**What are L-Systems?**

According to Wikipedia an L-system is an abbreviation for *Lindenmayer system* (named after the fancy biologist called Aristid Lindenmayer). Lindenmayer was a smart man with an unfortunate name. He came from Hungary. He was busy with all sorts of stuff. He studied at the University of Budapest. He also studied at the University of Michigan where he received his Ph.D. This was in plant physiology. After that he came to a fabulous country called the Netherlands. He went to Utrecht to become a professor. He taught Philosophy of Life Sciences and Biology and led the Theoretical Biology Group at the local university.

In Utrecht he was probably really depressed. He was walking a lot through the university's gardens. While doing this he noticed something weird. A really small plant that he had seen when he first came to Utrecht was now a very big plant. “How could this be,” he asked himself: “Every time I look at a plant it seems to stand still. But now this little plant – that I have know since my first arrival here – has become quit large.” This sudden epiphany made Lindenmayer do a lot of theoretical research. He studied in books, walked even more through the gardens, drew a few paintings and came up with a name for all his discoveries. He called it – very self-consciously – the Lindenmayer system.

But what was so special that a biologist had to name it after himself? Maybe he named it after the only person who could understand it. Probably not. So lets try to understand it ourselves. Maybe on the way, we find out if the name has anything to do with the explanation.

When – also looking at Wikipedia – I look up the L-system (this is how I will call the Lindenmayer system from now on) I see a lot of difficult mathematical code. *Hey, wasn’t this was supposed to be about flowers?* Evidently not. But when looking a little bit closer I see this:

*“Lindenmayer used L-systems to describe the behaviour of plant cells and to model the growth processes of plant development.”*

What a lot of fancy words. What is meant by this is that L-systems are a sort of tool that help you understand what happens when small plants become big plants. Wikipedia also says this:

*“L-systems have also been used to model the morphology of a variety of organisms.”*

Yeah… don’t try looking up the word morphology on Wikipedia. You immediately get sixteen different results. What it means is simply this: Morphology is the name people use when they study the form and structure of organisms (plants and such). So what this means is that they pay extra close attention when a plant grows. Sounds kind of lonely doesn’t it?

Believe it or not, many people find it very interesting. We are also going to try to find it interesting well. We start by looking at a simple example. There is a little plant called ‘algae’. Algae looks really cool under a microscope and feels very soft on your skin. Algae are also very horny, because it reproduces very fast. When an alga hits puberty and starts growing we can use the L-system to find out how it’s growing.

Here is a cool picture of seaweed (also an algae) that looks like the Kraken:



To determine how algae grows dr. Lindenmayer gave us four words:

* Variables
* Constants
* Axiom
* Rules

Sometimes you can add a few more, but let’s keep it simple. A *variable* is a set of symbols containing elements that can be replaced (we can use letters from the alphabet for example as these symbols). A *constant* is something that doesn’t change. An *axiom* is the starting point. And a *rule* is something that always does the same thing. A rule is somehow different from a constant. We’ll find out as we go.

Knowing this you can create the L-system code for algae. That looks like this:

**variables** : A B

**constants** : none

**axiom** : A

**rules** : (A → AB), (B → A)

So we have two variables, the ‘A’ and the ‘B’. We have no constants (wow, that’s easy). We start with an ‘A’. The rules are that ‘A’ becomes ‘AB’ and that ‘B’ becomes ‘A’. Now lets put this bitch to the test:

n=0: A

/ \

n=1: A B

/| \

n=2: A B A

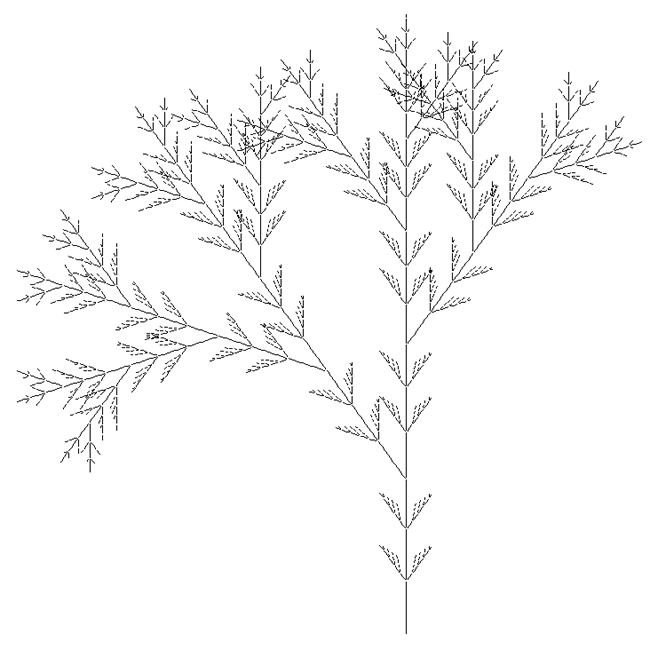
/| | |\

n=3: A B A A B

/| | |\ |\ \

n=4: A B A A B A B A

Isn’t that easy? Every time you go down one step on the ladder the rule does the same thing. ‘A’ becomes ‘AB’ and ‘B’ becomes ‘A’. You can do this as many times as you want. When you put this formula in a computer to make it into a picture it will generate something like this:



That looks already a lot more like a plant. And apparently this is also how an alga grows.

