

DISEÑO 3D

TUBOT 2016



Universidad
de Alcalá



Departamento de
electrónica

Introducción



OpenSCAD vs FreeCAD

OpenSCAD - Wades_Gears.scad

```
File Edit Design View Help
// Wade's Extruder Gears using Parametric Involute Bevel and Spur Gears by GregFr
// by Nicholas C. Lewis (A RepRap Breeding Program)
//
// It is licensed under the Creative Commons - GNU GPL license.
// Å© 2010 by Nicholas C. Lewis
// http://www.thingiverse.com/thing:4305

use <mendel_misc.inc>
use <parametric_involute_gear_v5.0.scad>

meshing_double_helix();

//WadesL(); //this module call will make the large gear
//WadesS(); //this module call will make the small gear

module WadesL(){
  //Large WADE's Gear
  //rotate([0,0,-2])translate([0,0,0])color([ 100/255, 255/255, 200/255])import
  ("39117p.stl");
  difference(){
    gear (number_of_teeth=39,
          circular_pitch=268,
          gear_thickness = 5,
          rim_thickness = 7,
          rim_width = 3,
          hub_thickness = 13,
          hub_diameter = 25,
          bore_diameter = 8,
          circles=0,
          twist = 0);
    translate([0,0,6])rotate([180,0,0])m8_hole_vert_with_hex(100);
  }
}

module WadesS(){
  //small WADE's Gear
  //rotate([180,0,-23.5])translate([-10,-10,-18])color([ 100/255, 255/255, 200
  5])import_stl("wades_gear.stl");
  difference(){
    gear (number_of_teeth=11,
          circular_pitch=268,
          gear_thickness = 9,
          rim_thickness = 9,
          hub_thickness = 18,
          hub_diameter = 20,
          bore_diameter = 5,
          circles=0,
          twist = 0);
    translate([0,-5,16])cube([5.5,2.3,9],center = true);
    translate([0,0,14])rotate([0,90,-90])cylinder(r=1.7,h=20);
  }
}

Viewport: translate = [ 24.54 -3.37 1.98 ], rotate = [ 50.80 0.00 322.40 ], distance = 617.28
```

FreeCAD

Combo View
Model Tasks

Labels & Attributes

Application

- Assembly
 - front-frame
 - claw
 - frame-rear-servos
 - battery-pack-holder
 - left-wheel
 - right-wheel
 - o-rings
 - 0129
 - Electronics

Property Value

View/Data

Assembly: 1

Facets: 3682
Volumes: 3
Total rendering time: 0 hours, 3 minutes, 4 seconds
Rendering finished.

