

# PSTricks

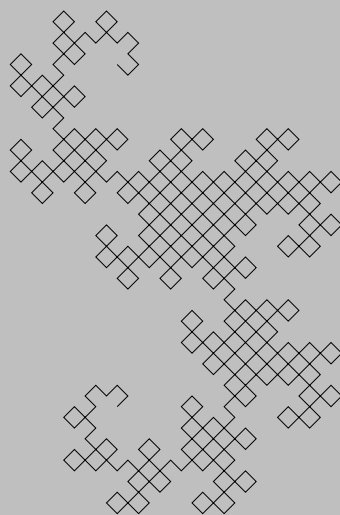
---

## **pst-lsystem**

Creating images, based on the Lindenmeyer-system; v.0.01

---

September 25, 2018



Package author(s):  
**Herbert Voß**

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Usage</b>	<b>3</b>
2.1	Optional arguments . . . . .	3
2.2	Different types for the F-rule . . . . .	3
2.3	Color . . . . .	4
<b>3</b>	<b>Examples</b>	<b>4</b>
3.1	Dragon . . . . .	4
3.2	Hilbert . . . . .	4
3.3	Kochflake . . . . .	4
3.4	Plant 1 . . . . .	5
3.5	Plant 2 . . . . .	6
3.6	Plant 3 . . . . .	6
3.7	Plant 4 . . . . .	6
3.8	Plant 5 . . . . .	7
3.9	Plant 6 . . . . .	7
3.10	Plant 7 . . . . .	8
3.11	Plant 8 . . . . .	8
3.12	Special case . . . . .	9
<b>4</b>	<b>List of all optional arguments for pst-lsystem</b>	<b>10</b>
	<b>References</b>	<b>10</b>

## Abstract

pst-lsystem loads by default the following packages: pst-xkey, and, of course pstricks. All should be already part of your local T<sub>E</sub>X installation. If not, or in case of having older versions, go to <http://www.CTAN.org/> and load the newest version.

Thanks to  
Michel Charpontier, and last but not least <http://mathworld.wolfram.com>.

## 1 Introduction

A L-system or Lindenmayer system is a parallel rewriting system and a type of formal grammar. An L-system consists of an alphabet of symbols that can be used to make strings, a collection of production rules that expand each symbol into some larger string of symbols, an initial »axiom« string from which to begin construction, and a mechanism for translating the generated strings into geometric structures. L-systems were introduced and developed in 1968 by Aristid Lindenmayer, a Hungarian theoretical biologist and botanist at the University of Utrecht. [5]

## 2 Usage

There are no optional arguments for the package:

```
\usepackage{pst-lsystem}
```

There is only one command:

```
\pslsystem [Options] (x,y)
```

If the coordinates for the origin are missing, then (0,0) is assumed. The L-System is defined by the three functions  $F$ ,  $X$ , and  $Y$ . At least one function must be given.

### 2.1 Optional arguments

The following optional arguments for the macro are possible:

<i>name</i>	<i>default</i>	<i>meaning</i>
$F$		Rule $F \rightarrow (F)$
$X$		Rule $X \rightarrow (F, X, Y)$
$Y$		Rule $Y \rightarrow (F, X, Y)$
Start		Startrule $S \rightarrow (F, X, Y)$
Angle	0	Angle for the direction change
N	5	Number of the recursive calls
Ftype	4	How the F-rule should be handled
BaseLength	1mm	The length of a base line, created by the rule F (Forward)
usecolor	0	Which color should be used for the lines.

If the rules contain square brackets then it must be enclosed by braces:  $X=F[-X]$  is wrong and  $X=\{F[-X]\}$  is correct.

### 2.2 Different types for the F-rule

It depends on the given rule(s) what kind of the F-type must be used. There are four possibilities:

```
\ifcase\psk@lsystem@Ftype
  /F { D } def      % type 0
\or % 1
  /F { 0 eq { D } if } def
\or % 2
  /F { dup 0 eq { D } if } def
\or % 3
  /F { pop D } def
```

```
\else % 4
  /F { dup 0 eq { D }{ 1 sub \psk@lsystem@Fnumber {dup} repeat \psk@lsystem@F } ifelse pop
\fi
```

The function D does nothing else than drawing a line in the current direction. Its length is given by the optional parameter BaseLength.

