

# Procedural Content Generation

## Introduction

In-class assignments will be the homework assignments.

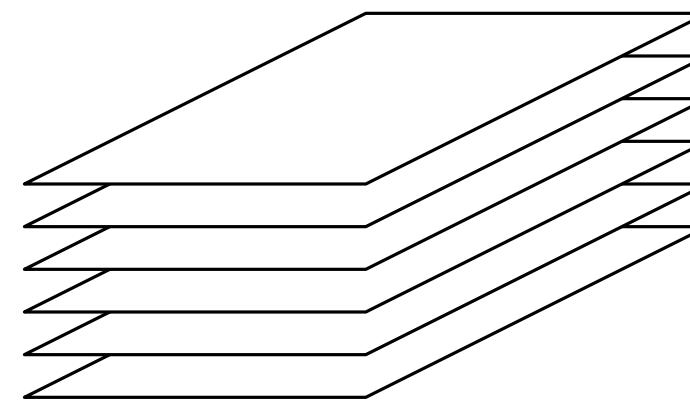
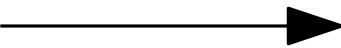
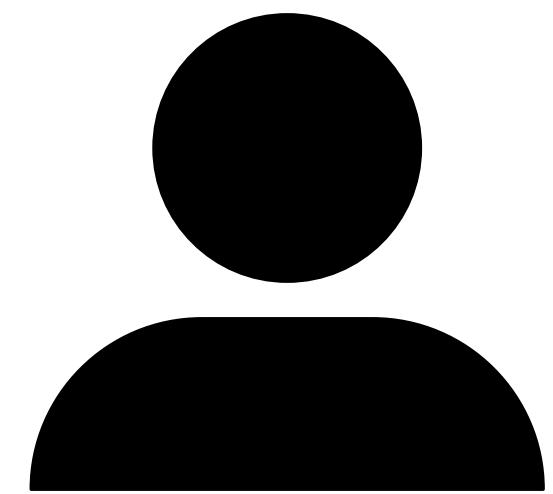
In-class assignments will be the homework assignments.

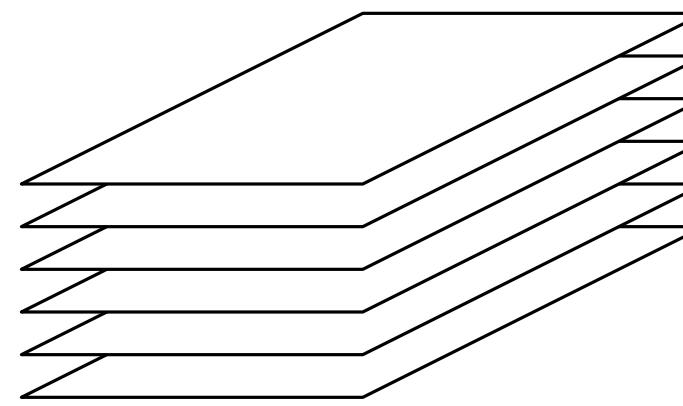
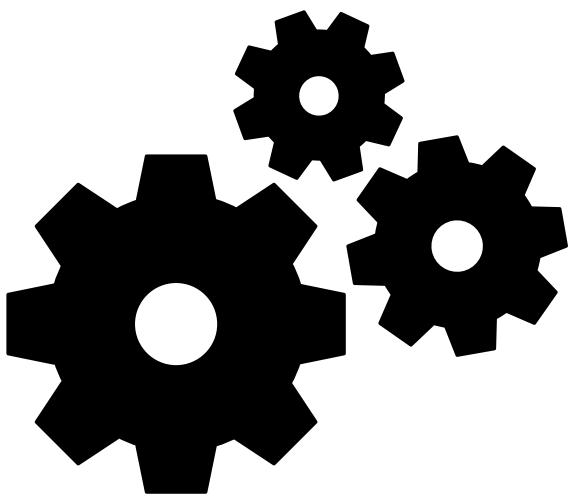
Implementation projects will be done in pairs.

In-class assignments will be the homework assignments.

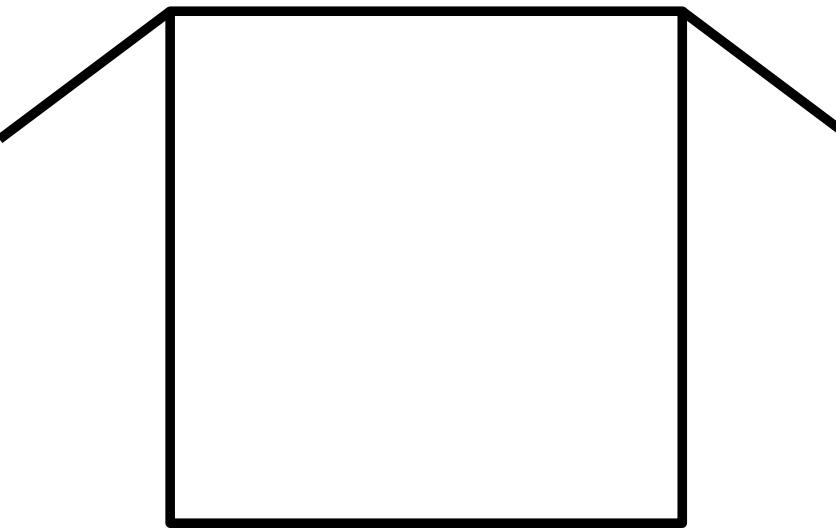
Implementation projects will be done in pairs.

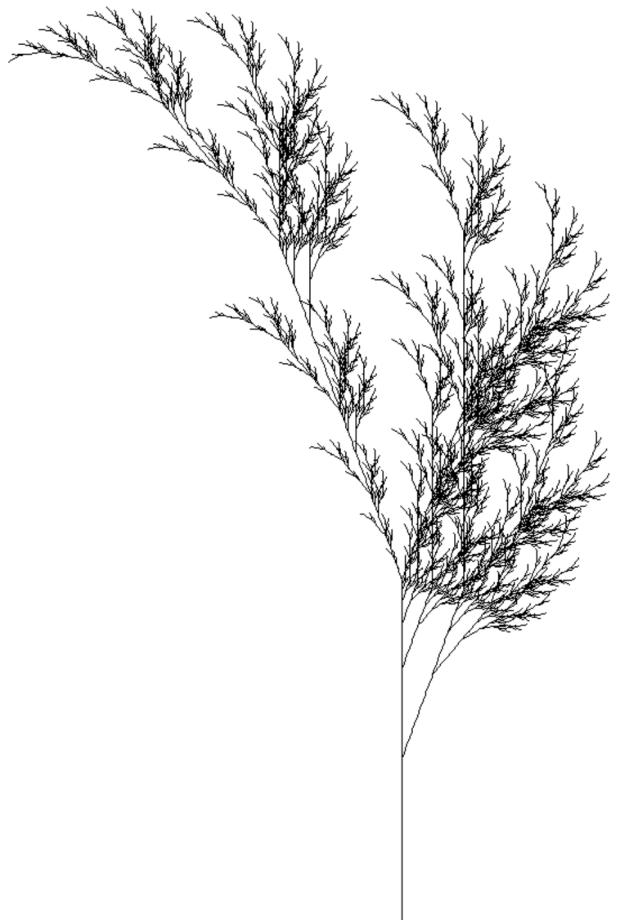
You can find your own partner until Friday, after that I will make random pairings.



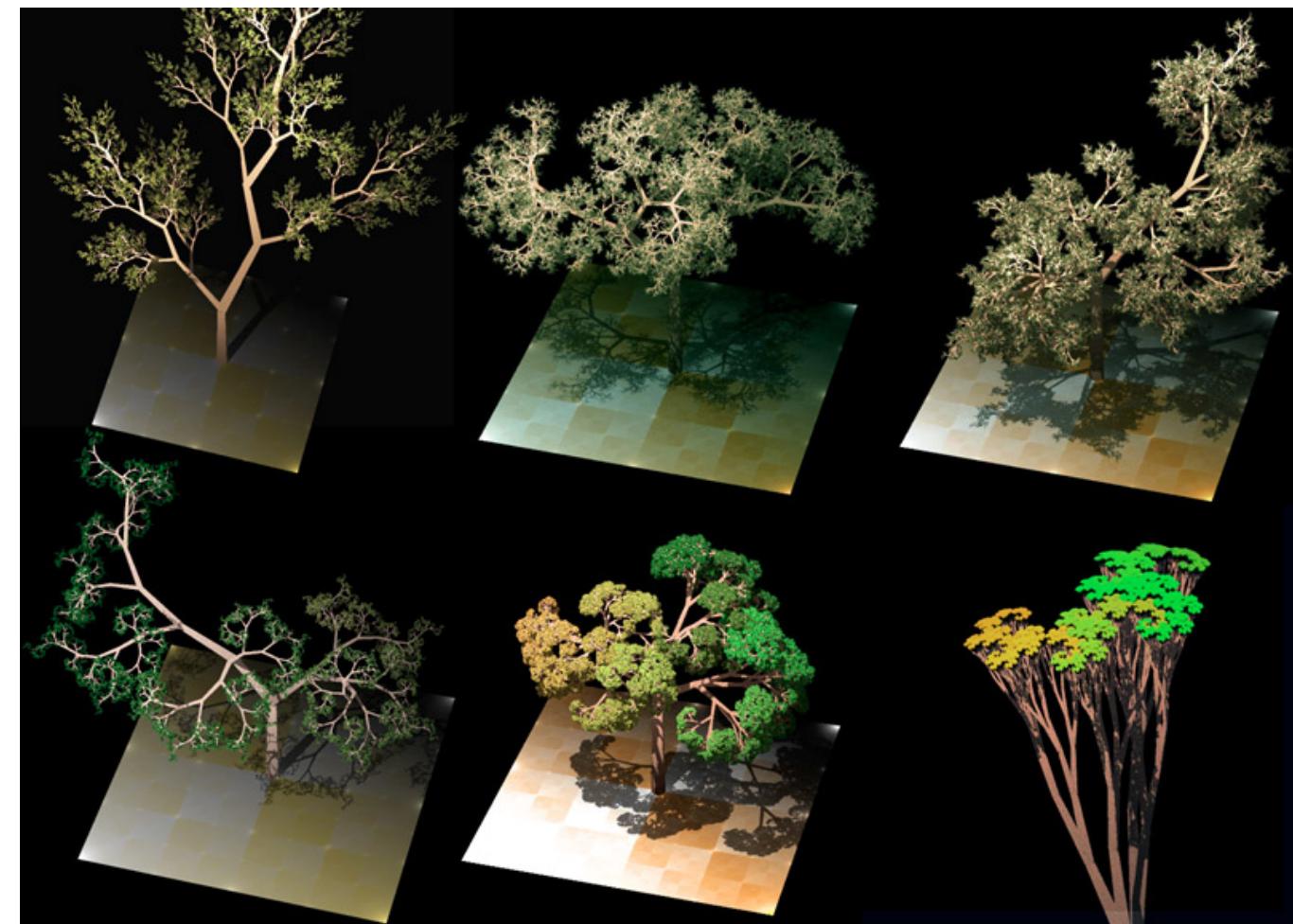


?

