```
        if(nothing && fabs(A.arr[i][A.m]) > eps)
            return -1;
    }

    for(int i = A.m-1; i>=0; i--)
    {
        if(where[i] == -1)  return inf;
        int row = where[i];
        double sltn = A.arr[row][A.m];
        for(int j = i+1; j<A.m; j++)
            sltn -= A.arr[row][j]*X[j];
        X[i] = sltn/A.arr[row][i];
    }

    return 1;
}
```

Hackenbush:

Almost Everything About Hackenbush

Green Hackenbush And Colon Principle:

Game Rules:

1. Players move in alternating turns, and both players always move optimally.
2. During each move, a player removes an edge from the tree, disconnecting one of its leaves or branches. The leaf or branch that was disconnected from the rooted tree is removed from the game.
3. The first player to be unable to make a move loses the game.

Solution:

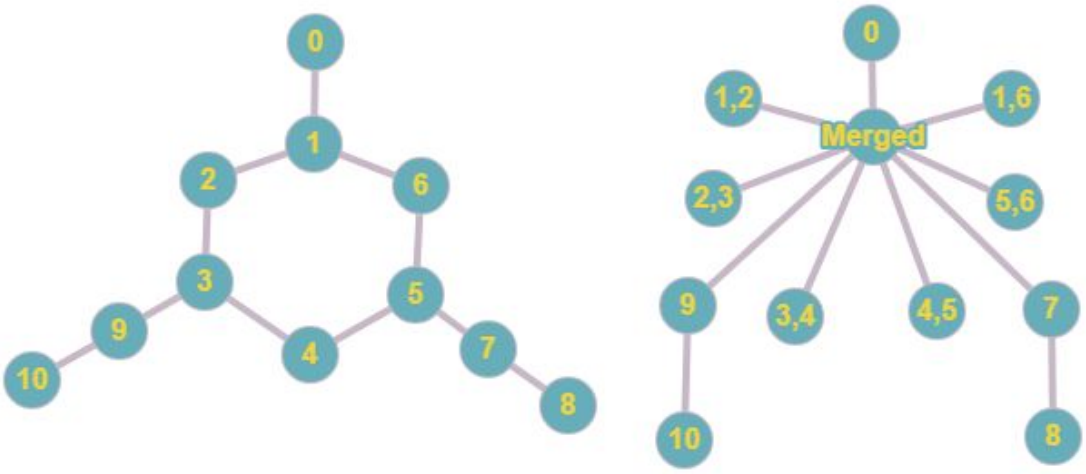
1. A single stalk is like a chain.
2. Size of a stalk is the number of edges in it.
3. A single stalk can be considered as a nim pile.
4. All kind of trees can be transformed to a single stalk using colon principle.
5. Colon Principle:
   If node u has 3 children a, b, c. then

stalkSize(u) = (stalkSize(a) + 1) ^ (stalkSize(b) + 1) ^ (stalkSize(c) + 1)

6. Look at the nim pile size of the stalk obtained from the tree and decide who wins the game.

Fusion Principle:

The Fusion Principle states that you can fuse all the nodes in any cycle of a Green Hackenbush and get a tree structure.



Node 1,2,3,4,5,6 are in a cycle cycle, they are merged to a single node. For every edge in the cycle a new node arises. Thus, we can get a tree from any graph and apply colon principle there.

If number of edges in the cycle is even, then the cycle can be replaced by a single node. If
number of edges in the cycle is odd, then the cycle can be replaced by a single special node with an additional edge. In both cases all the other edges of the nodes that just got fused, get attached to the new node.

Blue Red Stalk:

1. Blue player moves first
2. Blue player only cuts blue edges, Red player only cuts red edges. Substalk or subtree which has no attachment to the ground/root gets eliminated just like green hackenbush.
3. First player who cannot move loses.

Solution:

Blue is positive, Red is negative.