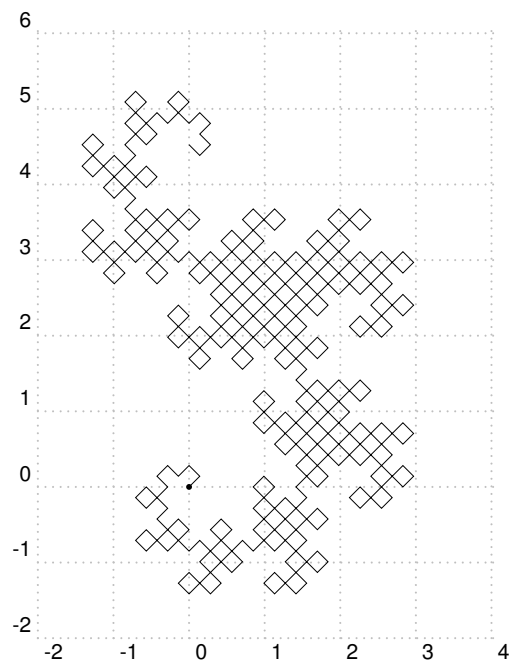
## 2.3 Color

There are four predefined color modes, where mode=0 is no color.

## 3 Examples

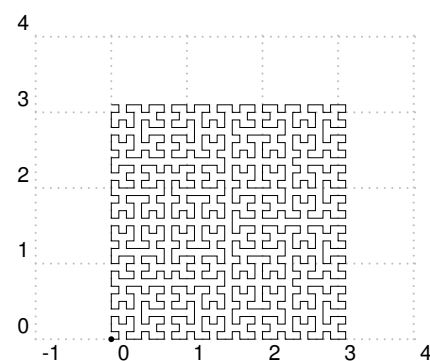
### 3.1 Dragon

```
\begin{pspicture}[showgrid=true](-2,-1.5)(4,6)
\pslsystem[
  X=-FX++FY-,
  Y=+FX--FY+,
  Ftype=1,
  Start=X,
  Angle=45,
  N=9,
  BaseLength=2mm](0,0)
\psdot(0,0)
\end{pspicture}
```



### 3.2 Hilbert

```
\begin{pspicture}[showgrid=true](-1,-0.5)(4,4)
\pslsystem[
  X=-YF+XFX+FY-,
  Y=+XF-YFY-FX+,
  Ftype=3,
  Start=X,
  Angle=90](0,0)
\psdot(0,0)
\end{pspicture}
```



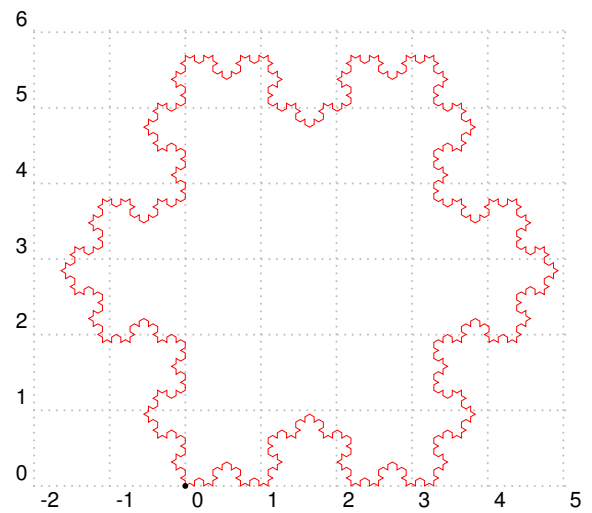
### 3.3 Kochflake

If the rule for Start has more than one function name, one has to repeat the number of iterations, which is N, before every following function, but not for the first one. That is done already internally:

```

\begin{pspicture}[showgrid=true](-2,-.5)(5,6)
\pslsystem[
  Start=F--NF--NF,% repeat number of iterations N
  F=F+F--F+F,
  Angle=60,
  N=4,
  BaseLength=2pt,
  linecolor=red](0,0)
\psdot(0,0)
\end{pspicture}

