Programming League National 2020

A Number Base

Code Name	`numberbase`
Input	Standard Input
Output	Standard Output
Time Limit	1 second
Memory Limit	256 megabytes

Problem Statement

There are many different number systems used nowadays. The most common number base used by humans is base-10 which is known by **Decimal**.

Computers on the other hand use base-2 which works greatly with two electronic states `ON` and `OFF`. And there are many bases that are used for different tasks.

Now, your next programming task is to create a simple program that can convert between two number base systems.

Input Format

- The first line has one integer T which represent the number of test cases.
- The next T following lines include one string S, which is the string representation of the number you want to convert from, and two integers I and O, which are the input base and the output base respectively.

Constrains

- $1 \leq T \leq 100$
- $1 \leq base10(S) \leq 10^9$, $base10(x)$ = the base10 of x.
- $2 \leq I \leq 10$
- $2 \leq O \leq 10$

Output Format

For every test case T, output S in the base O form. If S does not belong to base I output **"INVALID"**.

Sample Input	Sample Output
4 8 10 8 101 2 10 90 9 10 354 10 2	10 5 INVALID 101100010

Programming League National 2020

B Kill Process

Code Name	`killprocess`
Input	Standard Input
Output	Standard Output
Time Limit	2 second
Memory Limit	512 megabytes

Problem Statement

Ahmad is a system administrator. He handles Linux. Currently he have some process to kill. Unfortunately he can only run one kill command. In Linux, if a process is killed, all its descendant process will be killed too, so he knows that to kill two process in one command, he need to kill a common ancestor process of the two process.

Obviously, killing any process is not a good idea, and there is a monitoring system in place to detect that situation. The system constantly measure the memory usage of the server. When a process is killed, its memory is freed, so when the memory usage suddenly drops, the monitoring system knows some process is dead.

Ahmad know that, so he need to pick the process that will reduce the memory usage the least, or he risk having to explain himself to his manager. Given several query for two process that Ahmad need to kill, determine the least amount of memory that will be freed considering that Ahmad can only kill one command.

Input Format

The input starts with two integer n, q where n is the number of processes and q is the number of queries.

The next n line each represent a process. Each line consist of three integer a, b, c, where a is the *pid* or process id, b is the *ppid* or parent process id, and c is the memory usage.

The next q line each represent a query. Each line consist of a two integers r, s which is the two process id that Ahmad need to kill.

Constrains

- $1 \leq n, q \leq 10^5$
- $0 \leq a, b, c \leq 10^5$
- $0 \leq r, s \leq 10^5, r \neq s$

Output Format

For each query output the minimum amount of memory that will be freed given that Ahmad kill only a single process to kill both r and s.

Notes

In the first example:

The first query wants to kill process 4 and 5. Ahmad can kill process 2 and thus, effectively killing 2, 4 and 5, freeing 30 memory. He can also kill process 1, but that would take 50 memory.

The second query need to kill 2 and 3. The only way to kill both this process is to kill process 1, freeing 50 memory.

The third query need to kill process 4 and 2. Ahmad can kill process 2 which would also kill process 4 and 5, freeing 30 memory.

Sample Input	Sample Output
5 3 1 0 10 2 1 10 3 1 10 4 2 10 5 2 10 4 5 2 3 4 2	30 50 30
5 3 1 0 12 2 1 22 4 1 92 100 4 98 101 4 92 4 100 2 4 2 1	282 316 316

C Cubes

Code Name	`cubes`
Input	Standard Input
Output	Standard Output
Time Limit	1 second
Memory Limit	256 megabytes

Problem Statement

You are given a task to build a cube model using only *plasticine balls* and *wooden ssticks*. The sticks are all in length of $1cm$ each yet you are required to build a cube model of $n \times n \times n$ cm^3.

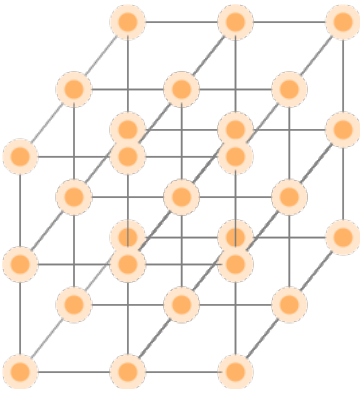


Figure on the right shows the example of how you can build a $2 \times 2 \times 2$ cm^3 cube using the materials given, the cube is built using 54 *wooden sticks* and 27 *plasticine balls*. You noticed that to build a larger cube requires a large number of materials and you need to have an exact amount of material needed to avoid any wasting.