345.70 x 198.88 mm



OpenSCAD

- Introducción
- Entorno
- Programación
- Ejemplos



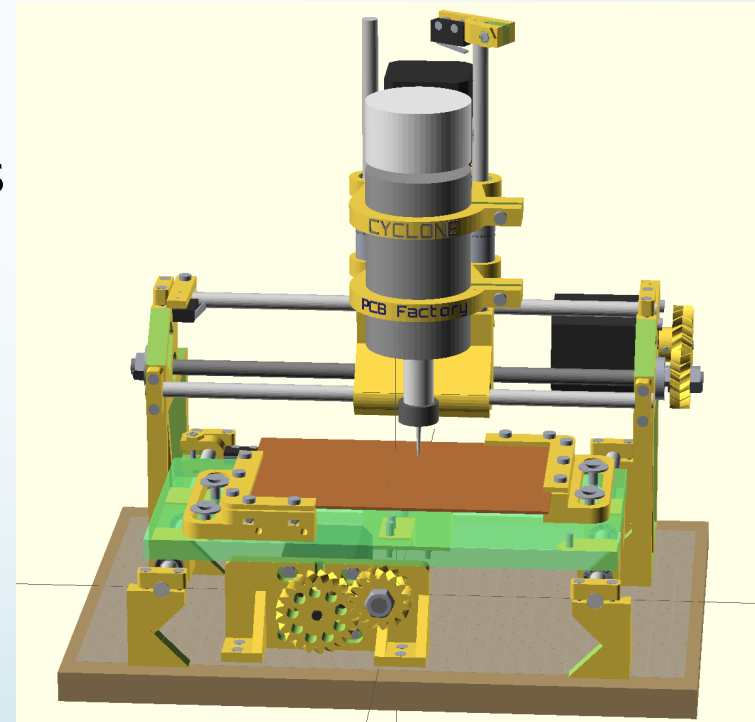
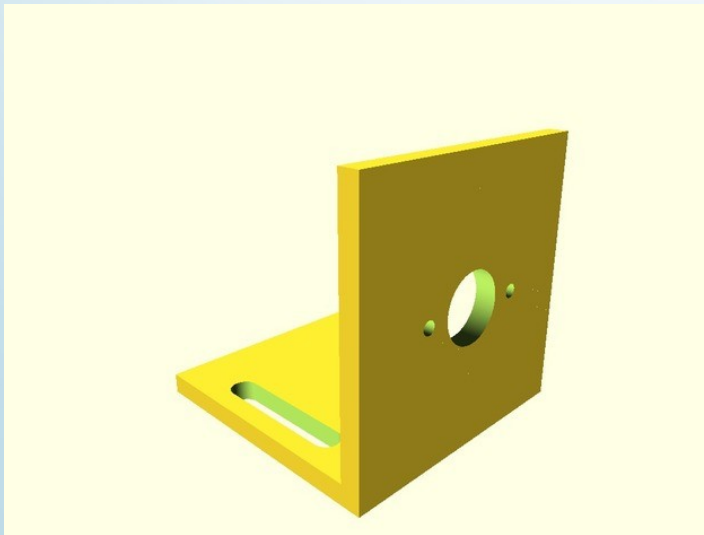
Universidad
de Alcalá