```

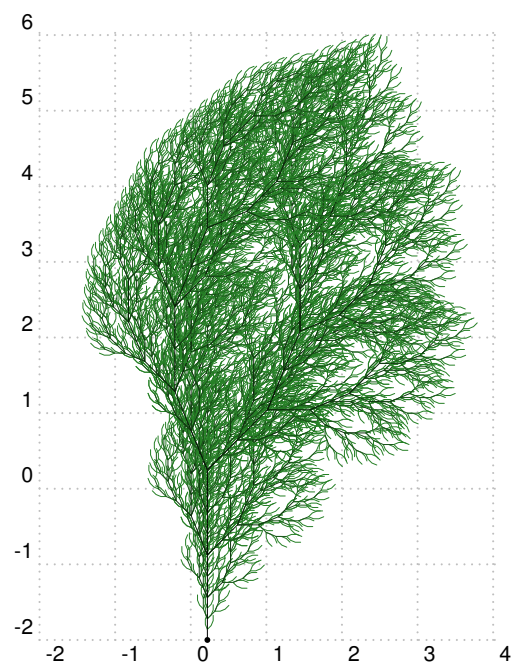


### 3.4 Plant 1

```

\begin{pspicture}[showgrid=true](-2,-2.3)(4,6)
\pslsystem[
  Start=F,
  F={FF-[-F+F+F]+[+F-F-F]},
  Angle=22.5,
  BaseLength=2pt,
  usecolor=4](0,-2)
\psdot(0,-2)
\end{pspicture}

```



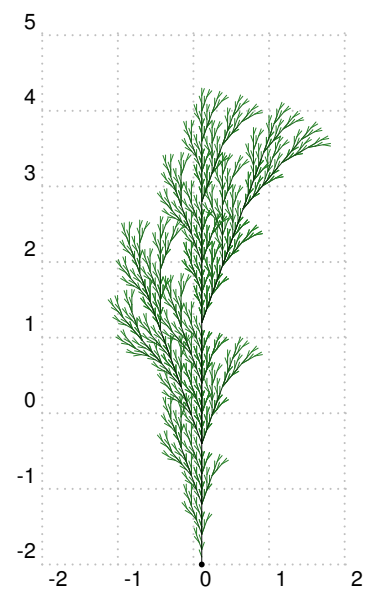
### 3.5 Plant 2

```
\begin{pspicture}[showgrid=true](-3,-2.3)(2,6)
\pslssystem[
  Start=X,
  X={F-[X]+X]+F[+FX]-X},
  F=FF,
  Angle=22.5,
  N=6,
  BaseLength=1.25pt,
  usecolor=3](0,-2)
\psdot(0,-2)
\end{pspicture}
```



### 3.6 Plant 3

```
\begin{pspicture}[showgrid=true](-2,-2.5)(2,5)
\psset{xunit=3}
\pslssystem[
  Start=F,
  F={F[+F]F[-F][F]},
  Angle=20,
  usecolor=3](0,-2)
\psdot(0,-2)
\end{pspicture}
```

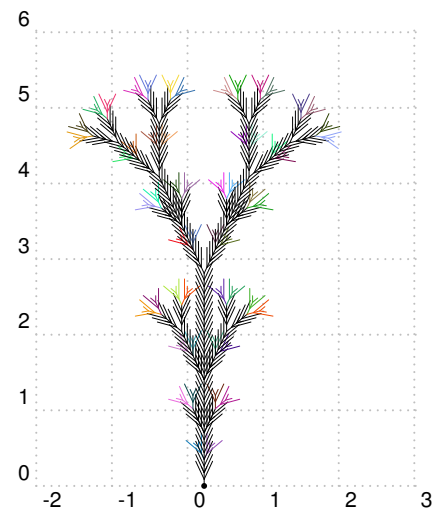


### 3.7 Plant 4

```

\begin{pspicture}[showgrid=true](-2,-0.5)(3,6)
\pslsystem[
  Start=Y,
  X={X[-FFF][+FFF]FX},
  Y={YFX[+Y][-Y]},
  Angle=25,
  N=6,
  Ftype=1,
  BaseLength=2.5pt,
  usecolor=2](0,0)
\psdot(0,0)
\end{pspicture}

```

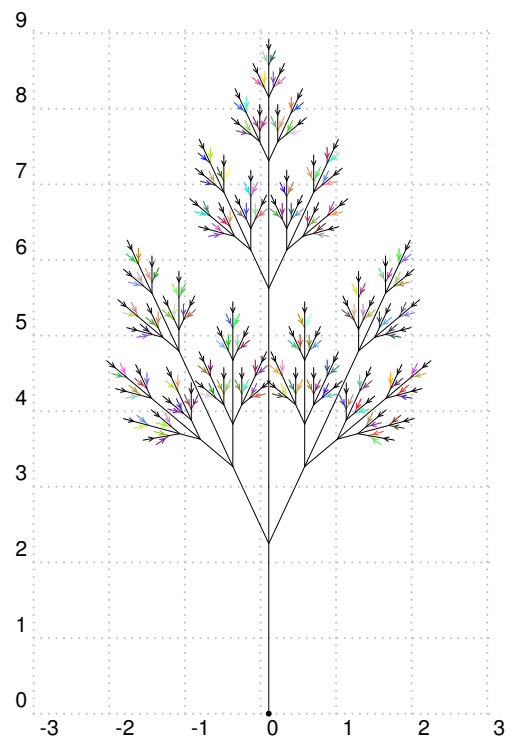


### 3.8 Plant 5

```

\begin{pspicture}[showgrid=true](-3,-.5)(3,9)
\pslsystem[
  Start=X,
  X={F[+X][-X]FX},
  F={FF},
  Angle=25,
  N=7,
  BaseLength=1pt,
  usecolor=2](0,0)
\psdot(0,0)
\end{pspicture}

