Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen (LEA)
Prof. Dr. Ernst W. Mayr
Christian Müller, Philipp Hoffmann, Chris Pinkau, Stefan Toman

Summer Term 2015
Problem Set 5
21.05.2015

Algorithms for Programming Contests

This problem set is due by

Thursday, 28.05.2015, 6:00 a.m.

Try to solve all the problems and submit them at

`http://judge.informatik.tu-muenchen.de/`

This week's problems are:

The following amount of points will be awarded for solving the problems.

Problem	SS15N05A	SS15N05B	SS15N05C	SS15N05D	SS15N05E
Difficulty	easy	easy	medium	medium	hard
Points	4	4	6	6	8

If the judge does not accept your solution but you are sure you solved it correctly, use the "request clarification" option. In your request, include:

- the name of the problem (by selecting it in the subject field)
- a verbose description of your approach to solve the problem
- the time you submitted the solution we should judge

We will check your submission and award you half the points if there is only a minor flaw in your code.

If you have any questions please ask by using the judge's clarification form.

SS15N05A Contact List

Author: Christian Müller

A few days ago, Lea experienced one of the horrors of modern life: She dropped her smartphone. Now, her screen is cracked and sometimes random locations on the screen act as if they had just been pressed. When sending a message to one of her contacts, she enters the name of the contact into a searchbox. If the name matches exactly, she can send the message with just another click. However, now that her screen is cracked, this means that sometimes her phone already sends the message to "Bob", while Lea meant for it to be sent to "Bobby", which are totally different people. This has embarassed Lea quite a few times now, so she wants to rename some of her contacts such that no contact is a prefix of another one. Can you tell her how many contacts she has to rename?

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a blank line.

Each test case consists of an integer n, the amount of contacts Lea has in her phone. n lines follow, each line containing the name of a contact (where the first letter is in "A" to "Z" and the rest is in "a" to "z").

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x is the minimal amount of contacts Lea has to rename. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 10000$
- Contact names are unique.
- Contact names are not longer than 500 characters

Sample Data

Input

```
1
7
Bob
Bobby
Boba
Charles
Charly
Julia
Julian
```

Output

```
Case #1: 2
```

SS15N05B Cable Car

Author: Christian Müller

Lea is a great fan of wintersports. She always follows the winter olympics on TV and just loves to go skiing herself. This year, she booked a room in an expensive hotel in a very exclusive ski resort called "Slippery Slopes and Hills". One evening at the Après-Ski-Party, she met an interesting man - the architect who planned all the cable cars taking the tourists up the mountain. They talked for a bit and he described his latest problem to her.

The ski resort is trying to build a new cable car up a glacier. Through complicated computation, they even found out exactly how many posts are needed to support the cable car. Since glaciers move (albeit really slowly), the individual posts supporting the cable car have to be spaced as far apart from each other as possible. Additionally, the cable car should also span a canyon in the middle of the route. Now the architect is hard at work, trying to figure how to place the posts. Lea, who is always on the lookout for interesting problems, tells you about it. Can you help the architect?

Input

The first line of the input contains an integer t. t test cases follow.

Each test case consists of a single line of four integers d, p, u, and v, where d is the length of the route (going from 0 to d) of the cable car, p is the amount of posts that should be placed, u is the beginning point of the canyon and v the end point. Posts may be placed anywhere between 0 and d, i.e. exactly on 0, d, u, and v, but not in between u and v.

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x is the maximal minimum distance between two posts that can be achieved with an absolute error of up to 10^{-4}. This means the maximum x such that the architect can place all the posts and no two posts are less than x apart. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 20$
- $1 \leq d \leq 1000000$
- $2 \leq p \leq 2000000$
- $0 \leq u \leq v \leq d$

Sample Data

Input

```
4
2 3 1 2
3 3 0 1
9 10 5 6
9 10 5 7
```

Output

```
Case #1: 1.0000000009
Case #2: 1.5000000007
Case #3: 1.0000000001
Case #4: 0.8333333338
```

SS15N05C Warp Speed Ahead

Author: Chris Pinkau

The media always creates such a fuzz and hype when it comes to big scientific results. Of course, an opportunity for colonising space and travelling into other star systems are great news for mankind, but the word *theoretically* is often overlooked (and missing in headline news). Trade routes have already been computed, all the seats on the next thousand space flights have been booked, the names for the colonies and planets have been announced, but there is not even a single space ship capable of interstellar flight. The work on the new propulsion engine takes a few more years, there are many more simulations that have to be done. So, after Lea's last endeavour, Dr. S. Pace recruits her skills once more to help him to run some simulations on the new engine **FAST** (**F**ast **A**cceleration for **S**pace **T**ravel). Its concept is very similar to an atomic bomb, but much less devastating, it relies on a particle chain reaction as well. To start the engine, a number of high energy particles are injected into the engine's combustion chamber, in each time step they react with each other and create new high energy particles, that react again and so on. This goes on for a number of steps, then the created energy is converted by the engine to push the space ship forward. To speed up the propulsion even more, after some predetermined interval of time steps, a new load of high energy particles is injected into the engine. At the end of the simulation, the overall propulsion is measured by the total number of high energy particles. Can you help Lea simulate the new **FAST** engine?

