

The input contains only a single case.

The first line contains two positive integers $n$ and $m$ ($1 \le n \le 500\,000$, $0 \le m \le n$), denoting the number of stations and the number of airports.

The second line contains $m$ distinct integers $x_1, x_2, \ldots, x_m$ ($1 \le x_i \le n$), denoting the indexes of station that has an airport.

## Output

Output one integer, the value of $\sum_{i=1}^{n} E(G_i)$ modulo 998 244 353.

## Examples

| standard input | standard output |
|---|---|
| 1 0 | 0 |
| 3 3<br>1 2 3 | 4 |
| 6 2<br>2 4 | 16637434 |

## Note

It can be proved that the answer can be represented as a rational number $\frac{p}{q}$ with $\gcd(p, q) = 1$. Therefore, you are asked to find the value of $pq^{-1} \bmod 998\,244\,353$. It can be shown that $q \bmod 998\,244\,353 \neq 0$ under the given constraints of the problem.

# Problem L. String Freshman

| | |
|---|---|
| Input file: | standard input |
| Output file: | standard output |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

Chenjb is struggling with the string theory now. He is trying to solve a string problem. In that problem you will be given two non-empty strings $S$ and $T$, and you need to report the number of substrings in string $S$ which matches $T$. For example, $S =$"ababac", $T =$"aba", the answer will be 2 because $T$ appears twice in $S$: "[aba]bac", "ab[aba]c".

Chenjb is a freshman in programming contest. Let array `S[1..n]` denote the string $S$ of length $n$, and let array `T[1..m]` denote the string $T$ of length $m$. Chenjb wrote down the following C/C++ code using the straight-forward greedy strategy:

```
int Find_Answer() {
    int j = 1, ans = 0;
    for (int i = 1; i <= n; i++) {
        if (S[i] == T[j]) j++; else j = 1;
        if (j > m) {
            ans++;
            j = 1;
        }
    }
    return ans;
}
```

Chenjb submitted his code, and fortunately, got accepted. But soon Chenjb realized that his greedy algorithm is not always correct. For example, $S =$"aaaa", $T =$"aaa", the answer will be 2, but Chenjb's code will return 1.

You know, Chenjb is a freshman, so he turns to you for help. You will be given **the template string** $T$, your task is to determine whether there is a non-empty string $S$ such that Chenjb's code will not pass.

## Input

The input contains only a single case.

The first line of the input contains a single integer $m$ ($1 \le m \le 100\,000$), denoting the length of the template string $T$.

The second line contains a string $T$ which consists of $m$ lower-case English letters.

## Output

If there exists a non-empty string $S$ such that Chenjb's code will not pass, print "`Wrong Answer`", otherwise print "`Correct`".

## Examples

| standard input | standard output |
|---|---|
| 3<br>abc | Correct |
| 3<br>aaa | Wrong Answer |

# Problem M. Game Theory

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

Grammy is a CS professor at Sakuya Academy and she teaches Game Theory this semester.

Including Grammy herself, there are $n$ people in the class. Today, in order to attract students' interest, she decides to play a game with all students.

For each student, Grammy will pick an integer $x$ ($1 \le x \le 20$). Without knowing what Grammy picks, the student will also pick another integer $y$ ($1 \le y \le 20$). In the next step, Grammy calculate the score through the following procedure with each student independently.

- Grammy will give the student $x$ points.
- The student will give Grammy $y$ points.
- If Grammy's integer $x$ is strictly greater than $y$, then she will obtain 10 extra points from the student.
- If Grammy's integer $x$ is strictly less than $y$, then she will give 10 extra points to the student.

Now Grammy wants to know the expected amount of points she may win from all students if she chooses to pick the integer randomly and independently, which means for all integers in $[1, 20]$, they all share the same possibility. Since students are very clever, you may assume that they will follow the optimal strategy in this game to maximize their final score.

## Input

The input contains only a single case.

The only line of the input contains an integer $n$ ($1 \le n \le 1\,000$), indicating the total number of people in the class (Including Grammy).

## Output

Output the answer in one line. Your answer will be considered correct if and only if the absolute or relative error does not exceed $10^{-4}$.

## Example

| standard input | standard output |
|---|---|
| 1 | 0.0000 |