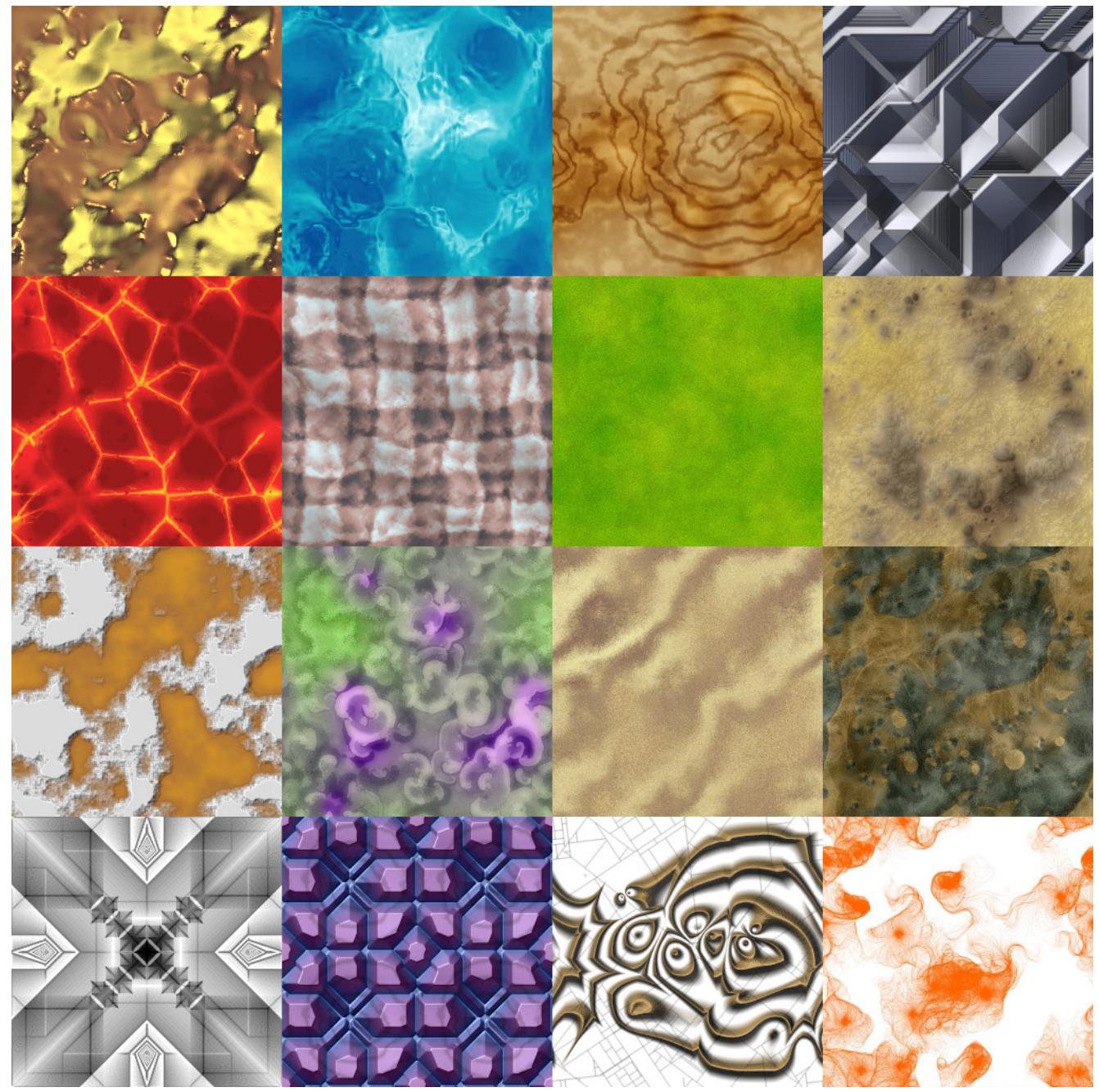




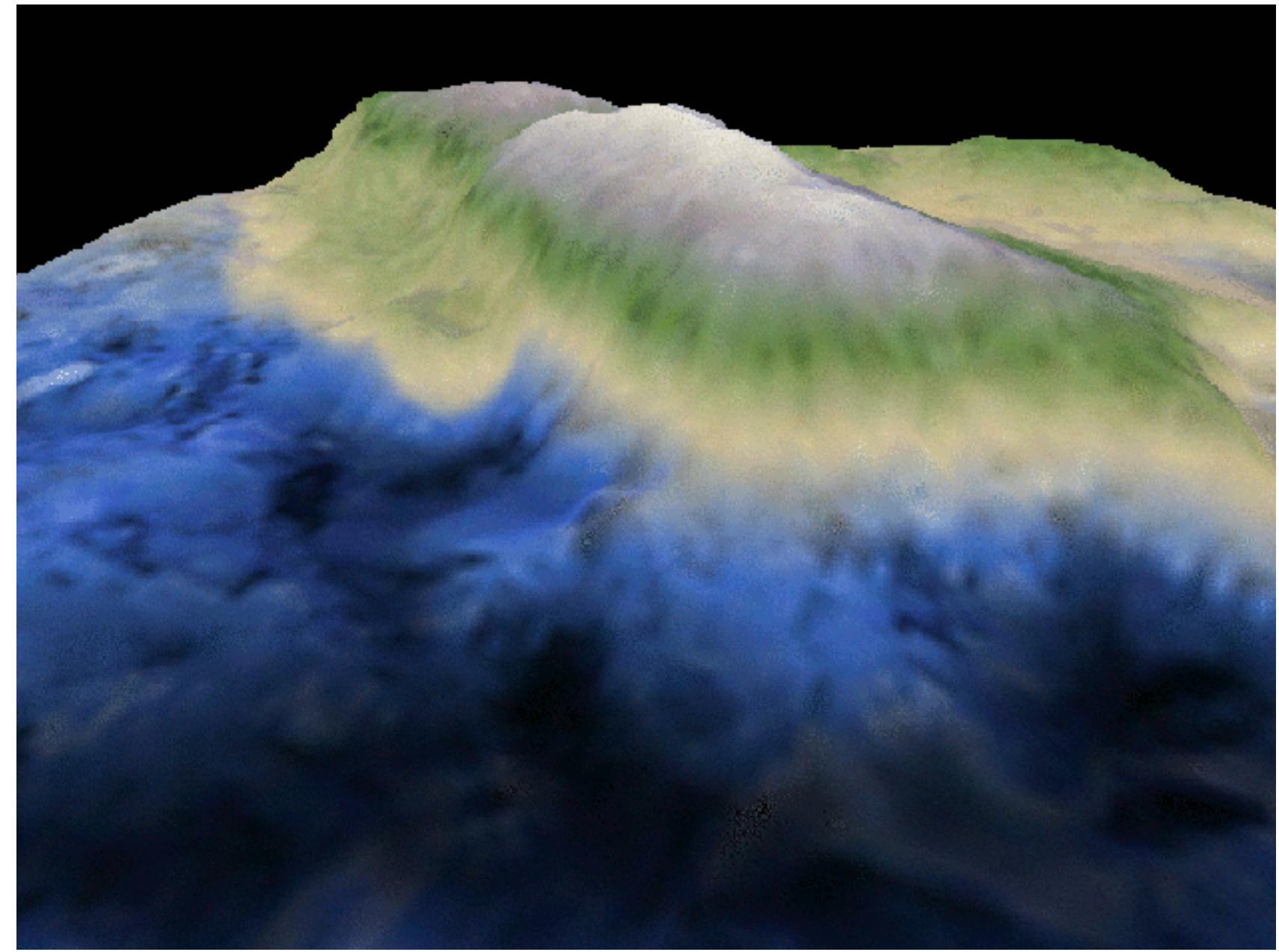
PCG book chapter 5



[en.wikipedia.org/wiki/Procedural\\_generation](https://en.wikipedia.org/wiki/Procedural_generation)



[gitlab.com/drummyfish/ptdesigner](https://gitlab.com/drummyfish/ptdesigner)



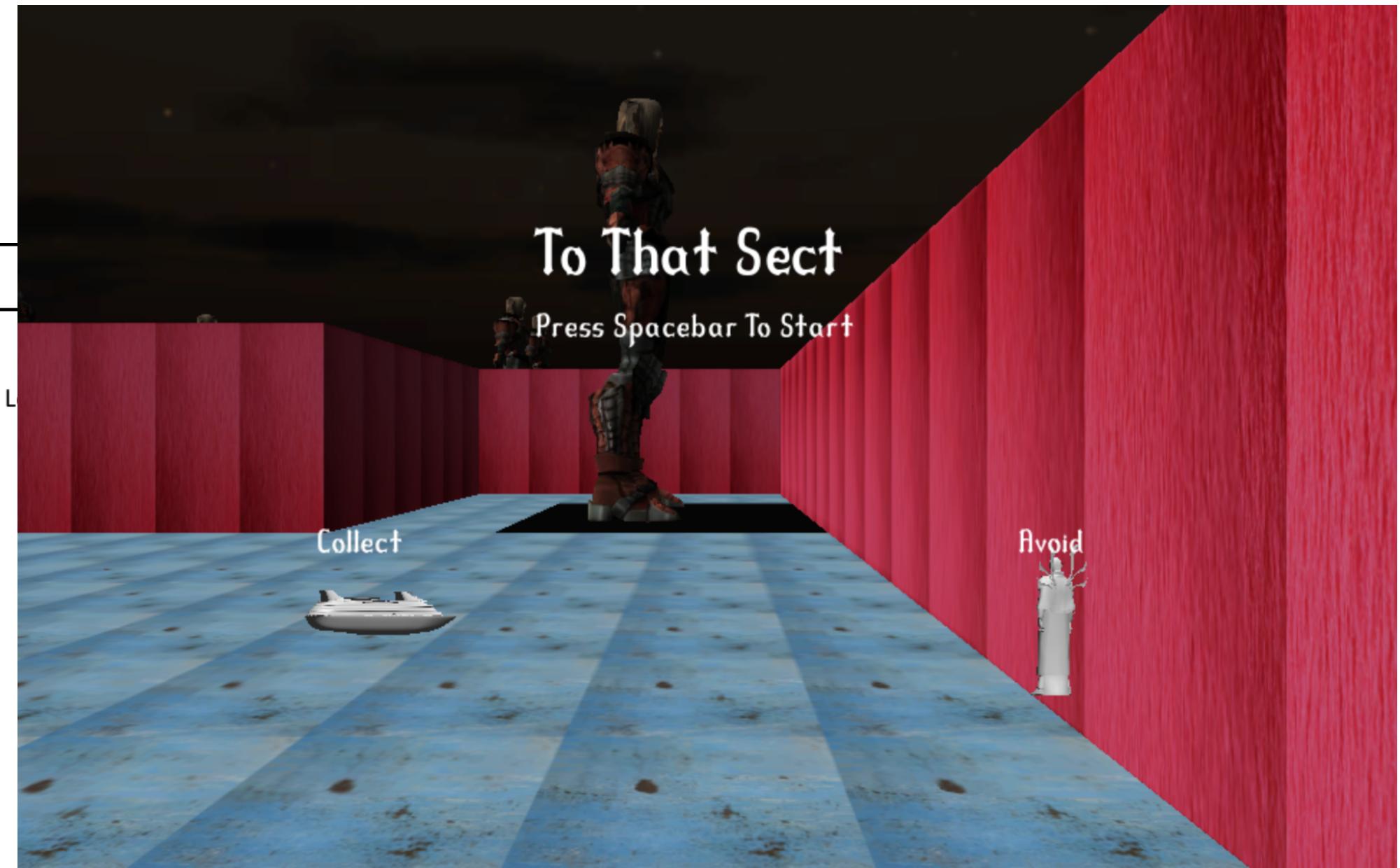
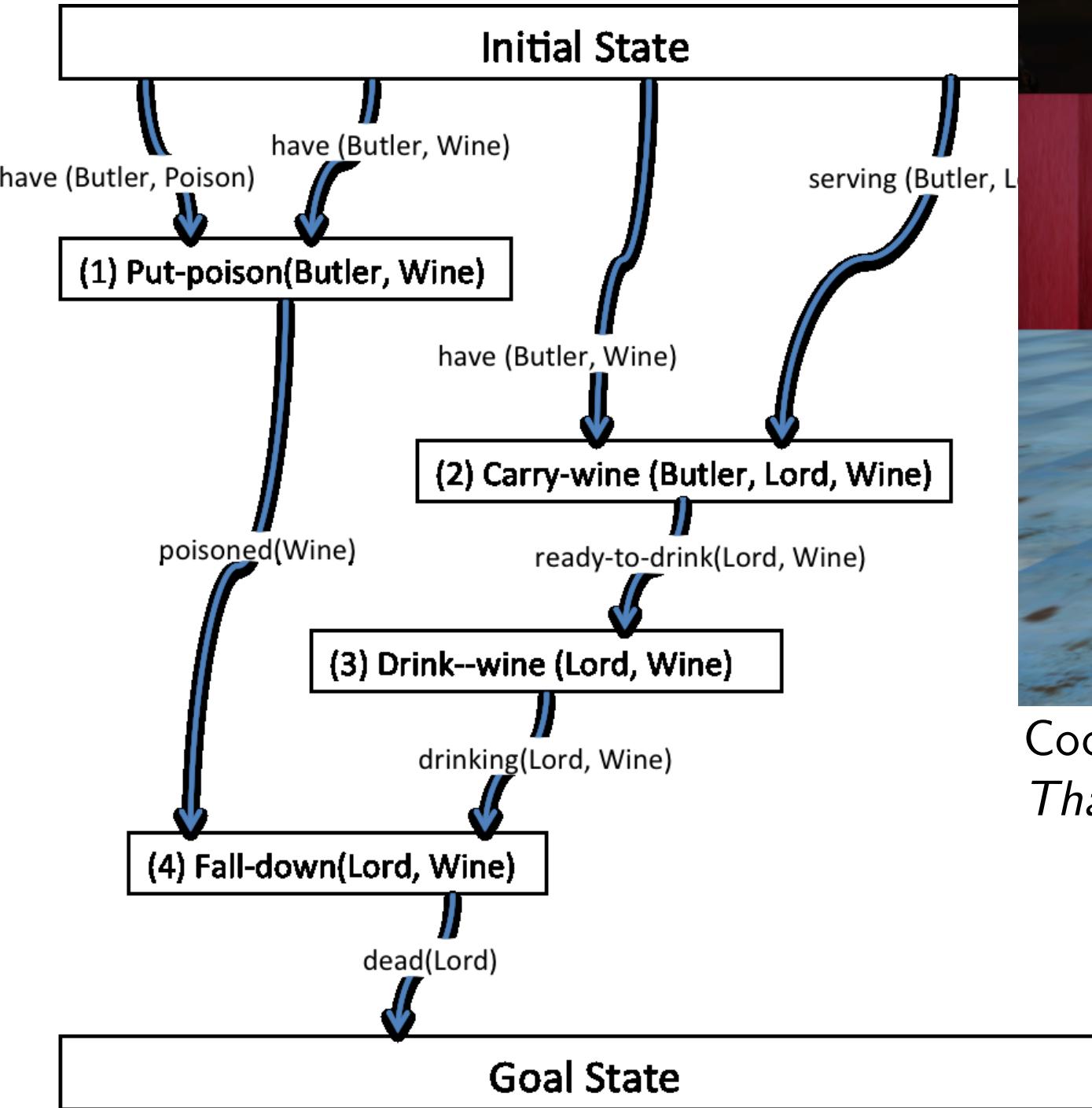
[cprogramming.com/discussionarticles/texture\\_generation.html](https://cprogramming.com/discussionarticles/texture_generation.html)



spelunkypedia.com



Age of Empires II: Definitive Edition



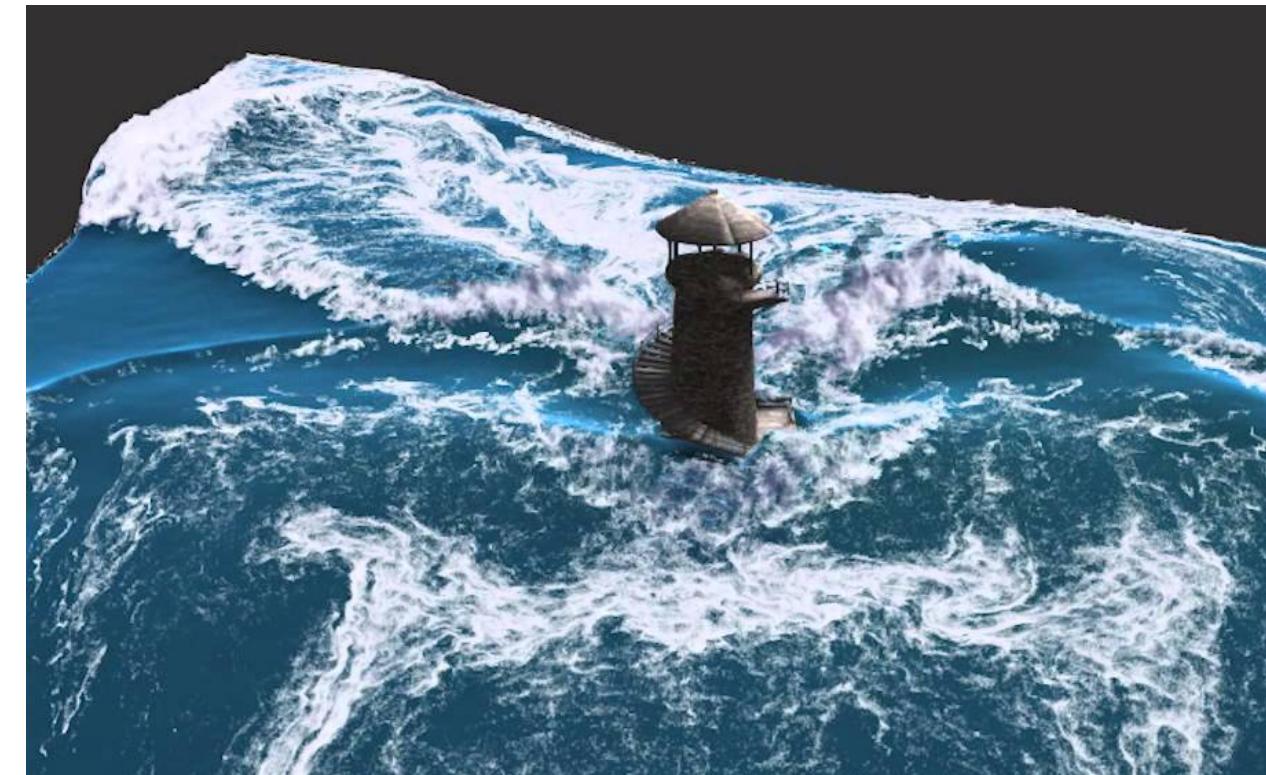
Cook and Colton, *Ludus Ex Machina: Building A 3D Game Designer That Competes Alongside Humans* (2014)



[staff.science.uu.nl/~gerae101/](http://staff.science.uu.nl/~gerae101/)



[github.com/keijiro/PuppetTest](https://github.com/keijiro/PuppetTest)

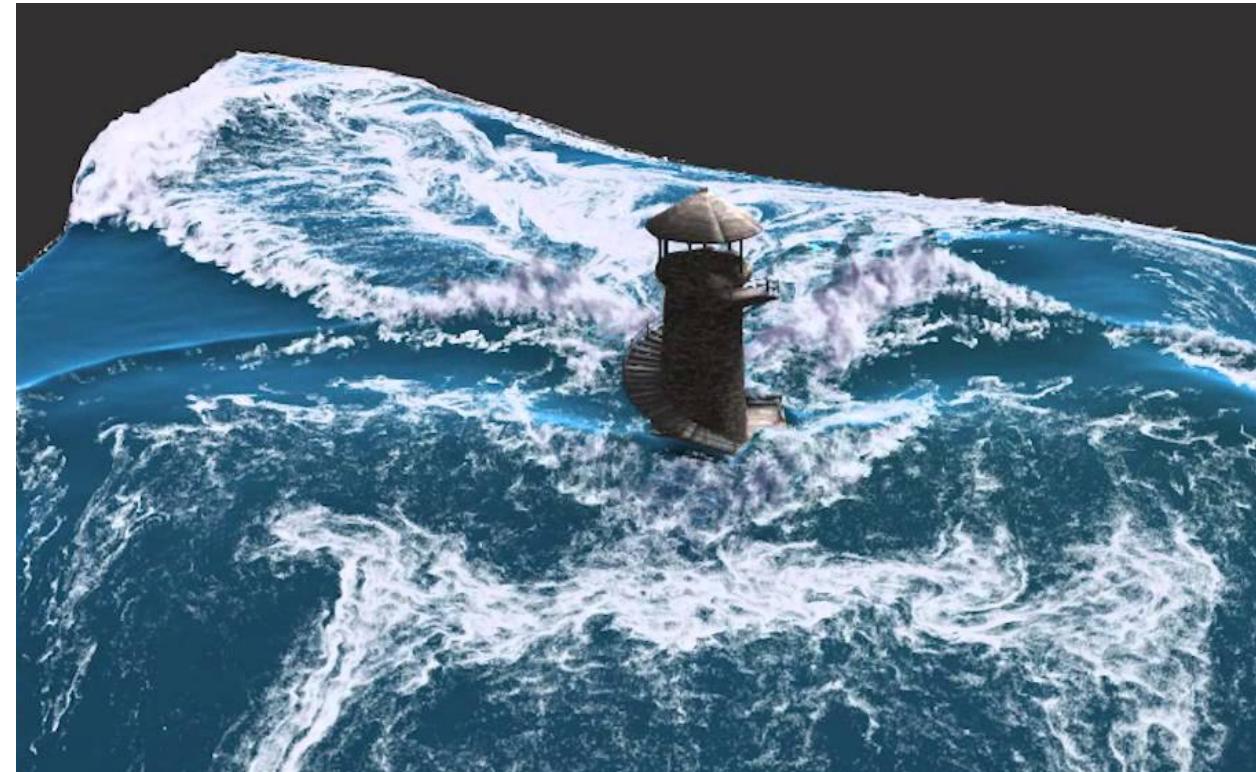


Ihmsen et al., *Unified spray, foam and air bubbles for particle-based fluids* (2012)



[staff.science.uu.nl/~gerae101/](http://staff.science.uu.nl/~gerae101/)

Computer animation  
Motion and manipulation  
Game physics  
Crowd simulation



Ihmsen et al., *Unified spray, foam and air bubbles for particle-based fluids* (2012)



[github.com/keijiro/PuppetTest](https://github.com/keijiro/PuppetTest)

# Assignment 1

Think of three applications for which PCG would be useful.  
What kind of content could be generated in each application?