There are a lot of different L-systems. An L-system with a very cool name is the *Dragon Curve*. The code for this one looks like this:

**variables** : X Y

**constants** : F + –

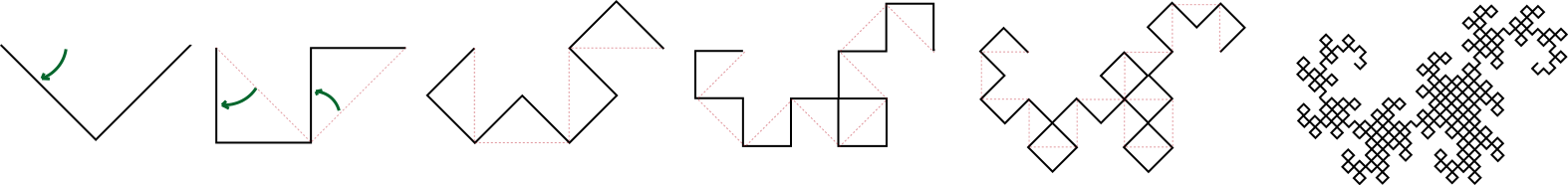
**axiom** : FX

**rules** : (X → X+YF+), (Y → -FX−Y)

**angle** : 90°

In this code we add *angle*. We also see that *rules* becomes a little more different. Don’t worry, I’ll explain. Here, ‘F’ means “draw forward”, ‘−’ means “turn left 90°”, and ‘+’ means “turn right 90°”. ‘X’ and ‘Y’ cannot be seen in the end result. We use them to control the change of the curve.

We see an example of the result here:



Did you know that when you fold a piece of paper, and fold it again, and again the creases that arise will have the same pattern as the Dragon Curve. Although that knowledge is very nice, it’s still quit difficult to understand.

n=0: FX

/ \

n=1: FX XF

/ | | \

n=2: FX XF XF FX

Or maybe we can make it easier by calling ‘F’ ‘ZIG’ and ‘X’ ‘ZAG” (like the folding of a paper).

n=0: ZIG ZAG

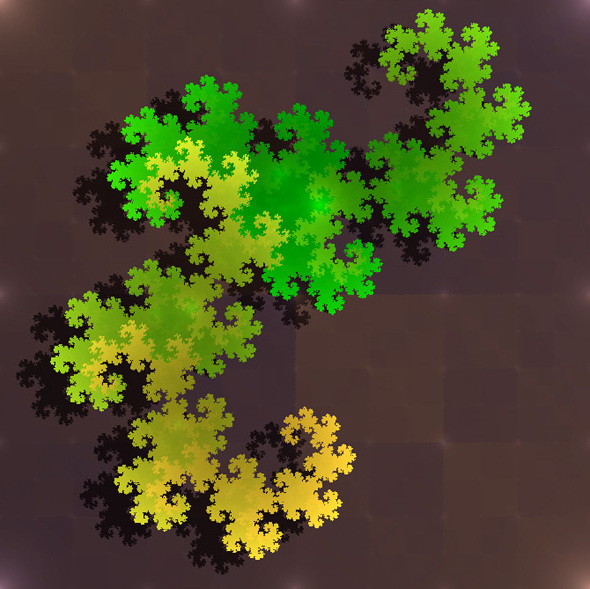
| |

n=1: ZIG ZAG ZAG ZIG

/ \ / \

n=2: ZIGZAG ZAGZIG ZAGZIG ZIGZAG

If you go on forever it will look something like this:



So what we’ve learned so far is that L-systems are mathematical codes that can go on forever. In math-language this is also called and *Algorithm*. An L-system is an algorithm. You can create fancy patterns with them. Or trees.

**Understanding L-systems**

Do you think we can try to make it even more difficult by using mathematical formulas?

I found this formula on Wikipedia:

**G** = (*V*, ω, *P*)

What do you think it means? It is actually a simplified explanation of the big formula’s we learned about before. Supposedly the ‘*V*’ is the *variable*. The ‘ω’ (the Greek letter Omega) is the *axiom*. We know that the axiom is a string (ooh la la) of symbols from ‘*V*’. The axiom defines the initial state of the system. Just like the letter ‘A’ in our algae example. The ‘*P*’ is a set of *production rules*. The ‘*P*’ defines the way we can replace *variables* with combinations of *constants* and other *variables*.

So:

**G** = Grammar

*V* = Variables (the alphabet)

ω = Axiom (start/initiator)

*P* = Production rules

The little formula above is also called a *Tuple*. A tuple is a finite ordered list of elements. We won’t go into this concept, because that would make it too difficult.

Now we start to grasp what L-system are and understand stem a little bit. But you know what would be really fun? Try to make something with it using the computer.

Let take a look on the Internet to see what we need. I came across this little program called *Processing*. Processing is a computer program that looks a lot like the inside of your math book. You can fill in some formulas, click on play and suddenly you have a really fancy image. Sometimes it can move. Sometimes it responds to your voice. Sometimes it does nothing at all (this means you’ve made a mistake). It can do all sorts of stuff. And also convert L-systems to images.