Departamento de
Electrónica

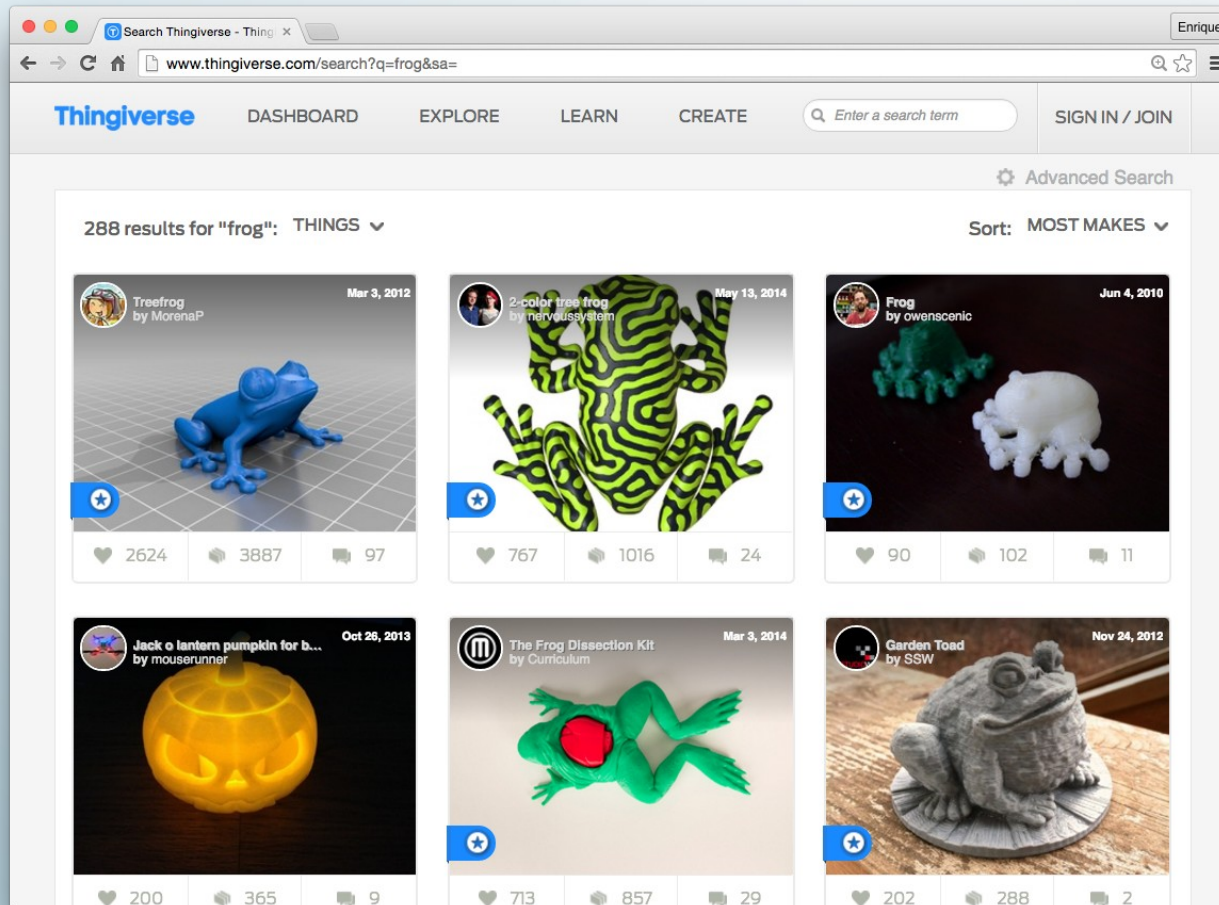
Introducción

- Software libre GPL
- Disponible para Linux, Windows
- Diseño por programación
- www.openscad.org



Introducción

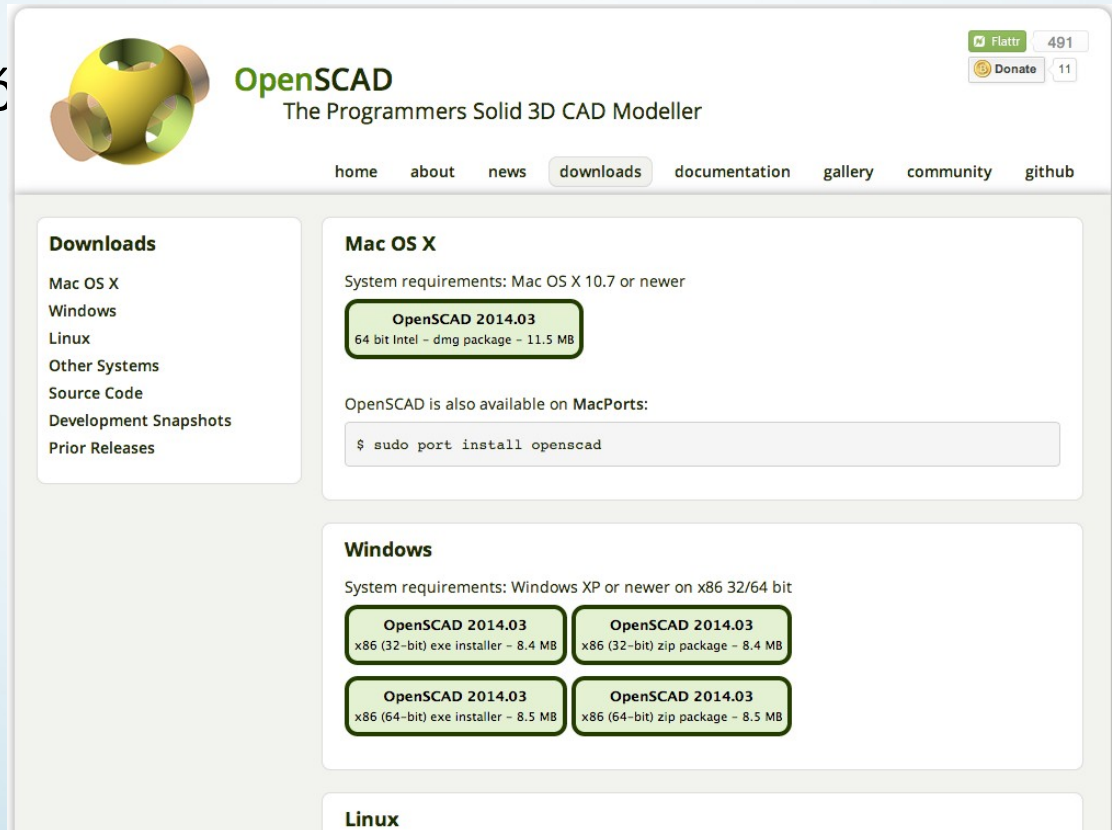
- www.thingiverse.com
 - Se pueden encontrar modelos compilados y fuentes



Introducción

- www.openscad.org

- Descargas
- Documentación
 - Manual
 - Librerías
- Galería
- ...



