Google Royale

Problem

While visiting the planet Theta VIII, your team of space explorers is forced to participate in the plot of a badly-written book, which takes place in a hotel/casino called the *Google Royale*. In order to escape the Royale, you will have to make enough money from gambling that you can buy the hotel for **V** dollars and leave.

You start with **A** dollars, and you will participate in *betting rounds* until one of two conditions is met. If you finish any betting round with ≤ 0 dollars, you will lose; if you finish a betting round with ≥ **V** dollars, you will buy the hotel and leave. Otherwise you'll keep starting new betting rounds.

Each betting round consists of one or more coin flips. If you have X dollars at the start of the round, you can choose any integer `B` between 1 and `min(X, M)` to bet on the first coin flip.

With probability 50%, you win the coin flip, and the Royale immediately pays you `B` dollars. You now have `X + B` dollars, and the betting round ends.

With probability 50%, you lose the coin flip and owe the Royale `B` dollars. You can now pay the `B` dollars you owe and end the round. Or if `2B ≤ M`, you can instead delay the payment and do a second coin flip with twice the bet: `2B` dollars. If you lose again, then you owe the Royale `B+2B=3B` dollars. You can continue doubling your bet in this way to `4B`, `8B`, etc., until either you win a coin flip, you choose to stop, or your next bet would exceed **M**. You can even continue if the total of all your bets in the current betting round exceeds X.

Once the round is over, you must pay the Royale for each coin flip you lost, and if you won a coin flip, the Royale pays you for that. For example, if you start with a bet of 1 dollars, lose three coin flips, and then win one, you would gain $8 - $4 - $2 - $1 = $1. If you lose three coin flips and then stopped, you would lose $4 + $2 + $1 = $7. If you are left with $0 or less after paying, then you are broke, and you have just lost the game.

Luckily you have brought an android with you, and he is able to compute the probability that you will win if you follow an optimal strategy. What is that probability, and what is the largest possible first bet you could make to have that probability? Remember that you are not allowed to bet more than **M**!

Example

Suppose that you decide to use the following (sub-optimal) strategy. You have **A**=5 dollars; **M**=20 and V=40. The following sequence of events is possible:

- Round 1: You can start by betting 1, 2, 3, 4 or 5 dollars. You decide to begin a betting round by betting 2 dollars.
    - Step 1 (B=$2): You win the first coin flip. You gain 2 dollars, and the betting round ends. Now you have 7 dollars.
- Round 2: You begin a betting round by betting 5 dollars.
    - Step 1 (B=$5): You lose the first coin flip. Now you owe the Royale 5 dollars. Since `5*2 ≤ 20`, you may do another coin flip with a bet of `5*2=10` dollars. You choose not to. You lose 5 dollars, and the betting round ends. Now you have 2 dollars.
- Round 3: You begin a betting round by betting 2 dollars.

- Step 1 (B=$2): You lose. Now you owe the Royale 2 dollars. You choose to flip another coin with a bet of 4 dollars.
- Step 2 (B=$4): You lose. Now you owe the Royale a total of 6 dollars. That's more than you have, which is okay. You choose to flip another coin with a bet of 8 dollars.
- Step 3 (B=$8): You win. You gain 8 dollars, pay the 2+4=6 dollars you owe, and the betting round ends. Now you have 4 dollars.
- Round 4: You begin a betting round by betting 2 dollars.
  - Step 1 (B=$2): You lose. Now you owe the Royale 2 dollars. You choose to flip another coin with a bet of 4 dollars.
  - Step 2 (B=$4): You lose. Now you owe the Royale a total of 6 dollars. You choose to flip another coin with a bet of 8 dollars.
  - Step 3 (B=$8): You lose. Now you owe the Royale a total of 14 dollars. You choose to flip another coin with a bet of 16 dollars.
  - Step 4 (B=$16): You lose. Now you owe the Royale a total of 30 dollars. Since `2*16>`**M**, you cannot flip another coin and you must pay what you owe. Now you have -26 dollars; you have lost.

Input

The first line of the input gives the number of test cases, **T**. **T** lines follow. Each line contains three integers separated by single spaces: **A**, **M** and **V**, in that order.

Output

For each test case, output one line containing "Case #x: y z", where **x** is the case number (starting from 1); **y** is the probability of winning if you follow an optimal strategy; and **z** is the maximum first bet you can make without reducing your probability of winning. **y** must be correct to within an absolute or relative error of 10^{-6}.

Limits

$1 \leq$ **T** ≤ 100.
Memory limit: 1GB.

Small dataset (Test set 1 - Visible)

$1 \leq$ **M** ≤ 20.
$1 \leq$ **A** $<$ **V** ≤ 20.
Time limit: 30 seconds.

Large dataset (Test set 2 - Hidden)

$1 \leq$ **M** $\leq 10^{16}$.
$1 \leq$ **A** $<$ **V** $\leq 10^{16}$.
Time limit: 60 seconds.

Sample

Sample Input	Sample Output
4 1 1 3 3 6 12	Case #1: 0.333333333 1 Case #2: 0.500000000 3

```
4 20 15
13 6 20
```

```
Case #3: 0.755555555 3
Case #4: 0.730769231 6
```