Input Format

The first line in the input file is an integer representing the number of test cases, n, followed by n lines of integer k

Constrains

- $1 \leq n \leq 1000$
- $1 \leq k \leq 500$

Output Format

For each test case, print the number of *wooden sticks* needed followed by the number of *plasticine balls* needed separated by a `space`.

Sample Input	Sample Output
3 1 2 100	12 8 54 27 3060300 1030301

Programming League National 2020

D Rescue the hostage

Code Name	`hostage`
Input	Standard Input
Output	Standard Output
Time Limit	3 second
Memory Limit	256 megabytes

Problem Statement

Johnny is a soldier from the Kingdom of Wakanka. He always receives missions to rescue hostages. However, most of the time he is not prepare well and, in the end, *failed* the mission as there are a lot of traps on the way to rescue the hostage.

The King of Wakanka is finding someone who is brilliant in designing a solution for Johnny to go on a mission carefully. Usually Johnny will receive a map from the *scouts*, the map contains marks of traps, obstacles and the hostage. Your task is to write a program, deciding whether Johnny should bring more *defuse kits* so that he won't always failed the mission due to unforeseen traps or not.

The map is a 2-D array which each integer in the map represent a thing.

- Walkable space = "`.`"
- Johnny's initial location = "`@`"
- Hostage's location = "`%`"
- Obstacle = "`#`"
- Trap = "`*`"

There are few requirements as below:

1. Johnny **cannot** walk through the obstacle.
2. Johnny has been given N *defuse kits*, for each trap he walked into, he can defuse the trap by using one *defuse kit*, and the number of defuse kits will be left is $N-1$.
3. If Johnny is out of *defuse kits* (i.e. $N = 0$), and then he encountered a trap, he will not be able to rescue the hostage and thus failing the mission.
4. If Johnny cannot rescue the hostage, he will be fail the mission as well

Input Format

- The first line is number of test cases T.
- For each test case, the first line is number of *defuse kits*, N, height of the map, H, and width of the map, W.
- For the next H lines, there are W characters describing the map, which have been described in the Problem Description.

Constrains

- $1 \leq T \leq 100$
- $1 \leq N \leq 20$
- $2 \leq H, W \leq 10$
- only one hostage

Output Format

For each test case, display 1 if Johnny can complete the mission without requiring more *defuse kits*, 0 otherwise.

Sample Input	Sample Output
5 1 2 2 @. .% 2 3 3 @.* *#* ..% 2 3 3 @.* *#* .#% 1 3 3 @*. *#* .#% 2 3 3 @.. .## .#%	1 1 1 0 0

Programming League National 2020

E Drawing Shapes

Code Name	`shape`
Input	Standard Input
Output	Standard Output
Time Limit	3 second
Memory Limit	1024 megabytes

Problem Statement

You are developing an A.I. application that can identify drawings. In the meantime you are making a *prototype* for the application, in the *prototype*, the user is going to place four points on the screen, the *prototype* then is going to connect the four points and guess the shape of the resulting *quadrilateral*.

You are given the four points, no three of them are collinear, you have to output the simple polygonal shape formed by these points in the following order:

```
Square
Rectangle
Rhombus
Parallelogram
Trapezium
Ordinary Quadrilateral
```

For example if it is possible to form a *square* with the four points you must output **'Square'**, if it is not possible to form a *square* but possible to form a *rectangle* you must output **'Rectangle'** and so on.

Input Format

- Input starts with an integer T, the number of test cases.
- Each test case contains (4) lines. Each of the lines contains two space separated integers x_i and yi which are the coordinate values of a point.

Constrains

- $1 \leq T \leq 50000$
- $-10000 \leq x_i, y_i \leq 10000$

Output Format

For each set of input, output *one* line which is the *shape* that can be constructed from the points as described above. See sample input output for more details.

Note: If you have forgotten elementary geometry, here is the definitions to remind you:

- **Square:** All sides are of equal size all angles are 90°.
- **Rectangle:** Opposite sides are of equal size and all angles are 90°.
- **Rhombus:** All sides are of equal size but no angle is 90°.
- **Parallelogram:** Opposite sides are of equal size but no angle is 90°.
- **Trapezium:** Any two opposite sides are parallel but the other two are not.
- **Ordinary Quadrilateral=Simple Polygon**: Polygon having no self intersecting edge.