# Grammars

# Grammars

ABC...

# Grammars

ABC...

Hello, world!

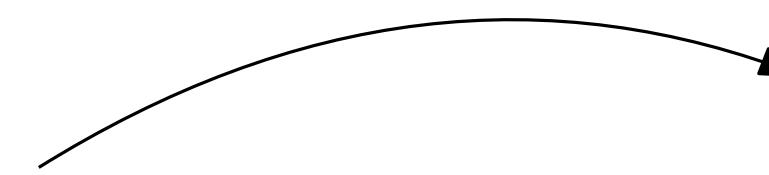
Call me Ishmael.

Boom shaka-laka-laka.

...

# Grammars

ABC...



Hello, world!

Call me Ishmael.

Boom shaka-laka-laka.

...

# Grammars

$S$  (start symbol)

$\mathcal{A}$  (terminal symbols)

$\mathcal{B}$  (non-terminal symbols)

$\alpha_1 \rightarrow \beta_1$  (production rules)

$\alpha_2 \rightarrow \beta_2$

# Grammars

$S$  (start symbol)

$\{a, b, c\}$  (terminal symbols)

$\{S, B\}$  (non-terminal symbols)

$S \rightarrow aBSc$  (production rules)

$S \rightarrow abc$

$Ba \rightarrow aB$

$Bb \rightarrow bb$

# Grammars

$S$	(start symbol)
$\{a, b, c\}$	(terminal symbols)
$\{S, B\}$	(non-terminal symbols)
$S \rightarrow aBSc$	(production rules)
$S \rightarrow abc$	
$Ba \rightarrow aB$	
$Bb \rightarrow bb$	
$S \rightarrow aBSc \rightarrow aBabcc \rightarrow aaBbcc \rightarrow aabbcc$	

# Chomsky hierarchy

# Chomsky hierarchy

Type-0: unrestricted

$$\gamma \rightarrow \beta$$

$\gamma$  non-empty

# Chomsky hierarchy

Type-0: unrestricted

$$\gamma \rightarrow \beta$$

$\gamma$  non-empty

Type-1: context-sensitive

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

# Chomsky hierarchy

Type-0: unrestricted

$$\gamma \rightarrow \beta$$

$\gamma$  non-empty

Type-1: context-sensitive

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

Type-2: context-free

$$A \rightarrow \alpha$$

# Chomsky hierarchy

Type-0: unrestricted

$$\gamma \rightarrow \beta$$

$\gamma$  non-empty

Type-1: context-sensitive

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

Type-2: context-free

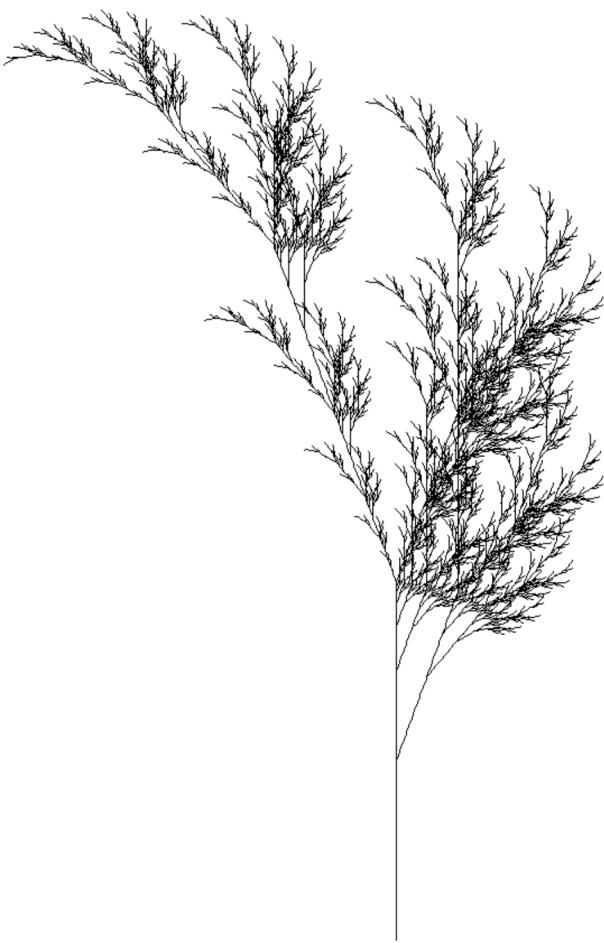
$$A \rightarrow \alpha$$

Type-3: regular

$$A \rightarrow a$$

$$A \rightarrow aB$$

# L-systems





Aristid Lindenmayer

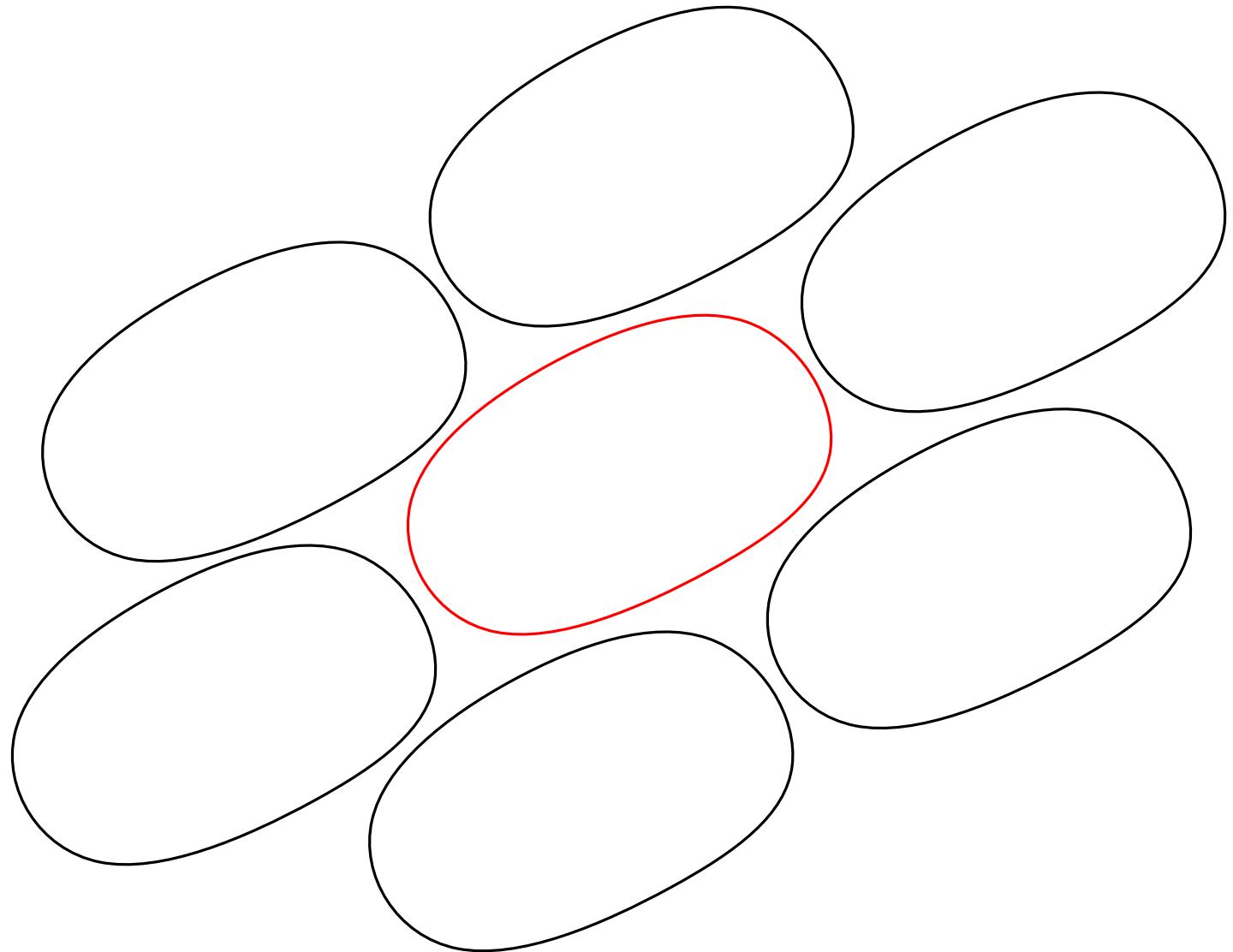
THE VIRTUAL LABORATORY

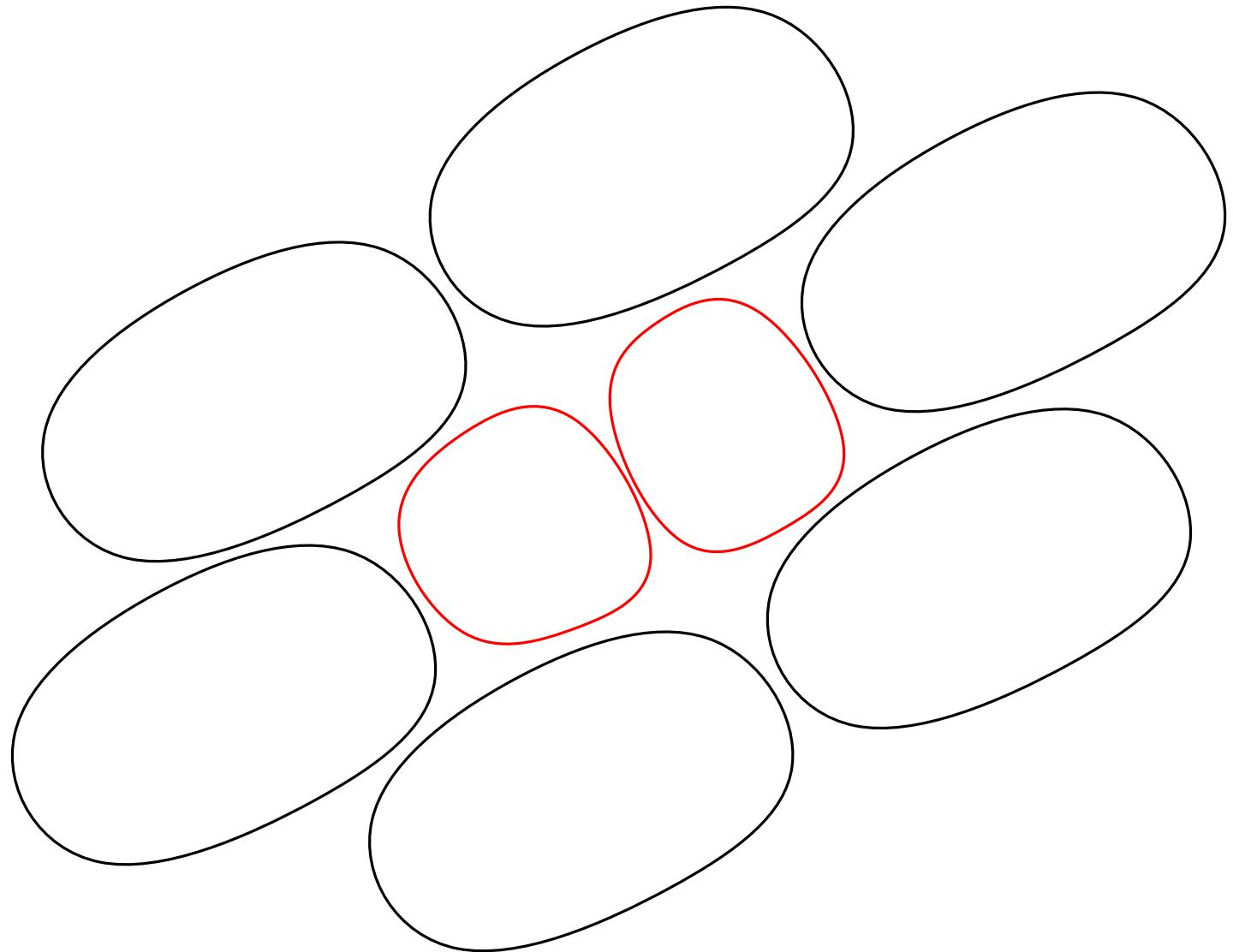
THE ALGORITHMIC BEAUTY OF PLANTS

PRZEMYSŁAW PRUSINKIEWICZ • ARISTID LINDENMAYER

The cover of the book 'The Algorithmic Beauty of Plants' by Przemysław Prusinkiewicz and Aristid Lindenmayer. The cover features a close-up photograph of three roses (red, orange, and dark red) against a white background.

SPRINGER - VERLAG





string.replace(...)