Until the color changes for the first time, each edge is worth +V or –V (depending on whether it is blue or red, respectively). Once color change occurs, each subsequent edge (regardless of being blue or red), is worth half of the previous value, with a +/– corresponding to the color.

For example: BBBRB = +V +V +V -V/2 +V/4. Now sum them to get value of each stalk.

For multiple stalk, instead of using XOR, we add them normally. If positive then Blue wins and if negative Red wins.

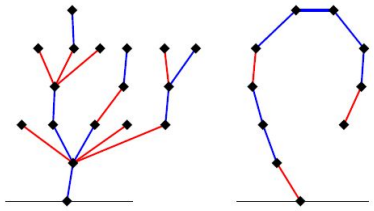
Blue Red Hackenbush Tree:

Same as blue red stalk, but with a tree.

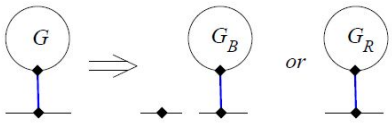
We can convert a tree to a stalk. Keep reading with patience.

11.1 Stalks and Trees

A "tree" is a connected position where there is a unique path from any node to the "ground" node. In other words, in a tree, there are no loops. A "stalk" is a very special kind of tree where there are no branches: every node has at most two segments connected to it. In both a tree and a stalk, nodes can be colored arbitrarily. In the figure below, the position on the left is a tree and the position on the right is a stalk (and also a tree, of course).



It turns out that there are some nice recursive methods to calculate the value of a tree or a stalk, and those recursive methods can sometimes be used to help calculate the values of more complex positions. The idea is based on the following theorem based on the figure below:



If G is some Red-Blue Hackenbush position having value x (for example, the position on the far left of the figure above), then if the grounded segments of G are connected instead to a single segment of color blue, then the value of "G on a stick" is equal to the first value from the series:

$$\frac{x+1}{1}, \frac{x+2}{2}, \frac{x+3}{4}, \frac{x+4}{8}, \frac{x+5}{16}, \frac{x+6}{32}, \cdots$$

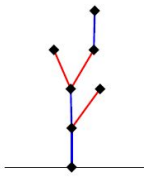
for which the numerator of the expression exceeds 1.

(If the "stick" under the game G with value x is instead red, then the value of that game on a stick is given by the first term in the series:

$$\frac{x-1}{1}, \frac{x-2}{2}, \frac{x-3}{4}, \frac{x-4}{8}, \frac{x-5}{16}, \frac{x-6}{32}, \cdots$$

for which the numerator of the expression is exceeded by -1.)

Let's now use this theorem to calculate the value of the position below:



The topmost single blue segment has value 1. The node below that has two red segments. One is a single segment with value -1, but the one on the right is blue on red (which should have value $-1/2$). To see that the $-1/2$ agrees with the

formulas, we need to look at the numerators: $x-1, x-2, \ldots$, where $x = 1$ (the value of the single blue strut) and find the first one exceeded by -1. We see that $1-1$ doesn't work, nor does $1-2$ (which is equal to -1, but is not exceeded it), but finally, $1-3 = -2$ is smaller than -1 therefore the stalk with blue on top of red is $\frac{1-3}{4} = -1/2$.

The next node down, which is one step above the ground, has a single red strut with value -1, plus a blue strut with a position equal to $-3/2 = -1 + (-1/2)$ on top of it. Now x will be $-3/2$ and we need to look at $x+1, x+2, \ldots$ until one is larger than 1. We see that $x+3$ works, so the value of the position is $(+3/2)/4 = +3/8$. That, plus the red segment with value -1 yields a game of value $-5/8$ on top of a blue stick. We have to add 2 to that to exceed 1, so the value of the full game will be: $((-5/8)+2)/2 = 11/16$.

Harmonic Sum:

```
/// Error is in the region of 2e-12 for values greater than 1000.
/// Run for loop for less than 1000.
const double Gamma = 0.5772156649;
double Hn(LL n)
{
```