Notes on how to solve Novice contest problems of 3-21-15

H. Building a Tree
Straightforward

I. Heads and Tails
The solution is straightforward. Use an array of 8 ints to collect the results. Develop a function `int convert(String s, int i)` that converts the three characters in s from position i to position i+2 to an int in [0,7]. Call this function in a for-loop 38 times, each time incrementing the result array in the position indicated by the function.

J. The Fly

(a) Ignore the fly
We just need to calculate the time at which the bicycles collide

(b) Convert the problem to one bicycle by adding the speeds of the two bicycles.

$$T = distance/speed$$

$$T = D/(A + B)$$

Then compute the distance of the fly:

$$fly\ distance = T * F$$

K. Sharing the Prize
This problem, and those like it, turn out to be the most frustrating for contestants. First, it is almost always best to convert monetary amounts to cents by multiplying by 100. Then, we add expenditures of Frankie and Johnny into two ints, call then Famt and Jamt. Then we subtract them, diff = Jamt - Famt and divide by 2 (integer division) giving halDiff = diff/2 and halfRem = diff % 2.

Consider the following examples:

Famt	Jamt	Outcome
25.01	25.01	Expenditures match
30.00	20.00	Johnny gives Frankie 5.00
20.00	30.00	Frankie gives Johnny 5.00
20.01	20.00	Johnny gives Frankie 0.01
20.00	20.01	Johnny gives Frankie 0.00 OR Frankie gives Johnny 0.00
30.01	20.00	Johnny gives Frankie 5.01
30.00	20.01	Johnny gives Frankie 5.00
20.01	30.00	Frankie gives Johnny 4.99
20.00	30.01	Frankie gives Johnny 5.00

The fifth case above does not occur in the judges' test data.
In the last 4 cases the total expenditure is the same, so the final amounts that each takes away must also be the same, with Frankie having 1c more to take home than Johnny.

L. Fibonacci Mod B

The first breakthrough is to realize that the sequence begins again with values 1 and 1. So we only have to look for the first occurrence of 1 after n=2.

Secondly, if you implement Fibonacci recursively, your program will run *for ever*. Fib(100) = Fib(99) + Fib(98) and the function repeatedly calculates values of Fib(n) as n gets smaller. The function has exponential runtime.

The iterative version is trivial to code. Start with cur = prv = 1, compute nxt = cur + prv and assign prv = cur, cur = nxt. This version has liner runtime. The rest is easy.

M. Hailstones

The coding is straightforward. Be careful to output the answer 1 if the input value is 1.

N. Square Spiral

This should be straightforward. Compute the x,y coordinates of the new endpoint (carefully) following the pattern,

```
int u = 0; int v = 0; int q = d;
for( int i=0; i<=1000; i++)  {
    u += q;
    q += d;
    v += q;   // now in 1st quadrant, not on axes
    if(x==u && y==v) break;
    q += d;
    u -= q;
    q += d;
    v -= q;
    q += d;
}
// etc.
```

Problems solved:

Advanced							Novice							
A	B	C	D	E	F	G	H	I	J	K	L	M	N	Total
15	6	4	18	6	0	17	27	29	18	3	7	24	18	192