$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rules)

 $B \rightarrow A$

$A$

(axiom)

$A \rightarrow AB$

(production rules)

$B \rightarrow A$

$A$

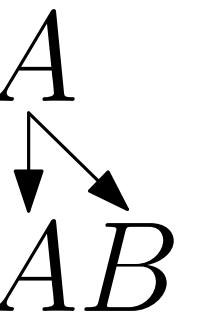
$A$

(axiom)

$A \rightarrow AB$

(production rules)

$B \rightarrow A$

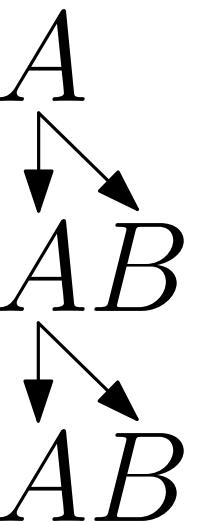


$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rules)

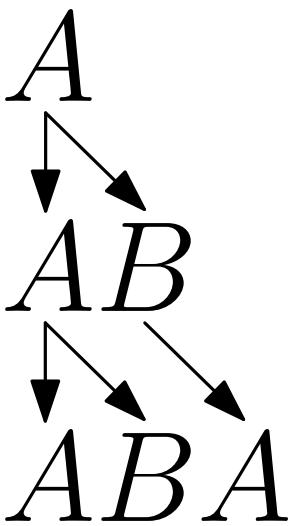
 $B \rightarrow A$ 

$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rules)

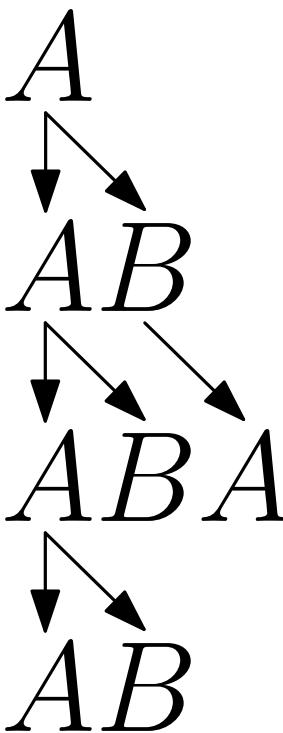
 $B \rightarrow A$ 

$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rules)

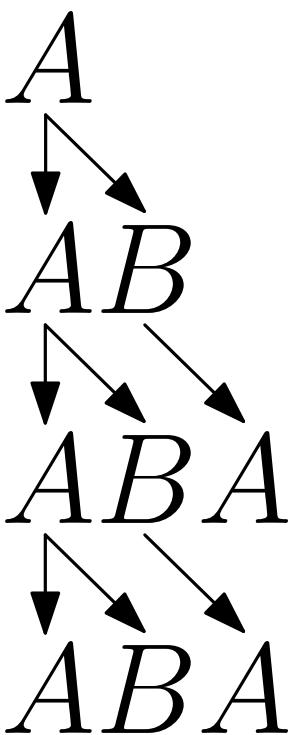
 $B \rightarrow A$ 

$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rules)

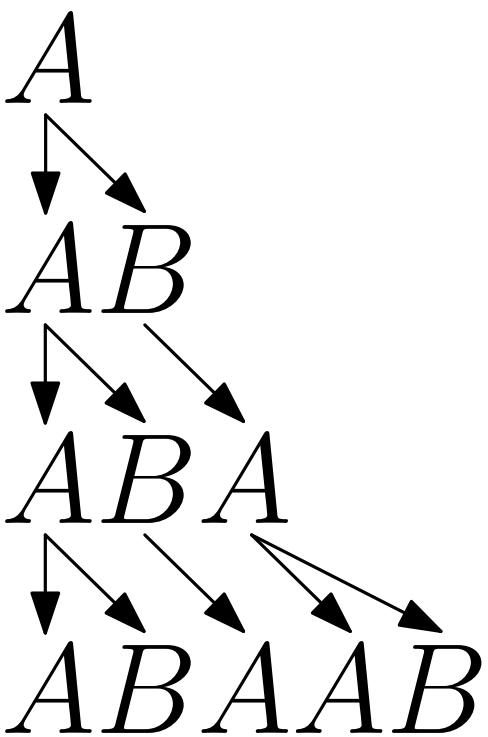
 $B \rightarrow A$ 

$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rules)

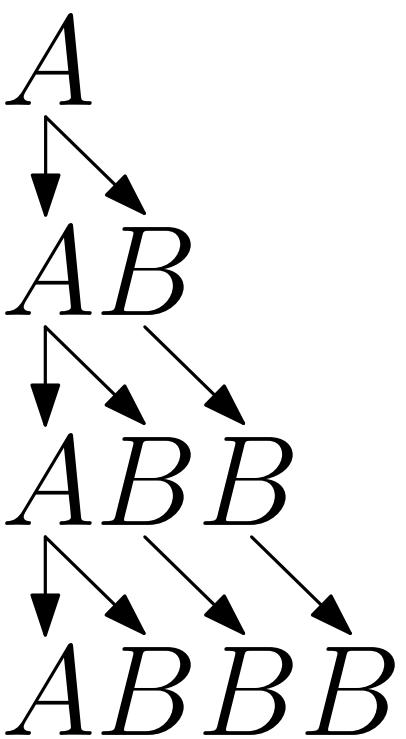
 $B \rightarrow A$ 

$A$ 

(axiom)

 $A \rightarrow AB$ 

(production rule)



# Assignment 2

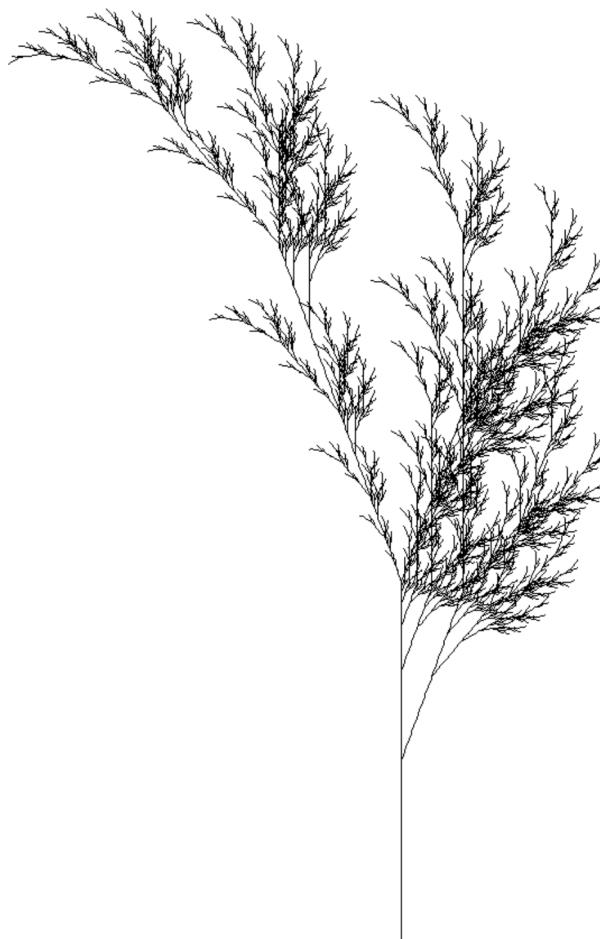
Evaluate the first level of the following L-system:

$F + +F + +F$  (axiom)

$F \rightarrow F - F + +F - F$  (production rule)

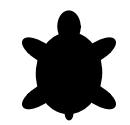
$ABAABABA \rightarrow$

?

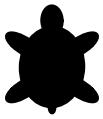




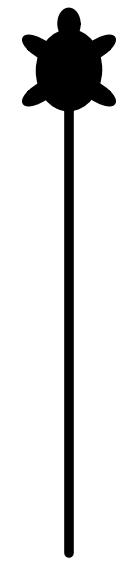
[en.wikipedia.org/wiki/Turtle](https://en.wikipedia.org/wiki/Turtle)



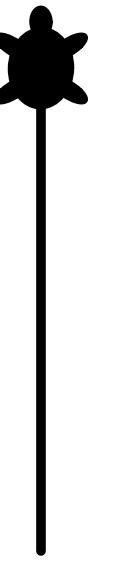
forward(10)



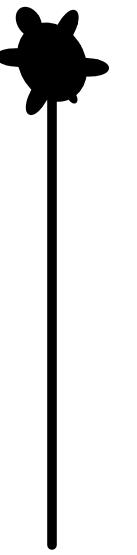
`forward(10)`



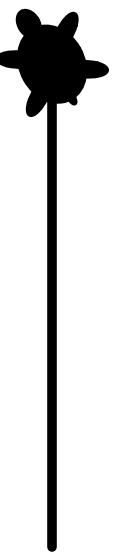
forward(10)  
left(30)



forward(10)  
left(30)

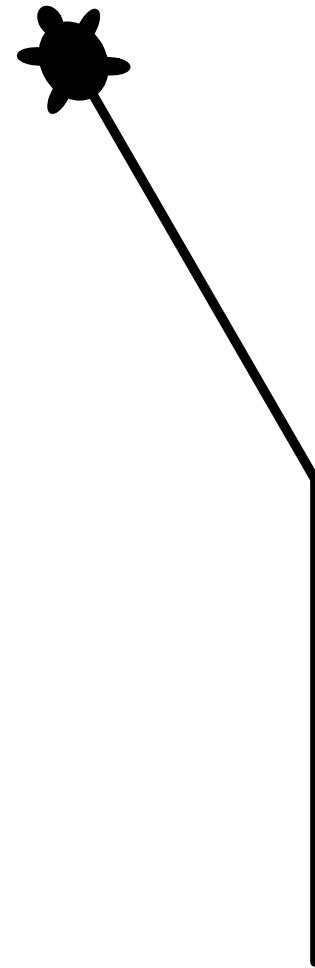


```
forward(10)  
left(30)  
forward(10)
```

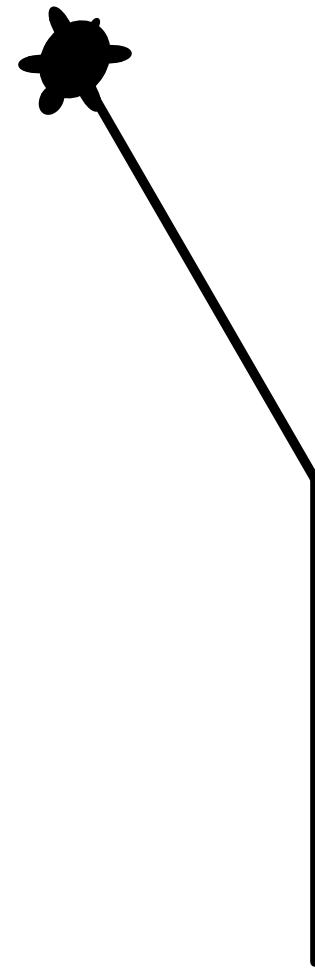




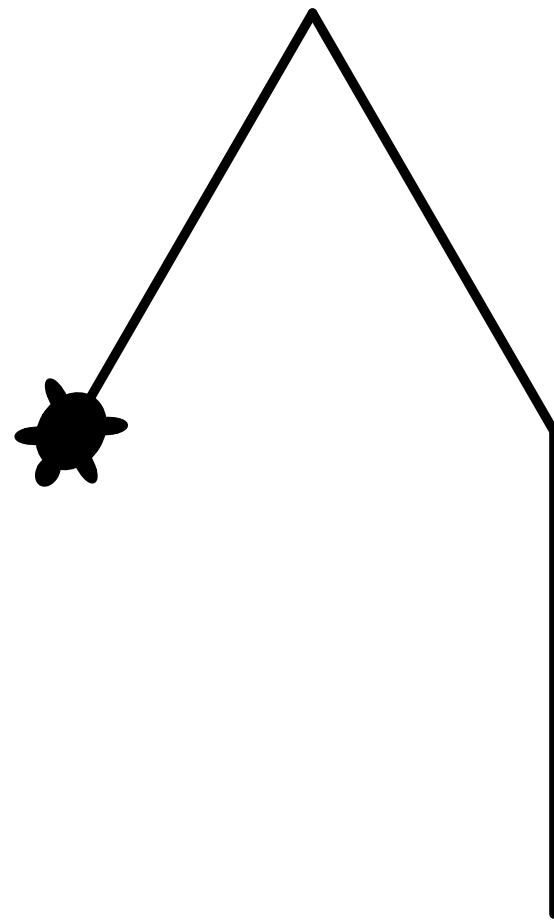
```
forward(10)  
left(30)  
forward(10)
```



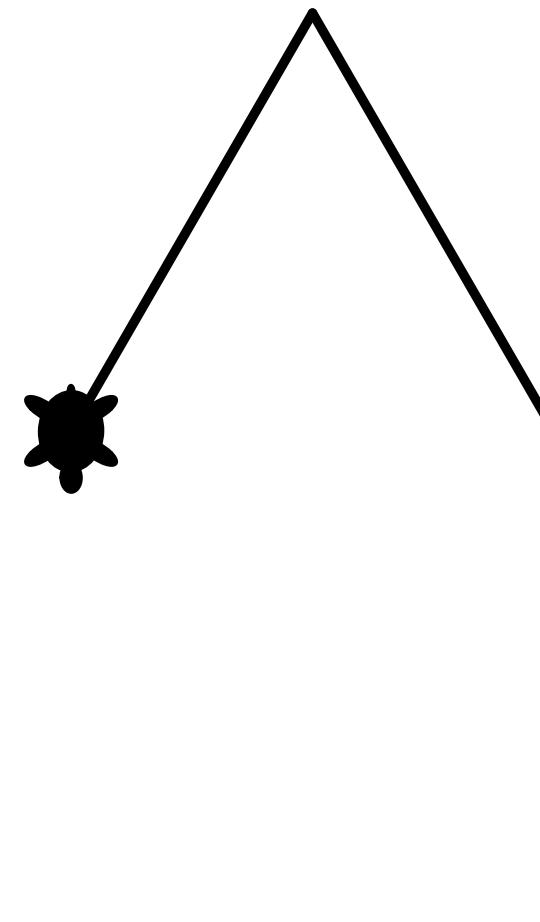
```
forward(10)
left(30)
forward(10)
left(120)
forward(10)
left(30)
forward(10)
left(90)
forward(10)
```



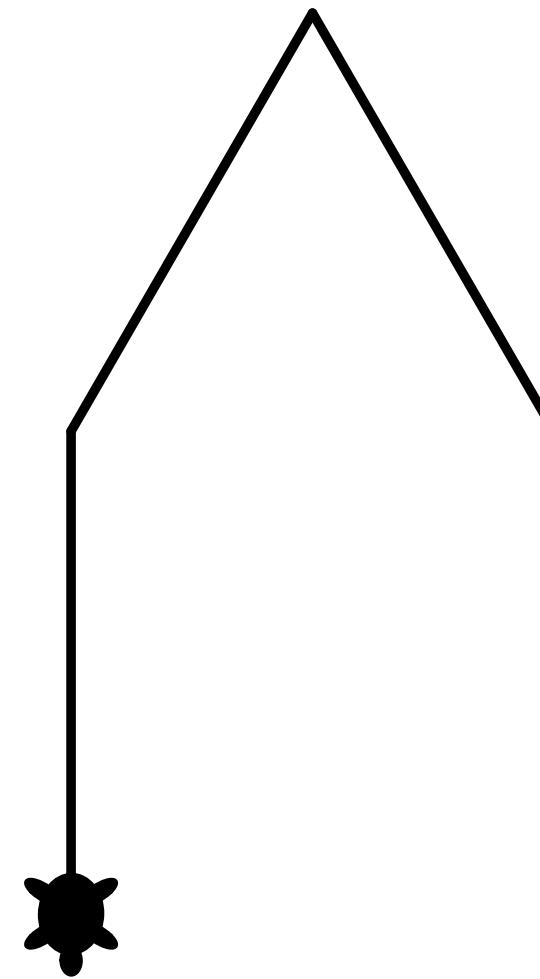
```
forward(10)
left(30)
forward(10)
left(120)
forward(10)
left(30)
forward(10)
left(90)
forward(10)
```



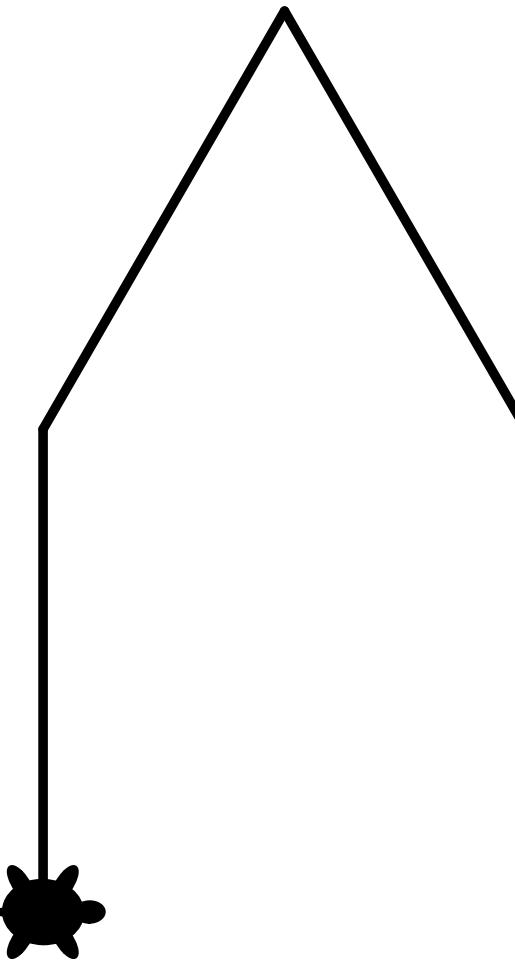
```
forward(10)
left(30)
forward(10)
left(120)
forward(10)
left(30)
forward(10)
left(90)
forward(10)
```