The screenshot shows the OpenSCAD website. At the top left is the OpenSCAD logo, a 3D sphere with colored segments. To its right is the text "OpenSCAD The Programmers Solid 3D CAD Modeller". In the top right corner, there are buttons for "Flattr" (491) and "Donate" (11). Below the logo and text is a navigation menu with links for "home", "about", "news", "downloads" (which is highlighted), "documentation", "gallery", "community", and "github".

The main content area is divided into sections. On the left is a "Downloads" sidebar with links for "Mac OS X", "Windows", "Linux", "Other Systems", "Source Code", "Development Snapshots", and "Prior Releases".

The "Mac OS X" section has the heading "Mac OS X" and the text "System requirements: Mac OS X 10.7 or newer". It features a button for "OpenSCAD 2014.03" with the subtext "64 bit Intel - dmg package - 11.5 MB". Below this, it says "OpenSCAD is also available on MacPorts:" followed by a code block:

```
$ sudo port install openscad
```

The "Windows" section has the heading "Windows" and the text "System requirements: Windows XP or newer on x86 32/64 bit". It contains four buttons for "OpenSCAD 2014.03":

- x86 (32-bit) exe installer - 8.4 MB
- x86 (32-bit) zip package - 8.4 MB
- x86 (64-bit) exe installer - 8.5 MB
- x86 (64-bit) zip package - 8.5 MB

The "Linux" section is partially visible at the bottom of the screenshot.

Introducción

- Documentación

- http://en.wikibooks.org/wiki/OpenSCAD_User_Manual



The screenshot shows the Wikibooks interface for the 'OpenSCAD User Manual' page. At the top right, there are links for 'Create account', 'Log in', and 'Sandbox'. Below this is a navigation bar with 'Book', 'Discussion', 'Read', 'Edit', and 'View history' tabs, and a search box. The main content area features a heading for 'OpenSCAD User Manual' and a sub-heading for 'Introduction [edit]'. The text describes OpenSCAD as a software for creating solid 3D CAD objects, available for GNU/Linux, MS Windows, and Apple OS X. It contrasts OpenSCAD with Blender, highlighting its focus on CAD aspects. A 3D model of a yellow sphere with three holes is shown on the right. At the bottom, it mentions two main modelling techniques: constructive solid geometry (CSG) and extrusion of 2D.

WIKIBOOKS
Open books for an open world

Main Page
Help
Browse
Cookbook
Wikijunior
Featured books
Recent changes
Donations
Random book
Using Wikibooks

Community
Reading room
Community portal
Bulletin Board
Help out!
Policies and guidelines
Contact us

Create account Log in Sandbox

Book Discussion Read Edit View history Search

Here at Wikibooks we're proud of our **Featured books** - the very best of Wikibooks. Vote for books that you think are worthy of being Featured Books at **Featured book nominations**

OpenSCAD User Manual

Introduction [edit]

OpenSCAD is a software for creating solid 3D CAD objects. It is [free software](#) and available for [GNU/Linux](#), MS Windows and Apple OS X.

Unlike most free software for creating 3D models (such as the well-known application [Blender](#)), OpenSCAD does not focus on the artistic aspects of 3D modelling, but instead focuses on the CAD aspects. So it might be the application you are looking for when you are planning to create 3D models of machine parts, but probably is not what you are looking for when you are more interested in creating computer-animated movies.