Sample Input	Sample Output
6 0 0 2 0 2 2 0 2 0 0 3 0 3 2 0 2 0 0 8 4 5 0 3 4 0 0 2 0 3 2 1 2 0 0 5 0 4 3 1 3 0 0 5 0 4 3 1 4	Square Rectangle Rhombus Parallelogram Trapezium Ordinary Quadrilateral

F Word Transformation

Code Name	`words`
Input	Standard Input
Output	Standard Output
Time Limit	3 second
Memory Limit	1024 megabytes

Problem Statement

A common word puzzle found in many newspapers and magazines is the word transformation. By taking a starting word and successively altering a single letter to make a new word, one can build a sequence of words which changes the original word to a given end word. For instance, the word **"spice"** can be transformed in four steps to the word **"stock"** according to the following sequence: `spice`, `slice`, `slick`, `stick`, `stock`. Each successive word differs from the previous word in only a single character position while the word length remains the same.

Given a dictionary of words from which to make transformations, plus a list of starting and ending words, your task is to write a program to determine the number of steps in the shortest possible transformation.

Input Format

- The first line of the input is an integer T, indicating the number of test cases.
- Each test case will start with integer N, which is the number of words in the dictionary.
- The next N lines contain one word s_i per line which is one word of the dictionary. Words can appear in the dictionary in any order.
- Following the dictionary is an integer M which is the number of queries.
- M lines follow in which there is a pair of words separated by space, these pairs represent the starting and ending words in a transformation. All pairs are guaranteed to have a transformation using the dictionary given. The starting and ending words will appear in the dictionary.

Constrains

- $1 \leq T \leq 100$
- $1 \leq N \leq 200$
- $1 \leq s_i \leq 10$
- $1 \leq M \leq N$

Output Format

The output should contain one line per word pair for each test set, and must include the starting word, the ending word, and the number of steps in the shortest possible transformation, separated by single spaces.

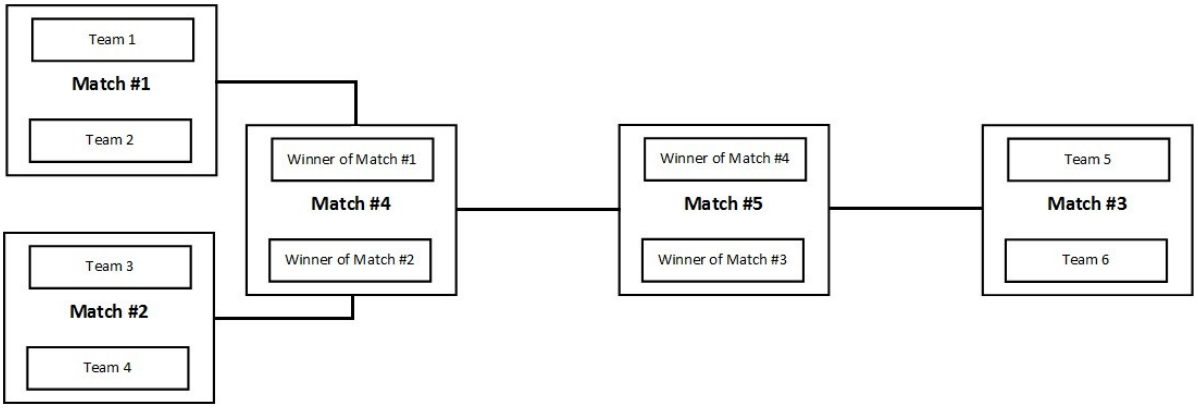
Sample Input	Sample Output
1 17 dip lip mad map maple may pad pip pod pop sap sip slice slick spice stick stock 2 spice stock may pod	spice stock 4 may pod 3

G	The League	
	Code Name	`league`
	Input	Standard Input
	Output	Standard Output
	Time Limit	2 second
	Memory Limit	256 megabytes

Problem Statement

In a knockout stage of a football league, the loser of a match is out of the league and the winner advances to play with another winner of a different match. This method is adopted in the FIFA World Cup and many other leagues around the world. There are no draws in this stage, the game continues until a winner is decided (whether in the extra time or by shots from the penalty mark).