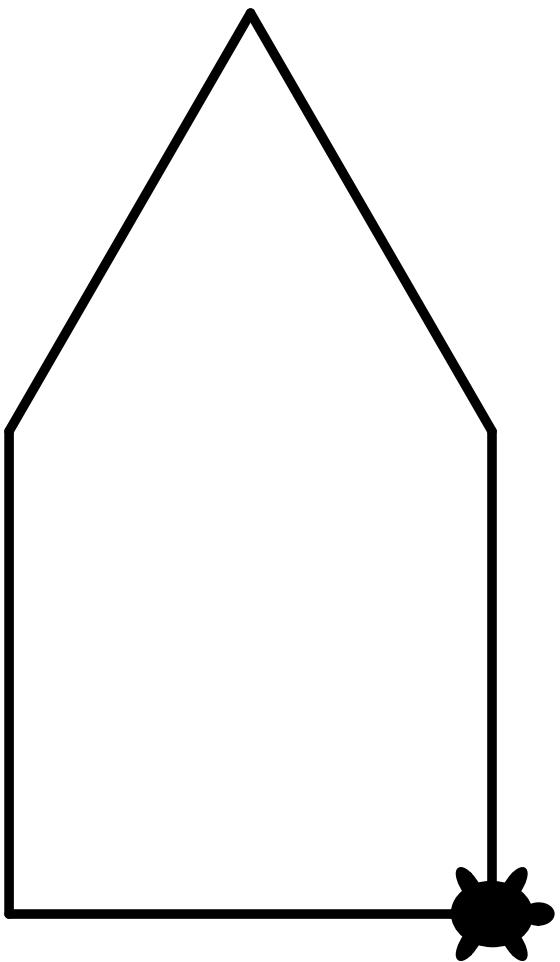
forward(10)  
left(30)  
forward(10)  
left(120)  
forward(10)  
left(30)  
forward(10)  
left(90)  
forward(10)



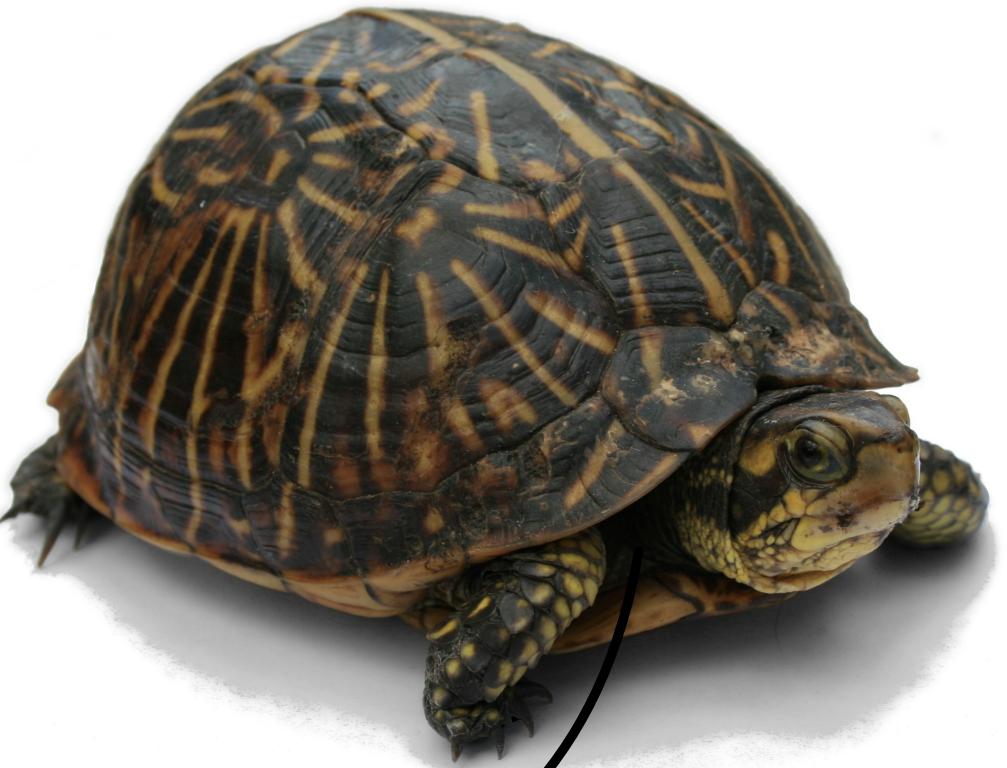
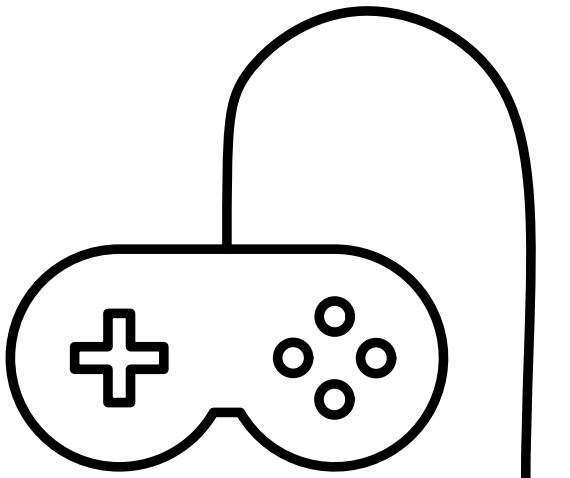
```
forward(10)
left(30)
forward(10)
left(120)
forward(10)
left(30)
forward(10)
left(90)
forward(10)
```



forward(10)  
left(30)  
forward(10)  
left(120)  
forward(10)  
left(30)  
forward(10)  
left(90)  
forward(10)



forward(10)  
left(30)  
forward(10)  
left(120)  
forward(10)  
left(30)  
forward(10)  
left(90)  
forward(10)

$ABAABABA =$ 

parameters: length  $d$  and angle  $\delta$

parameters: length  $d$  and angle  $\delta$

$F$  : forward( $d$ )

parameters: length  $d$  and angle  $\delta$

$F$  : forward( $d$ )

$f$  : forward( $d$ ) (don't draw)

parameters: length  $d$  and angle  $\delta$

$F$  : forward( $d$ )

$f$  : forward( $d$ ) (don't draw)

$+$  : left( $\delta$ )

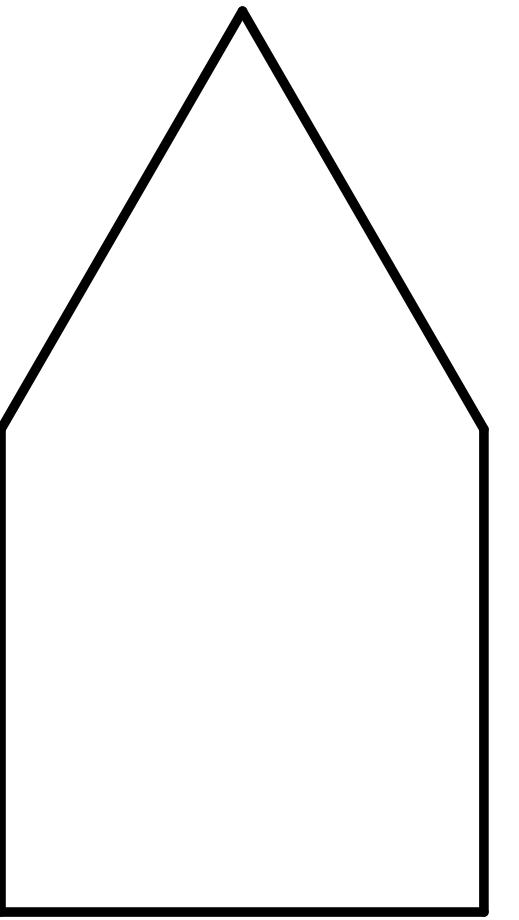
parameters: length  $d$  and angle  $\delta$

$F$  : forward( $d$ )

$f$  : forward( $d$ )    (don't draw)

$+$  : left( $\delta$ )

$-$  : right( $\delta$ )



$$\delta = 30^\circ$$

$$F + F + + + + F + F + + + F$$

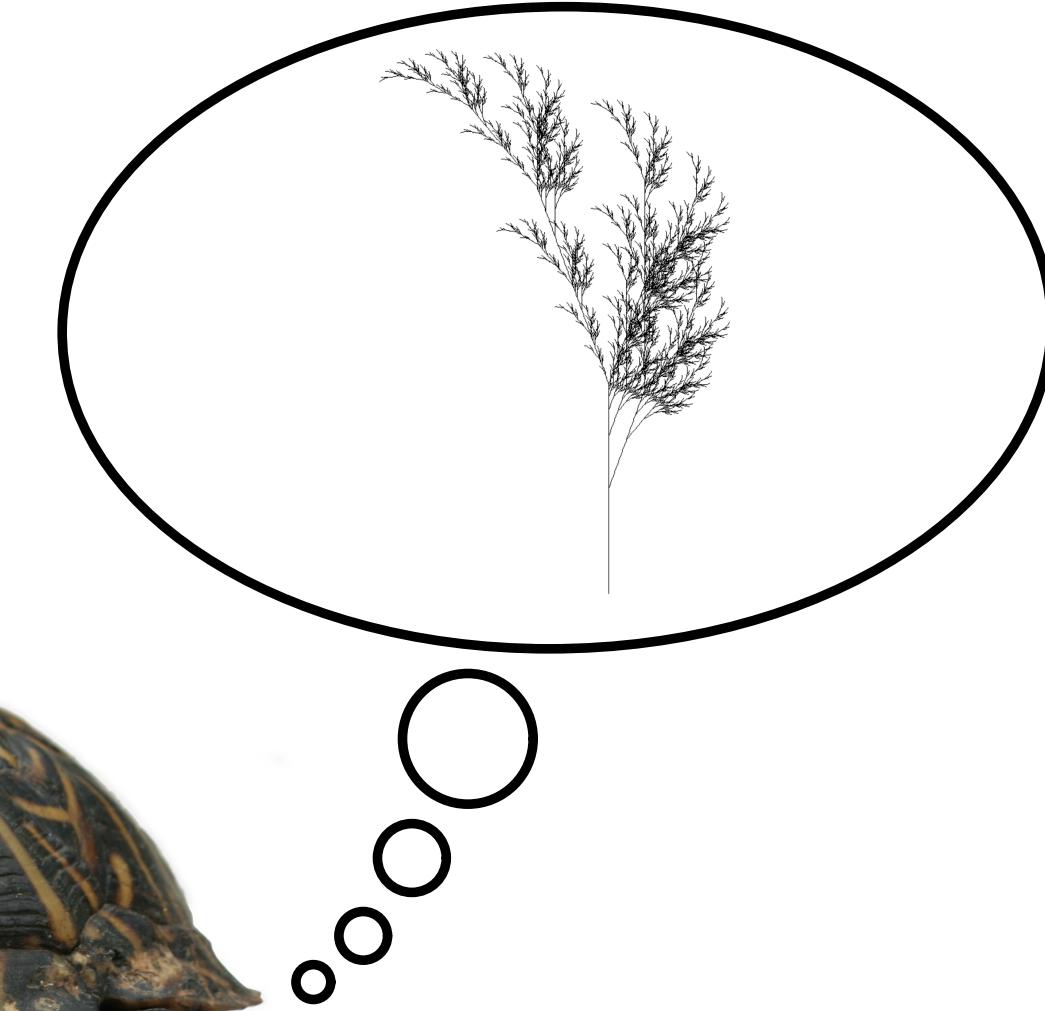
# Assignment 3

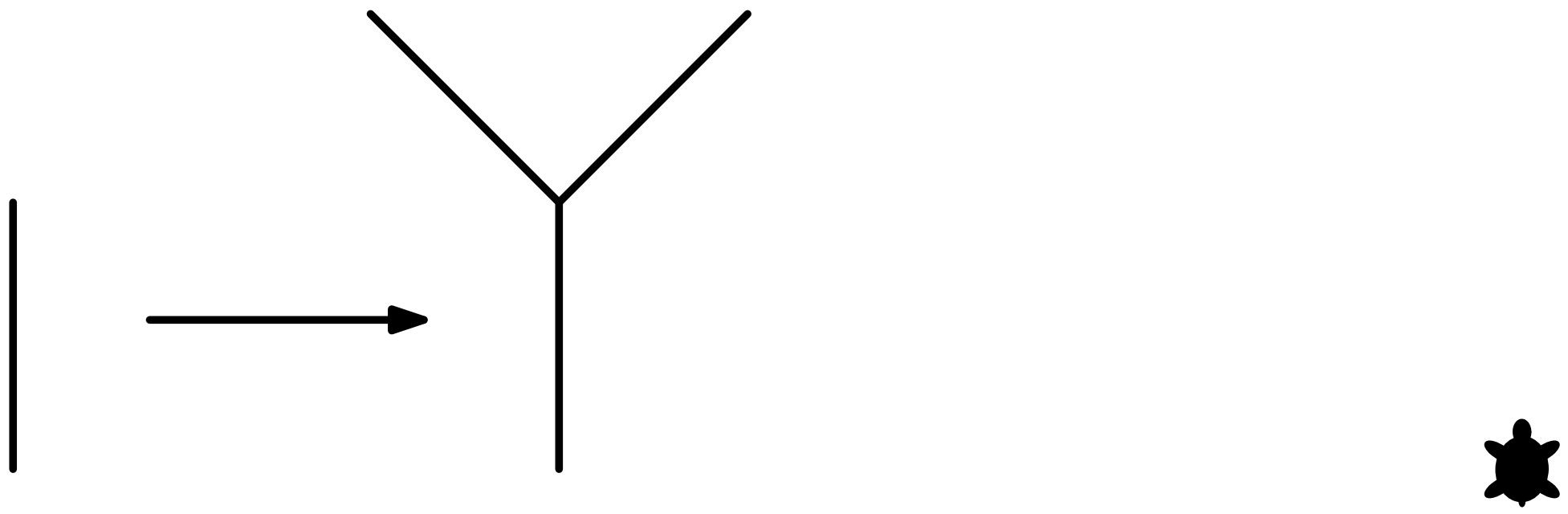
Consider the L-system from Assignment 2:

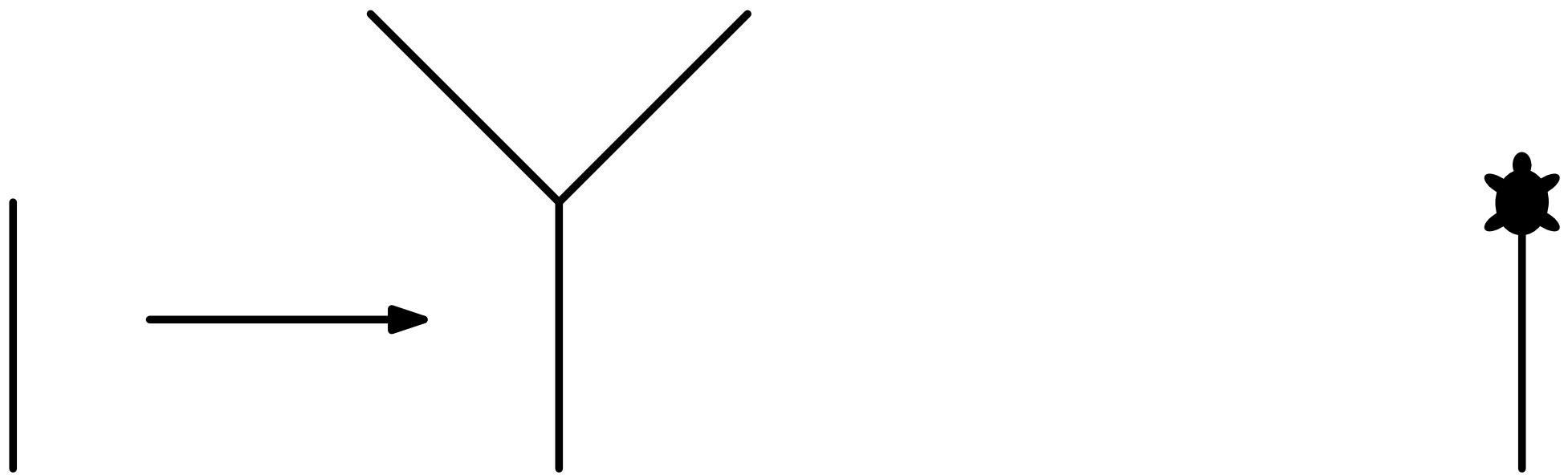
$F + +F + +F$  (axiom)