```

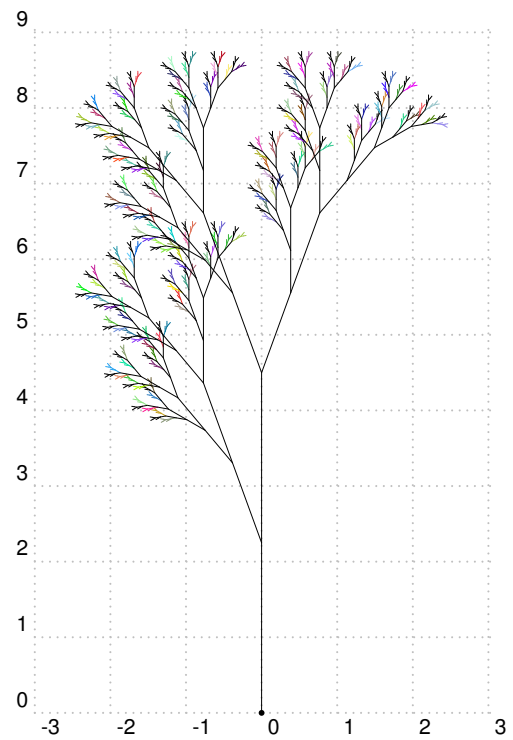


### 3.9 Plant 6

```

\begin{pspicture}[showgrid=true](-3,-.5)(3,9)
\pslssystem[
  Start=X,
  X={F[+X]F[-X]+X},
  F={FF},
  Angle=20,
  N=7,
  BaseLength=1pt,
  usecolor=2](0,0)
\psdot(0,0)
\end{pspicture}

```

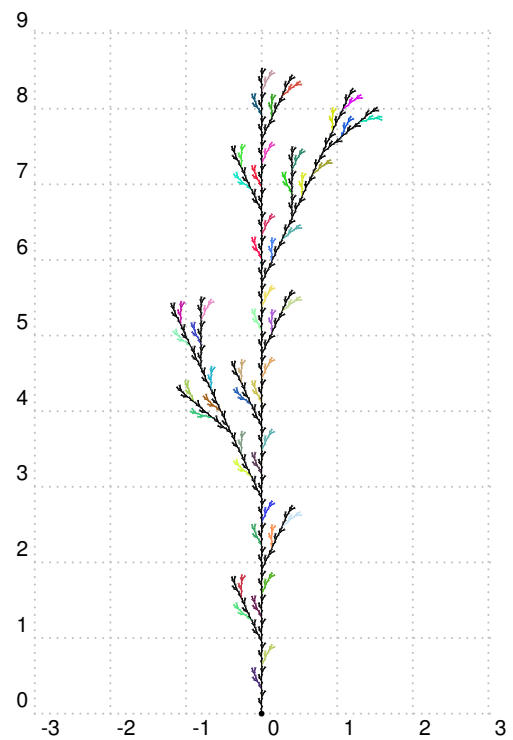


### 3.10 Plant 7

```

\begin{pspicture}[showgrid=true](-3,-.5)(3,9)
\pslssystem[
  Start=F,
  F={F[+F]F[-F]F},
  Angle=25,
  BaseLength=1pt,
  usecolor=2](0,0)
\psdot(0,0)
\end{pspicture}