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a line break.

Each test case contains five integers, n k N m x, the number of high energy particles at the beginning n, the factor by which the number of particles grows in one time step k, the total number of time steps N, and m and x that describe that after every m time steps x new particles are injected into the engine.

Output

For each test case, output one line containing "Case #i: y" where i is its number, starting at 1, and y is the propulsion the engine creates. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 2^{40}$
- $1 \leq k \leq 1000$

- $1 \leq N \leq 1000$
- $1 \leq m \leq N$
- $1 \leq x \leq 2^{40}$
- If $m \mid N$, then, in the last time step, new particles are injected before the overall propulsion is measured.

Sample Data

Input

```
2
5 2 8 3 1
10 4 4 2 10
```

Output

```
Case #1: 1316
Case #2: 2730
```

SS15N05D Game Show

Author: Chris Pinkau

Who does not know "Fools do Anything for Loads of Cash", the famous game show about cupcakes and telephone poles. Lea definitely does. She has even been in the audience several times, but has never had the chance to participate in it. Until now! She received an invitation yesterday and almost could not believe that after numerous attempts with copious amounts of letters, emails, telephone interviews, meetings, letters again, secretive dark alley meetings, and some more letters, she finally was invited. A few days more, and she would have totally bribed the game show host. The show consists of several games, where a win in the first game is awarded with r points, r^2 points in the second, r^3 points in the third, and so on. Although most games are not known at all, Lea feels confident about being an all-rounder and is assure that she has a certain chance to win any game. There is only one game where Lea is sure about: the usual task in the first game is for the candidate to calculate the maximal number of points that can be won in the whole show. And because Lea is fully occupied with training for the other games, she wants you to help her with the first game.

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a line break.

Each test case consists of three integers n p and q, where n is the number of games that will be played, and $\frac{p}{q} =: r$ is the number of points awarded for a win in the first game.

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x is the maximal number of points that can be achieved. The points should be printed as a simplified rational number in the format "numerator/denominator". Simplified means that the numerator and denominator should not have a common divisor bigger than one and should not be negative. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 50$
- $1 \leq n \leq 350$
- $1 \leq p, q \leq 10^5$

Sample Data

Input

```
3
5 4 3
3 2 5
4 49 7
```

Output

```
Case #1: 3124/243
Case #2: 78/125
Case #3: 2800/1
```

SS15N05E Ghost

Author: Christian Müller

One lazy afternoon, Lea and her friend Bea sit at home. They are both bored out of their minds, so they decide to play a game to pass the time. Since both are too lazy to get up from the sofa and get a boardgame from the cupboard, they settle on the game "Ghost". "Ghost" is a simple word game, where Lea and Bea take turns announcing a single letter to build a word fragment. The rules are as follows: The announced letters have to form the beginning of a valid word. However, the player who completes a word loses.

So for example if Lea starts with "P" and Bea replies with "i", then Lea would lose if she said "g" or "e". She could, however, say "z" (starting to spell "Pizza", for example) and thus continue the game. However, if Lea (or Bea, respectively) did not know any word starting with "Piy", then announcing "y" would be against the rules.

Knowing this day would come, Lea has already prepared a list of all the words she knows. Smugly, she assumes that Bea does not know any words she does not know herself. Now, Lea and Bea play a few rounds culminating in a "final showdown" round. The loser of the last round has to get up and get a boardgame from the cupboard. Bea is also quite skilled at the game, so Lea wants to know if she can win against Bea even if Bea plays perfectly.

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a blank line.

Each test case consists of two integers n and w, with n being the amount of rounds played and w being the amount of words Lea knows. w lines follow, each containing a word s, consisting of lowercase letters from "a" to "z".

Output

For each test case, output one line containing "Case #i:" where i is its number, starting at 1. Then output four lines, corresponding to four different scenarios. In the i-th line, print "victory", if Lea can win the last round n according to the i-th scenario, and "defeat" otherwise.

- Scenario 1: Lea begins in round 1. The winner of round k begins in the next round $k + 1$.
- Scenario 2: Lea begins in round 1. The loser of round k begins in the next round $k + 1$.
- Scenario 3: Bea begins in round 1. The winner of round k begins in the next round $k + 1$.
- Scenario 4: Bea begins in round 1. The loser of round k begins in the next round $k + 1$.

Constraints

- $1 \leq t \leq 50$
- $1 \leq n \leq 1000$
- $1 \leq w \leq 10000$
- $1 \leq |s| \leq 300$

Sample Data

Input

```
1
2 5
chilling
wicked
haunting
spooky
fortress
```

Output

```
Case #1:
victory
defeat
defeat
victory
```