$F \rightarrow F - F + +F - F$  (production rule)

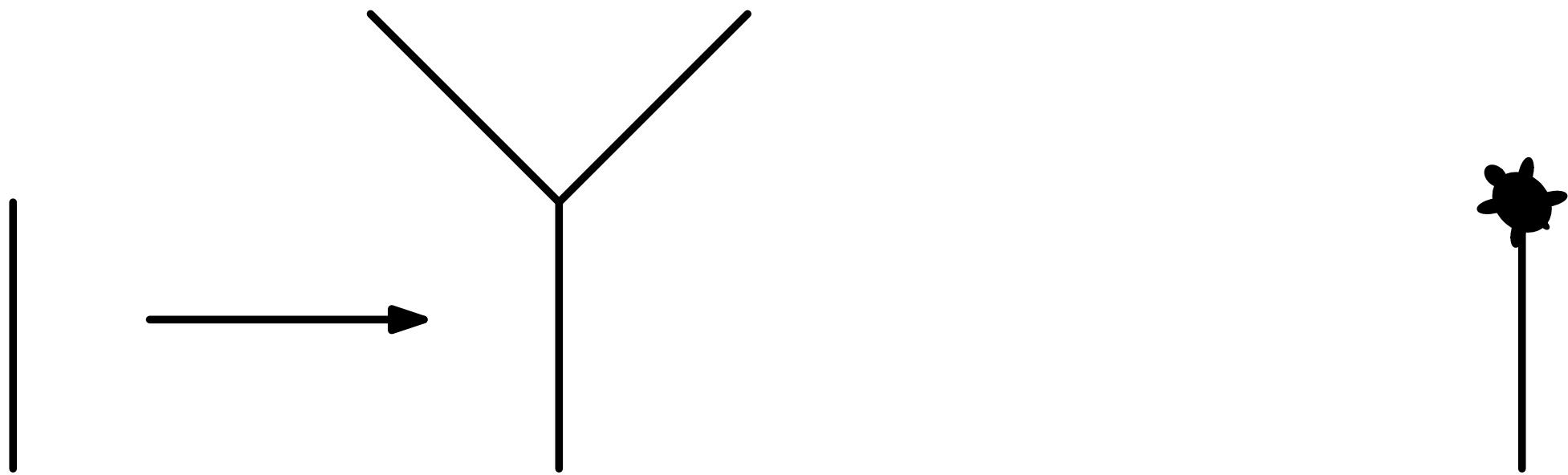
Assuming  $\delta = 60^\circ$ , what shape does the axiom give? What happens when you apply the production rule?