```



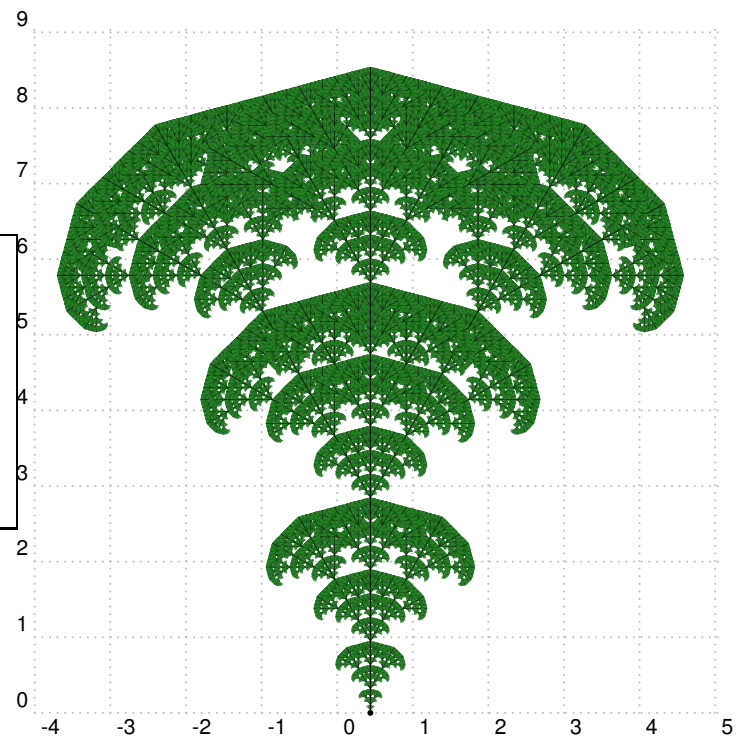
### 3.11 Plant 8



```

\begin{pspicture}[showgrid=true](-4,-.5)(5,9)
\pslsystem[
  Start=F,
  F={F[+F[+F][-F]F][-F[+F][-F]F]F[+F][-F]F},
  Angle=30,
  BaseLength=1pt,
  usecolor=4](0,0)
\psdot(0,0)
\end{pspicture}

```

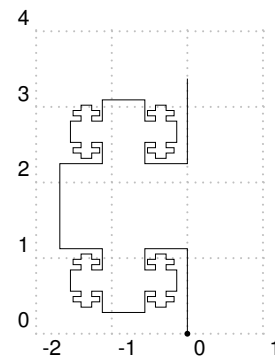


### 3.12 Special case

```

\begin{pspicture}[showgrid=true](-2,-0.5)(1,4)
\pslsystem[
  Start=X,
  F=FF,
  X=F+X-F-X+F,
  Angle=90,
  N=6,
  BaseLength=1pt](0,0)
\psdot(0,0)
\end{pspicture}

```



#### 4 List of all optional arguments for pst-lsystem

Key	Type	Default
F	ordinary	
X	ordinary	
Y	ordinary	
Start	ordinary	
Angle	ordinary	0
N	ordinary	5
Ftype	ordinary	4
N2	ordinary	4
BaseLength	ordinary	1mm
usecolor	ordinary	0
order	ordinary	5

#### References

- [1] Michel Goossens et al. *The L<sup>A</sup>T<sub>E</sub>X Graphics Companion*. 2nd ed. Reading, Mass.: Addison-Wesley Publishing Company, 2007.
- [2] Nikolai G. Kollock. *PostScript richtig eingesetzt: vom Konzept zum praktischen Einsatz*. Vaterstetten: IWT, 1989.
- [3] Herbert Voß. *PSTricks – Grafik für T<sub>E</sub>X und L<sup>A</sup>T<sub>E</sub>X*. 4. Heidelberg/Hamburg: DANTE – Lehmanns, 2007.
- [4] Herbert Voß. *L<sup>A</sup>T<sub>E</sub>X Referenz*. 1. Heidelberg/Hamburg: DANTE – Lehmanns, 2007.
- [5] Wikipedia. *L-system*. Sept. 8, 2018. URL: <https://en.wikipedia.org/wiki/L-system> (visited on 09/22/2018).
- [6] Timothy van Zandt. *PSTricks - PostScript macros for generic T<sub>E</sub>X*. <http://www.tug.org/application/PSTricks>, 1993.

## Index

### A

Angle, 3

### B

BaseLength, 3

### F

F, 3

Ftype, 3

### K

Keyword

– Angle, 3

– BaseLength, 3

– F, 3

– Ftype, 3

– N, 3, 4

– Start, 3, 4

– usecolor, 3

– X, 3

– Y, 3

### M

Macro

– \pslssystem, 3

### N

N, 3, 4

### P

Package

– pst-lssystem, 2

– pst-xkey, 2

– pstricks, 2

\pslssystem, 3

pst-lssystem, 2

pst-xkey, 2

pstricks, 2

### S

Start, 3, 4

### U

usecolor, 3

### X

X, 3

### Y

Y, 3