If you can code, you can CAD!

Kyle J. Davis
Agenda

• The CAD Landscape
• Parametric Design
• OpenSCAD
• Model to Physical Thing
• Q&A
About me

- I work for Amazon
  - This talk has nothing to do with my work
  - Talk to me at the Amazon booth about Linux
- Coding for a long time
- CAD for slightly less time
- Picked up 3D printing at the start of the pandemic
The CAD Landscape
Popular CAD Software

- Autodesk Fusion360
- Dassault Systèmes SolidWorks
- Trimble SketchUp
- Autodesk TinkerCAD
- FreeCAD / LGPL-2.0-or-later / www.freecadweb.org
- OpenSCAD / GPL-2.0-or-later / openscad.org
Parametric Design
Parametric

of, or relating, to a parameter, mathematical, or statistical variable
Instead of *drawing* you’re *specifying*

- Traditional CAD: A combination of clicking, placing, entering values.
- Parametric CAD: A full, or complete reliance, on specifying values and calculating positions.
The buttons are too close together

1. Click each button
2. Change the width of the button by dragging
3. Adjust the spacing between each button

1. Change the button spacing variable
Draw this organic shape

1. Use the mouse to set points
2. Compile
3. Repeat $\infty$ ***
OpenSCAD
Basic Concepts

• Variables
  • Numbers, Booleans, Strings, Ranges, Vectors, and undef

• Functions
  • Take in parameters, returns values

• Modules
  • Draws in geometry
  • Has children, but returns nothing
Simple Example

```
credit_card_dimensions = [85.6, 53.98, 0.76];

card();
translate(diag_position(1,credit_card_dimensions))
card();
translate(diag_position(2,credit_card_dimensions))
card();

function diag_position(n, dims) = [dims.x * n, dims.y * n, 0];

module_card()
  cube(credit_card_dimensions);
```
credit_card_dimensions = [85.6, 53.98, 0.76];

card();
translate(diag_position(1, credit_card_dimensions))
card();
translate(diag_position(2, credit_card_dimensions))
card();

function diag_position(n, dims) =
    [dims.x * n, dims.y * n, 0];

module card()
cube(credit_card_dimensions);
Iteration

credit_card_dimensions = [85.6, 53.98, 0.76];

for (i = [1:1:10])
    translate(diag_position(i, credit_card_dimensions))
        card();

function diag_position(n, dims) =
    [dims.x * n, dims.y * n, 0];

module card()
    cube(credit_card_dimensions);
<table>
<thead>
<tr>
<th>Two Dimensions</th>
<th>Three Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circle</td>
<td>Sphere / Cylinder</td>
</tr>
<tr>
<td>Square</td>
<td>Cube</td>
</tr>
<tr>
<td>Polygon</td>
<td>Polyhedron</td>
</tr>
<tr>
<td>Imported Geometry</td>
<td></td>
</tr>
<tr>
<td>(DXF/SVG for 2D,</td>
<td></td>
</tr>
<tr>
<td>STL/OFF/AMF/3MF for 3D)</td>
<td></td>
</tr>
<tr>
<td>→</td>
<td>Linear Extrude</td>
</tr>
<tr>
<td>→</td>
<td>Rotate Extrude</td>
</tr>
</tbody>
</table>
2D vs 3D
Cube Example

dimensions = [5, 10];
height = 7;

linear_extrude(height)
square(dimensions);

dimensions = [5, 10, 7];
cube(dimensions);
$fn = 30;
dimensions = [10,10,10];
sphere_diameter_multiple = 1.2;

module my_cube()
  cube(dimensions, center = true);
module my_sphere()
  sphere(d=dimensions.x * 1.2);

intersection() {
  my_cube();
  my_sphere();
}

difference() {
  my_cube();
  my_sphere();
}

union() {
  my_cube();
  my_sphere();
}
Move and alter

- Translate
- Rotate / Scale / Resize / Mirror / Multimatrix
- Color
Changing Shapes

Offset

Hull

Minkowski
Built-in IDE

- Code Editor
- Viewer
- Console
Using VS Code
With OpenSCAD for VS Code Plugin
Model to Physical Thing
Process overview

OpenSCAD → SCAD → Compile → STL → Slice → gcode → Print → Your Thing

Preview / Render / Export
Slicing
Triangles to motion

- STL
- Object into layers
- Each layer into lines
- Specific to your printer
- Results in gcode (RS-274)
- Tools
  - Cura / LGPLv3 / github.com/Ultimaker/Cura
  - Slic3r & PrusaSlicer / SuperSlicer / AGPL-3.0 / github.com/prusa3d/PrusaSlicer
Gcode to printer

- Save gcode from slicer
- Good: Put gcode on a (micro) SD card and print from LCD
- Better: Use Octoprint (https://octoprint.org)
  - Written in Python
  - AGPL-3.0
  - Usually on a Raspberry Pi and connected to print via USB
  - Lots of great plugins - webcam, bed level, etc.
Printing

• Every printer has a “main board”
• Usually: 8 or 32 bit microcontroller
• Converts gcode to movement of motors / fans / accessories
• Firmware:
  • Marlin / GPLv3 / marlinfw.org
  • RepRap / GPLv3 / www.reprapfirmware.org
  • Klipper / GPLv3 / www.klipper3d.org
Q&A
Thank you!

Connect with me:
@stockholmux (Twitter & Github)
On Printables:
printables.com/social/260544-kyle