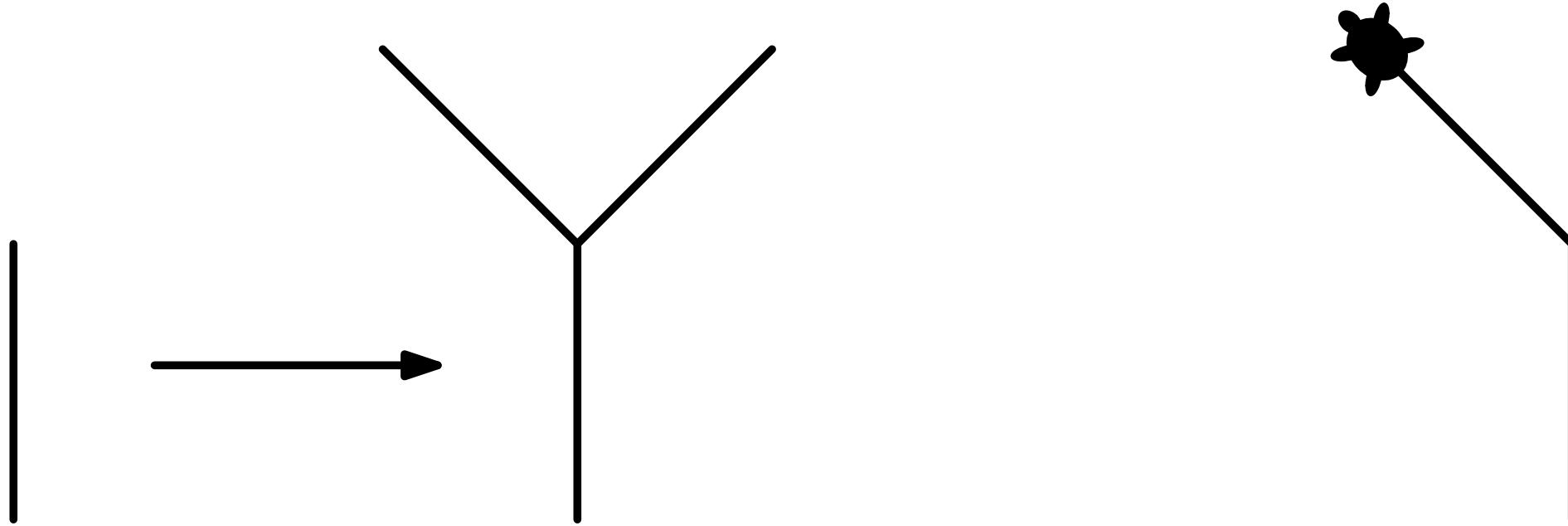
 $F \rightarrow$



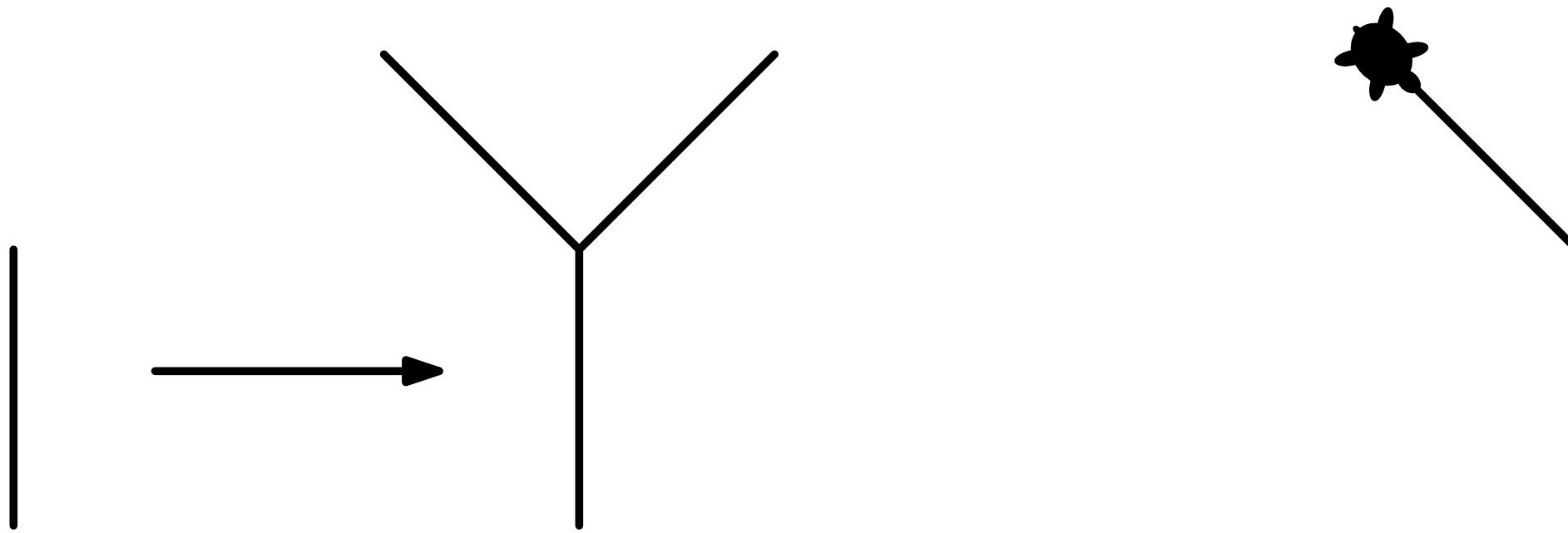
$$F \rightarrow F$$



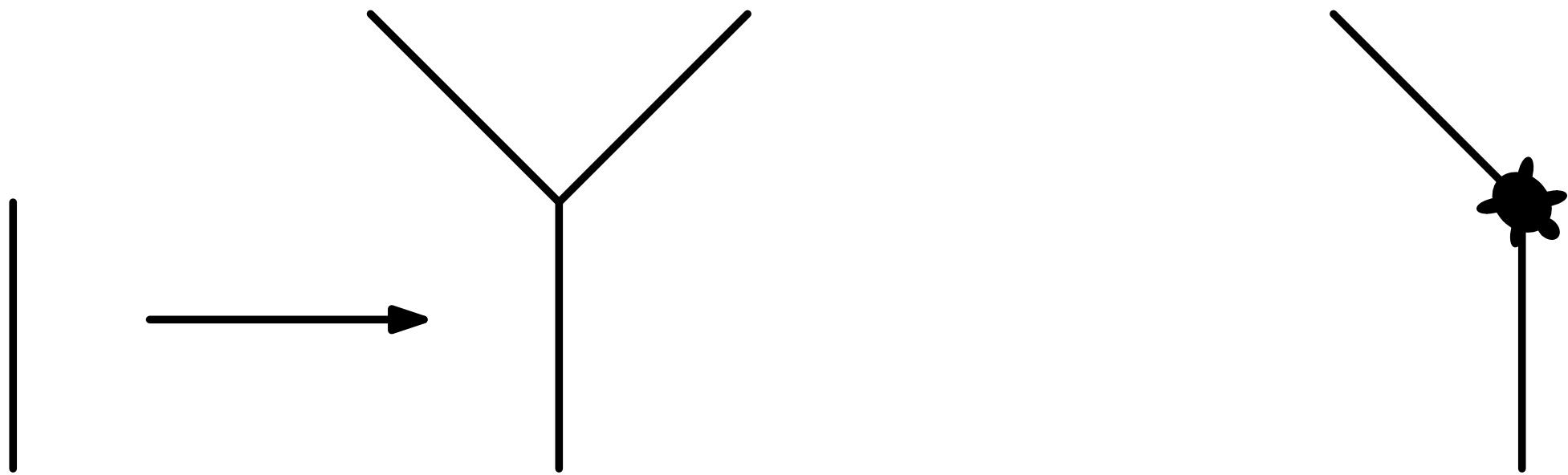
$$F \rightarrow F +$$



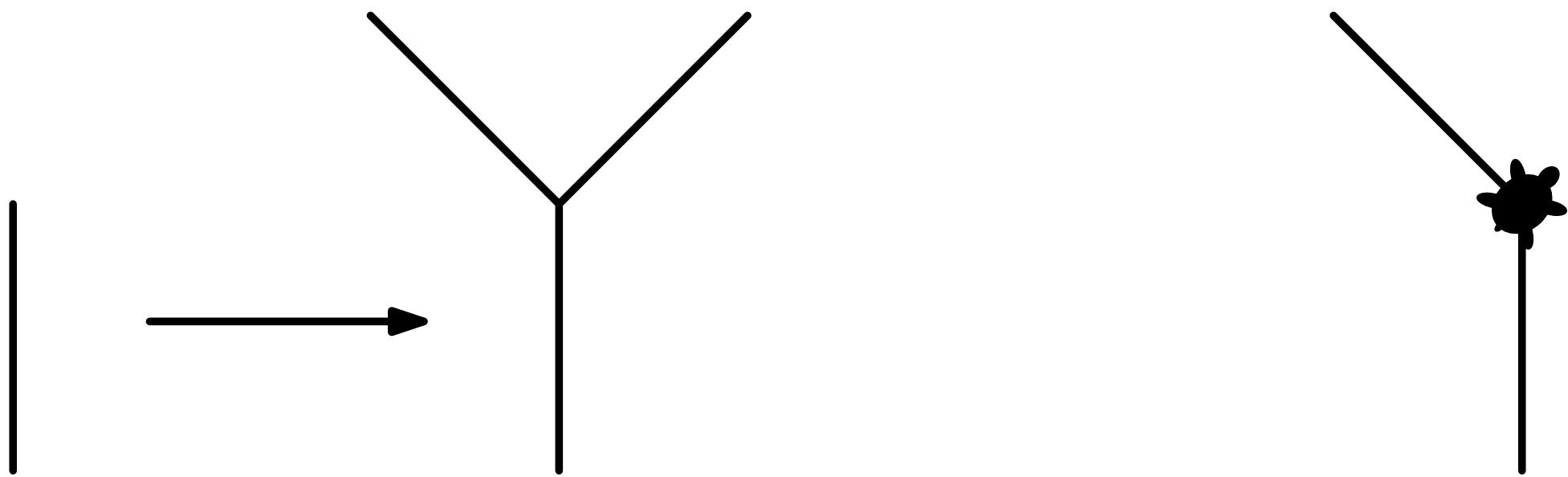
$$F \rightarrow F + F$$



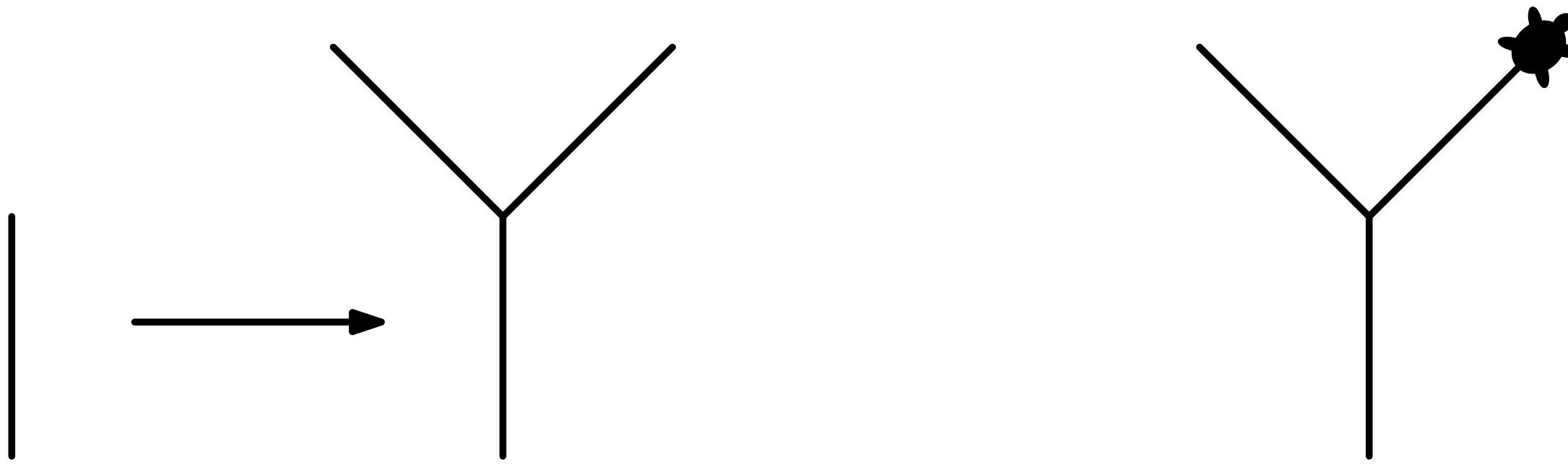
$$F \rightarrow F + F + + + +$$



$$F \rightarrow F + F + + + + f$$



$$F \rightarrow F + F + + + + f + +$$

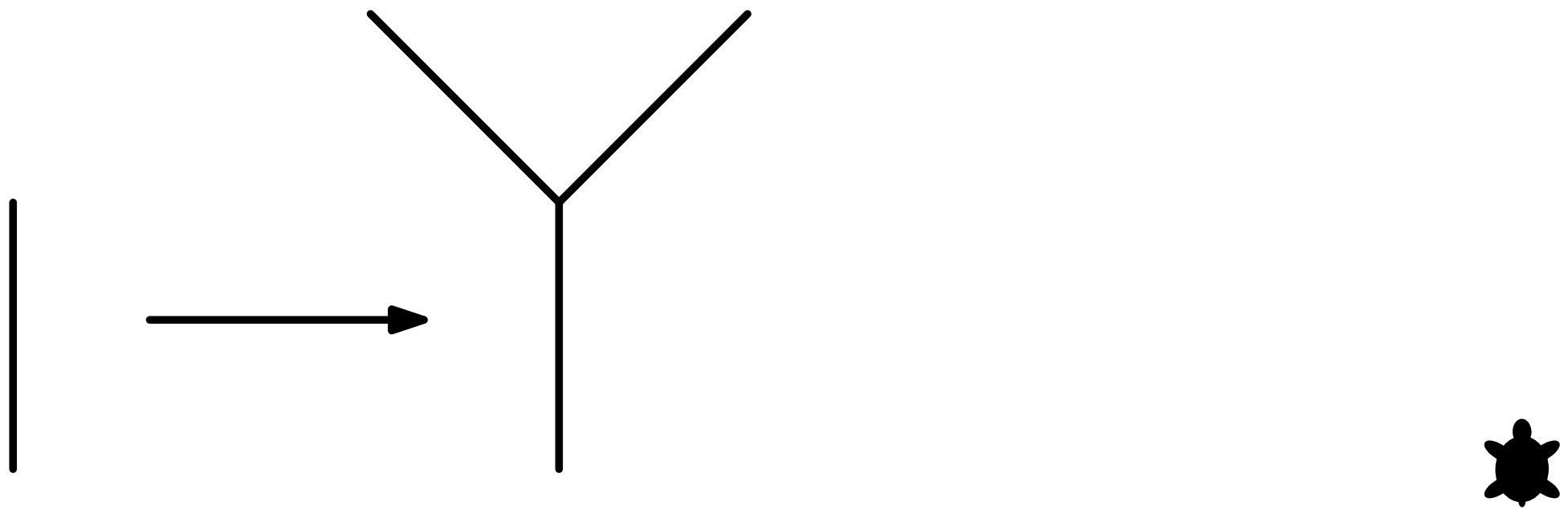


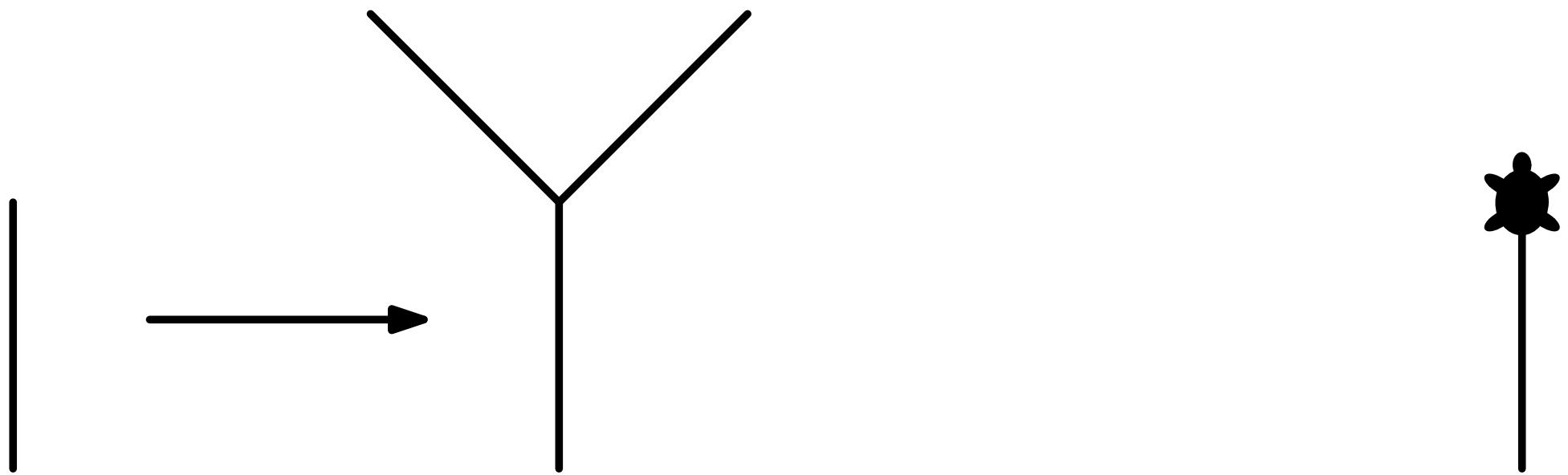
$$F \rightarrow F + F + + + + f + + F$$

Two new symbols:

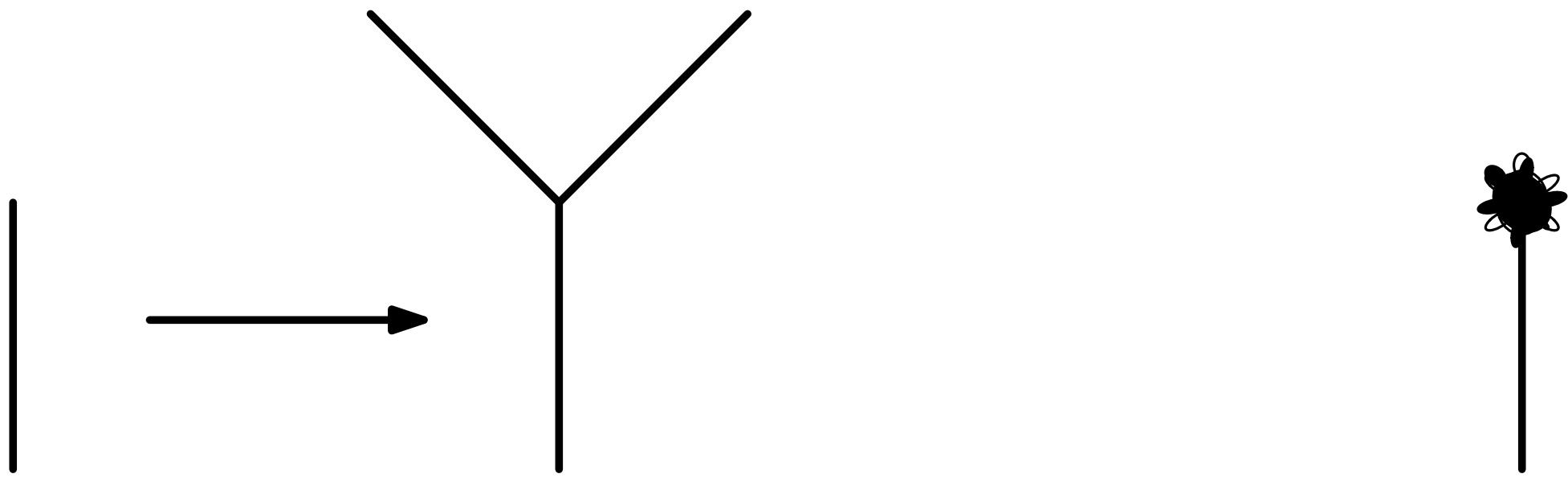
[ : push turtle state onto stack

] : pop turtle state from stack

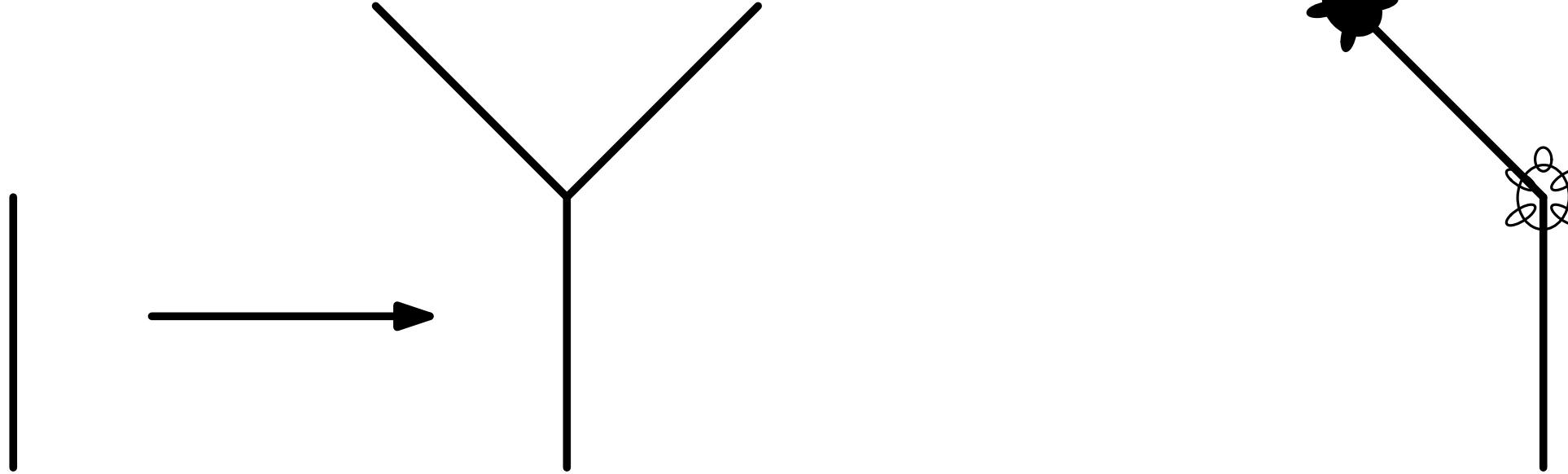
 $F \rightarrow$



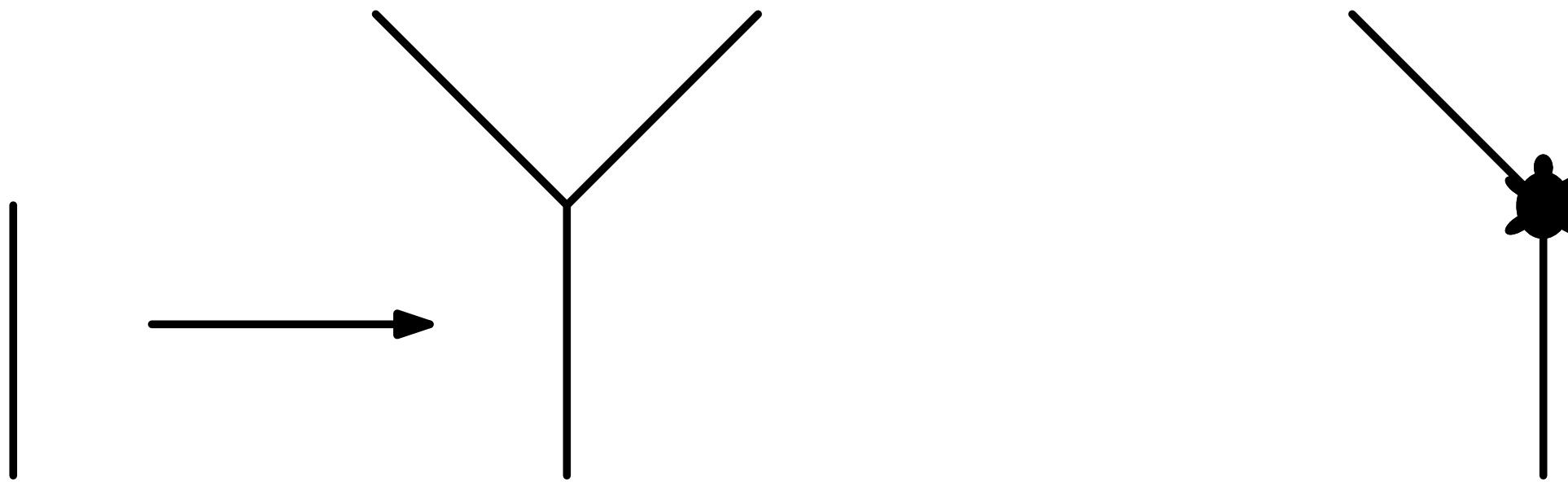
$$F \rightarrow F$$



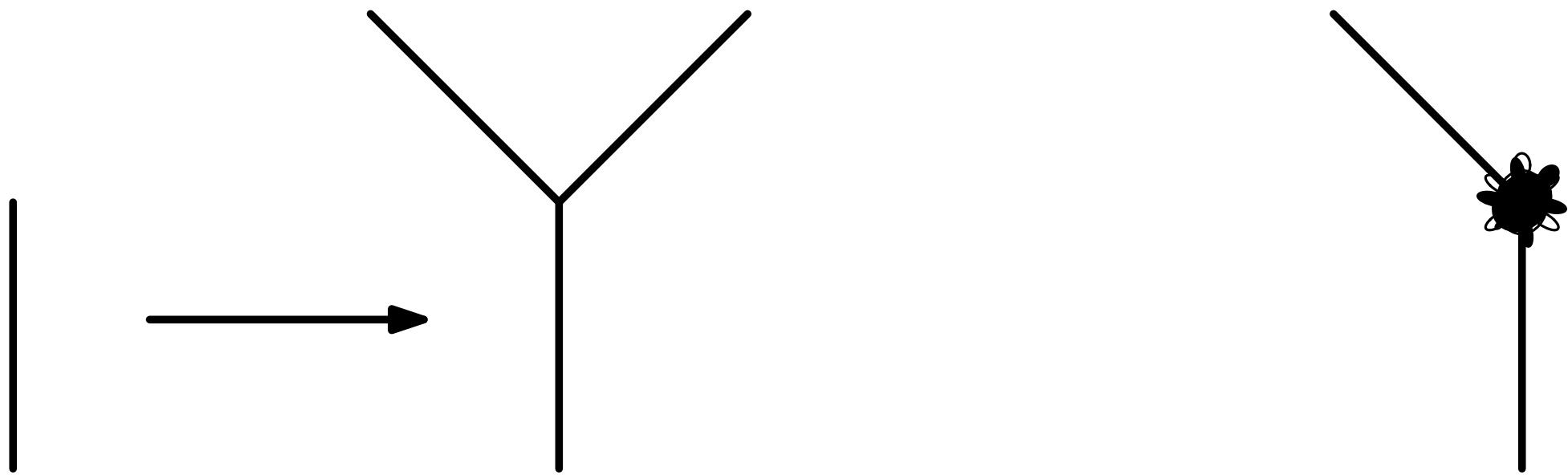
$$F \rightarrow F[+$$



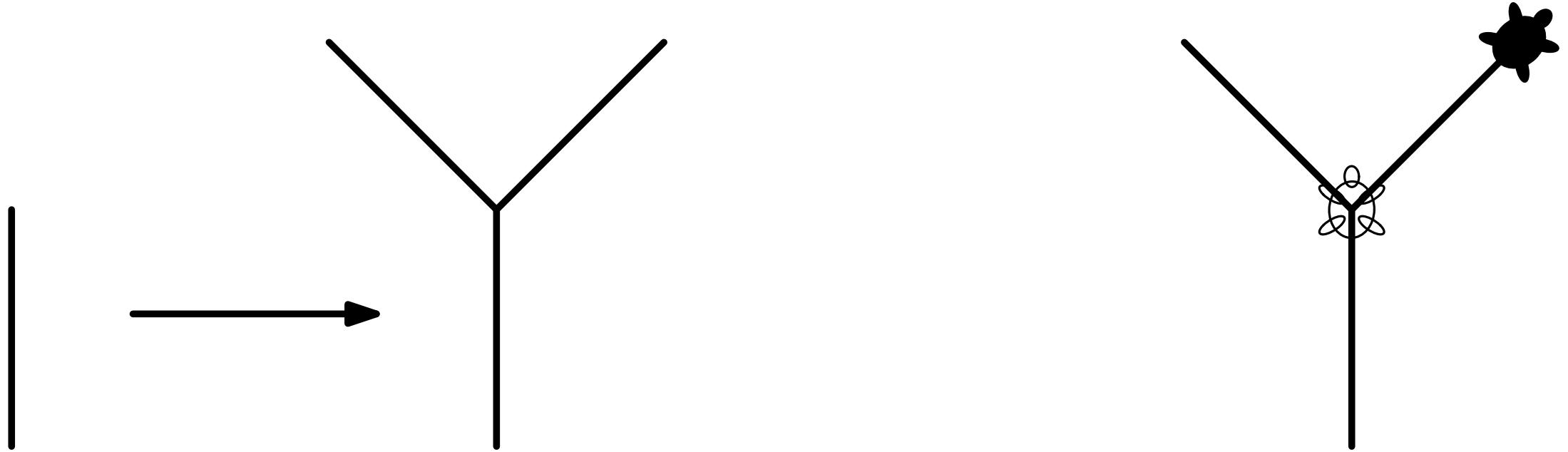
$$F \rightarrow F[+F$$



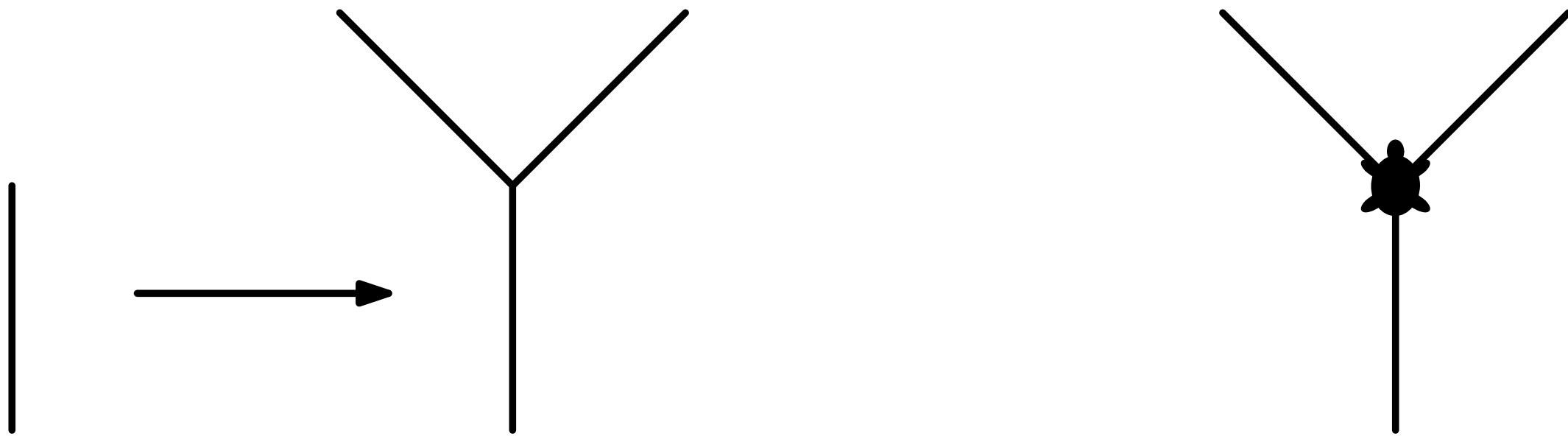
$$F \rightarrow F[+F]$$



$$F \rightarrow F[+F][-$$



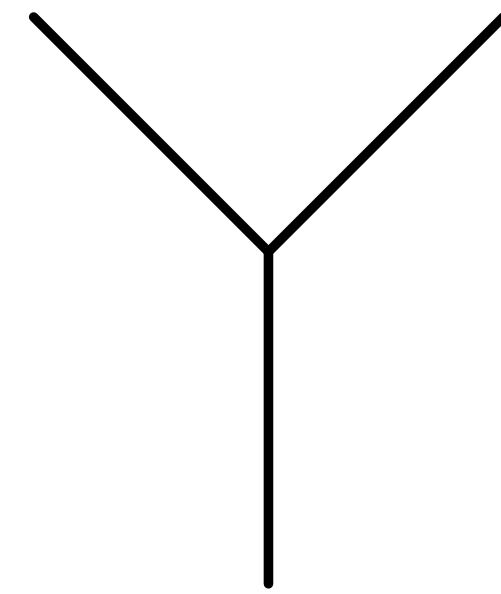
$$F \rightarrow F[+F][-F]$$



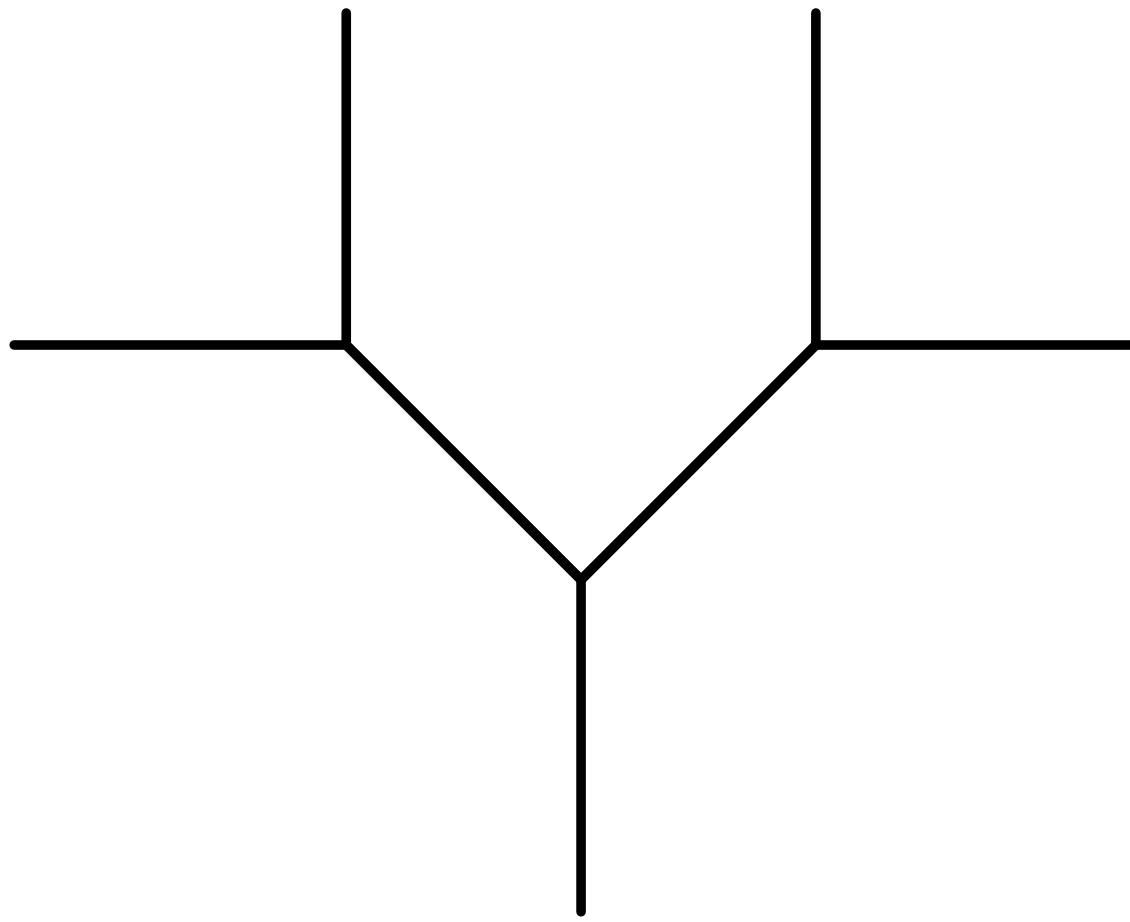
$$F \rightarrow F[+F][-F]$$

$|$

$F$



$F[+F][-F]$



$F[+F][-F][+F[+F][-F]][-F[+F][-F]]$

# Assignment 4

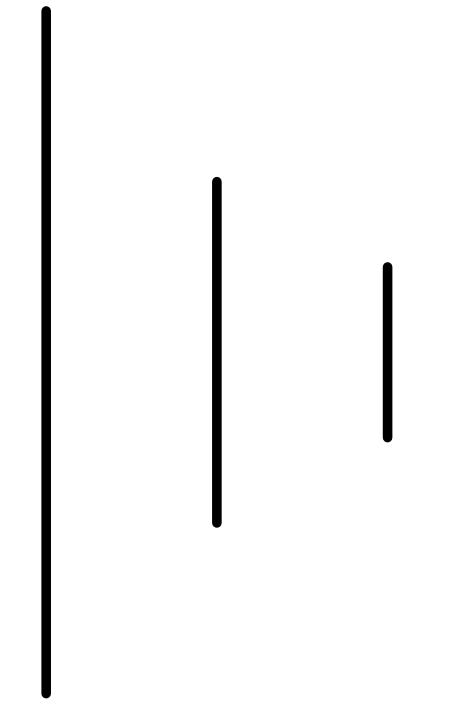
Name two types of shapes that can be suitably drawn by L-systems (e.g. trees), and name two properties of a shape that prevent it from being generated by an L-system.

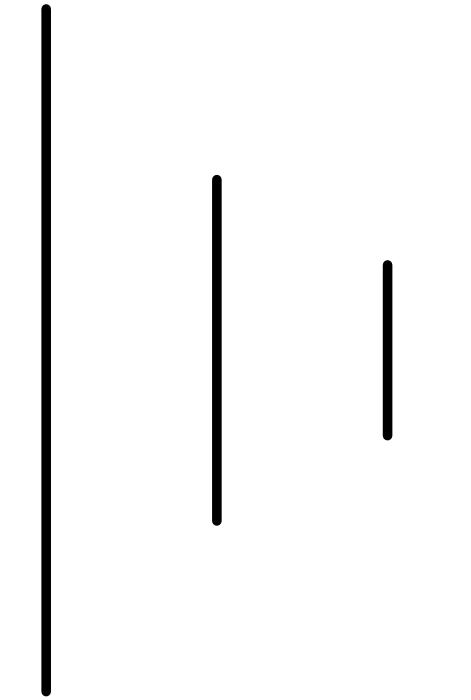


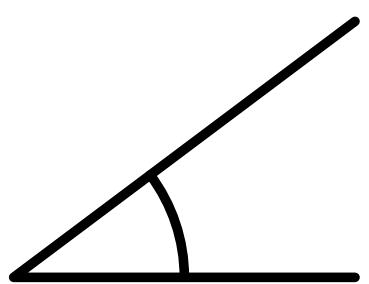
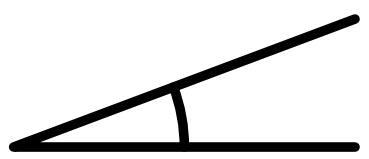


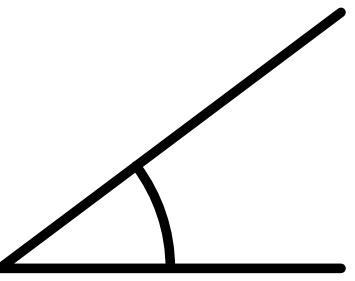
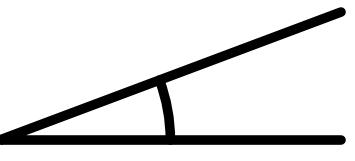
$$F \xrightarrow{0.5} F[-F]$$

$$F \xrightarrow{0.5} F[+F]$$

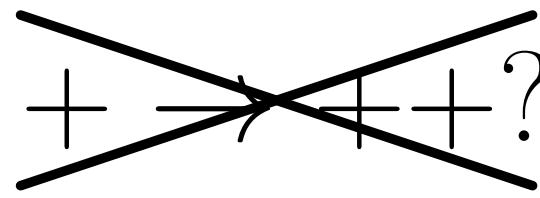
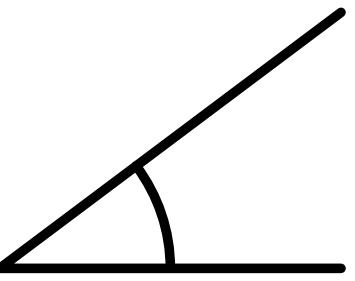
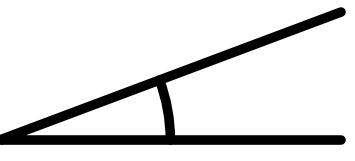


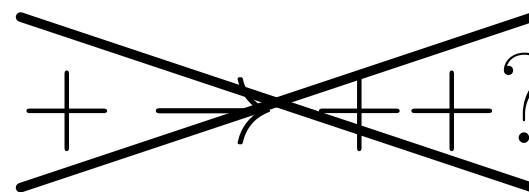
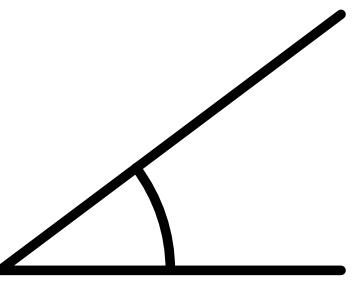
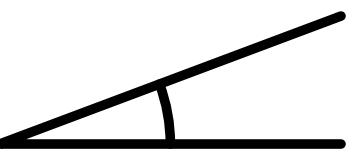

$$F \rightarrow FF$$





$+$   $\rightarrow$   $++?$



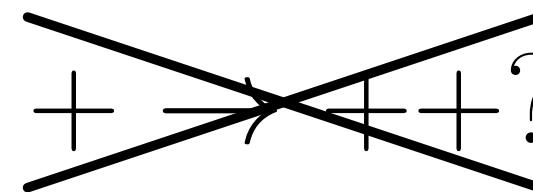
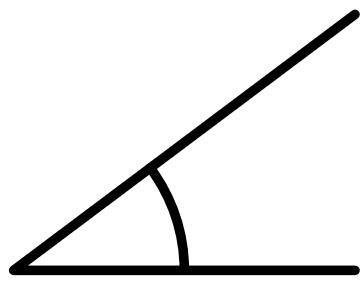
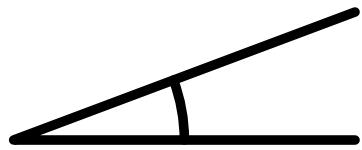


$F(x)$  : forward( $x$ )

$f(x)$  : forward( $x$ ) **(don't draw)**

$+(x)$  : left( $x$ )

$-(x)$  : right( $x$ )



$F(x)$  : forward( $x$ )

$f(x)$  : forward( $x$ ) **(don't draw)**

$+(x)$  : left( $x$ )

$-(x)$  : right( $x$ )

$A(1)$

$A(x) \rightarrow F(x)[+A(x/R)][-A(x/R)]$

$$B < A \rightarrow B$$

$$B \rightarrow A$$

$B < A \rightarrow B$

$B \rightarrow A$

$BAAAAAAA$

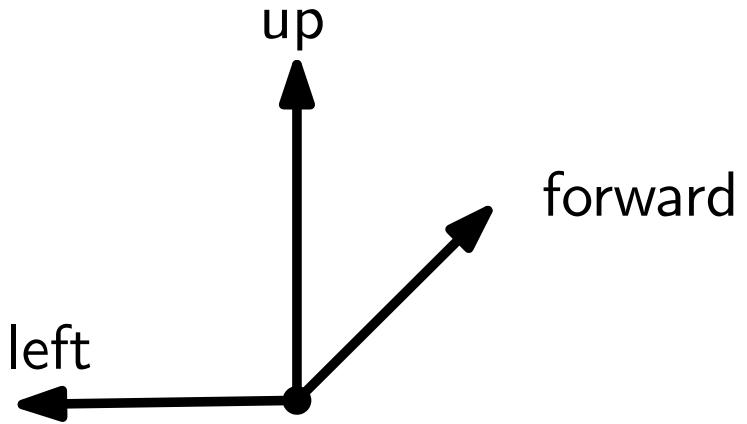
$ABA AAAA$

$AAB AAAA$

$AAABA AAA$

$AAAAB AAA$

$\dots$



+ : turn left

- : turn right

& : pitch down

^ : pitch up

\ : roll left

/ : roll right

# Assignment 5

Design an L-system that draws the Sierpiński triangle.

