

symbol

- a symbol is defined as $a < x, y, z >$ where x, y, z are args
- args are evaluated in reverse, so x can depend on y and z , and y can depend on z
- when a symbol is expanded through a rule x, y, z refer to the symbol that is expanded and can be used in other symbols:

$$A < x, y, z > \rightarrow B < x > - < z >$$

- $A \rightarrow$
 - A (all default)
 - $A <; >$ (all default)
 - $A < x; y; z >$ (no change)
 - $A < x;; >$ (y, z default)
 - $A < x >$ (y, z default)
 - $A < \{*, 2\} >$ (default for x scaled)
 - $A < x\{*, 2\} >$ (x scaled)
- symbols
 - $[,]$ (start / end branch)
 - $+, -$ (rotate left / right) \rightarrow default_rotation_angle
 - $, /$ (increase / decrease width) \rightarrow “?? where to save this information??”
 - $!, ?$ (increase / decrease hue) \rightarrow “implement later, do the above first”
 - ABCD (draw branch) \rightarrow default_len
 - abcd (jump) \rightarrow default_len

plant

- is generated out of a lstring
- is generated by a turtle
- the turtle gets generated together with the plant
- each plant has a turtle

turtle

- is responsible for generating the plant out of the lstring
- holds all information of the plant at a given node

Imgui

- all imgui text fields should be `char*`

components

- the L-String is generated via button press or at least separately and first
- Structures are generated and drawn over one or more frames
- they get regenerated if global vars or the viewport changes
 - “maybe they could exist in local coordinate systems, so that they don't need to be regenerated on viewport changes. Maybe this can be done with blending?”
-