For the purposes of this problem, a league can contain any number of teams. Your task is to organize the matches between the teams to decide a winner while minimizing the number of matches. For instance, if a league has 6 teams, the figure below shows one way to organize the matches resulting in 5 matches to decide a winner of the league.



Given the number of teams in a league, your task is to determine the minimum number of matches required to decide a winner.

Input Format

- The first line contains an integer N indicating the number of test cases.
- Each test case consists of one line containing an integer T representing the number of teams in the league.

Constrains

- $1 \leq N \leq 10000$
- $1 \leq T \leq 10^9$

Output Format

Output a single integer M, which is the minimum number of matches required.

Sample Input	Sample Output
2 6 8	5 7

Programming League National 2020

H Pacman

Code Name	`pacman`
Input	Standard Input
Output	Standard Output
Time Limit	3 second
Memory Limit	512 megabytes

Problem Statement

In this problem, you will be given a 2D matrix with size $n \times m$. You are playing as *Pacman* and your task is to reach the point (n, m) from $(1, 1)$.

On each cell of this map there are a number of *coins* c_{ij} where i is the row number and j is the column number. When *Pacman* moves over the cell, it can collect the *coins*. Note that *Pacman* can only move to *right* and *down* positions.

Your task is to calculate the maximum number of coins *Pacman* can earn when reaching the point (n, m).

Input Format

- First line of input contains an integer T that denotes the number of test cases.
- Next line contains two integers n and m separated by space.
- Next n lines contains m integers c_{ij} separated by space which denotes the *coin value* of that cell.

Constrains

- $1 \leq T \leq 100$
- $1 \leq n, m \leq 100$
- $1 \leq c_{ij} \leq 1000$

Output Format

For each set of inputs, print a single integer in a single line, which is the maximum coin values that Pacman can earn.

Sample Input	Sample Output
2 5 6 1 6 2 3 4 5 9 1 2 1 8 6 2 5 8 6 10 5 8 5 9 4 2 8 23 5 8 5 9 6 2 3 1 5 3 4 2 6	76 15

Programming League National 2020

Stable Braces

Code Name	`braces`
Input	Standard Input
Output	Standard Output
Time Limit	3 second
Memory Limit	256 megabytes

Problem Statement

You're given a non-empty string made in its entirety from opening and closing braces. Your task is to find the minimum number of *"operations"* needed to make the string stable. The definition for being stable is as follows:

1. An empty string is stable.
2. If S is stable, then $\{S\}$ is also stable.
3. If S and T are both stable, then ST (the concatenation of the two) is also stable.

All of these strings are stable: {}, {}{}, and {{}{}} ; But none of these: }{, {{}{, nor {}{.

The only *operation* allowed on the string is to replace an opening brace with a closing brace, or visa-versa.

Input Format

- The first line contains the number T, the number of test cases.
- Each of T lines afterward has a non-empty string B composed of opening and closing braces and nothing else.

Constrains

- $1 \leq T \leq 1000$
- $2 \leq |B| \leq 2000$; $|B|$ is *even*.

Output Format

For each test case print a line which contains a minimum number of *operations* needed to convert the given string into a Stable one.

Sample Input	Sample Output
3 }{ {}{}{} {{{}	2 0 1

J

Candy Distribution

Code Name	`candy`
Input	Standard Input
Output	Standard Output
Time Limit	2 second
Memory Limit	512 megabytes

Problem Statement

Hanna is a kindergarten teacher. She wants to give some candies to the children in her class. All the children sit in a line and each of them has a rating score according to his or her performance in the class. Hanna wants to give at least 1 candy to each child. If two children sit next to each other, then the one with the higher rating must get more candies. Hanna wants to minimize the total number of candies she must buy.

For example, assume her students' ratings are $[4, 6, 4, 5, 6, 2]$. She gives the students candy in the following minimal amounts: $[1, 2, 1, 2, 3, 1]$. She must buy a minimum of 10 candies.

Hanna knows that you are a good problem solver, and asks for your help.

Input Format

- The first line contains an integer, n, the number of children.
- Each of the next n lines contains an integer s_i indicating the rating of the student at position i.

Constrains

- $1 \leq n \leq 10^5$
- $1 \leq s_i \leq 10^5$

Output Format

Output a single line containing the minimum number of candies Hanna must buy

Sample Input	Sample Output
3 1 2 2	4