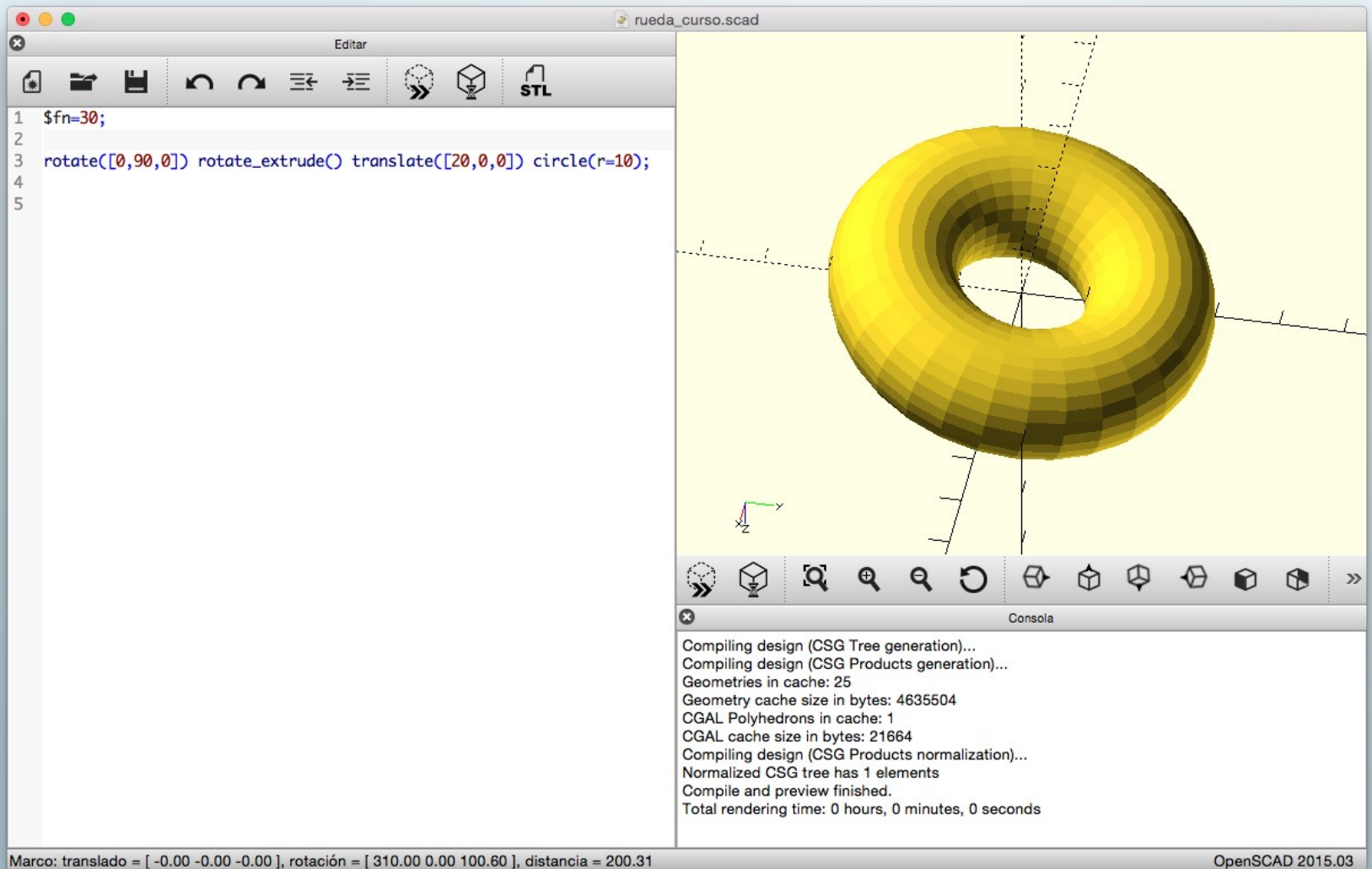
OpenSCAD is not an interactive modeller. Instead it is something like a 3D interpreter that reads in a script file that describes the object and renders the 3D model from the script file. This gives you (the designer) full control over the modelling process and enables you to easily change any step in the modelling process, or even to produce designs that are defined by configurable parameters.

OpenSCAD provides two main modelling techniques: First. constructive solid geometry (CSG) and second. extrusion of 2D

Search this book

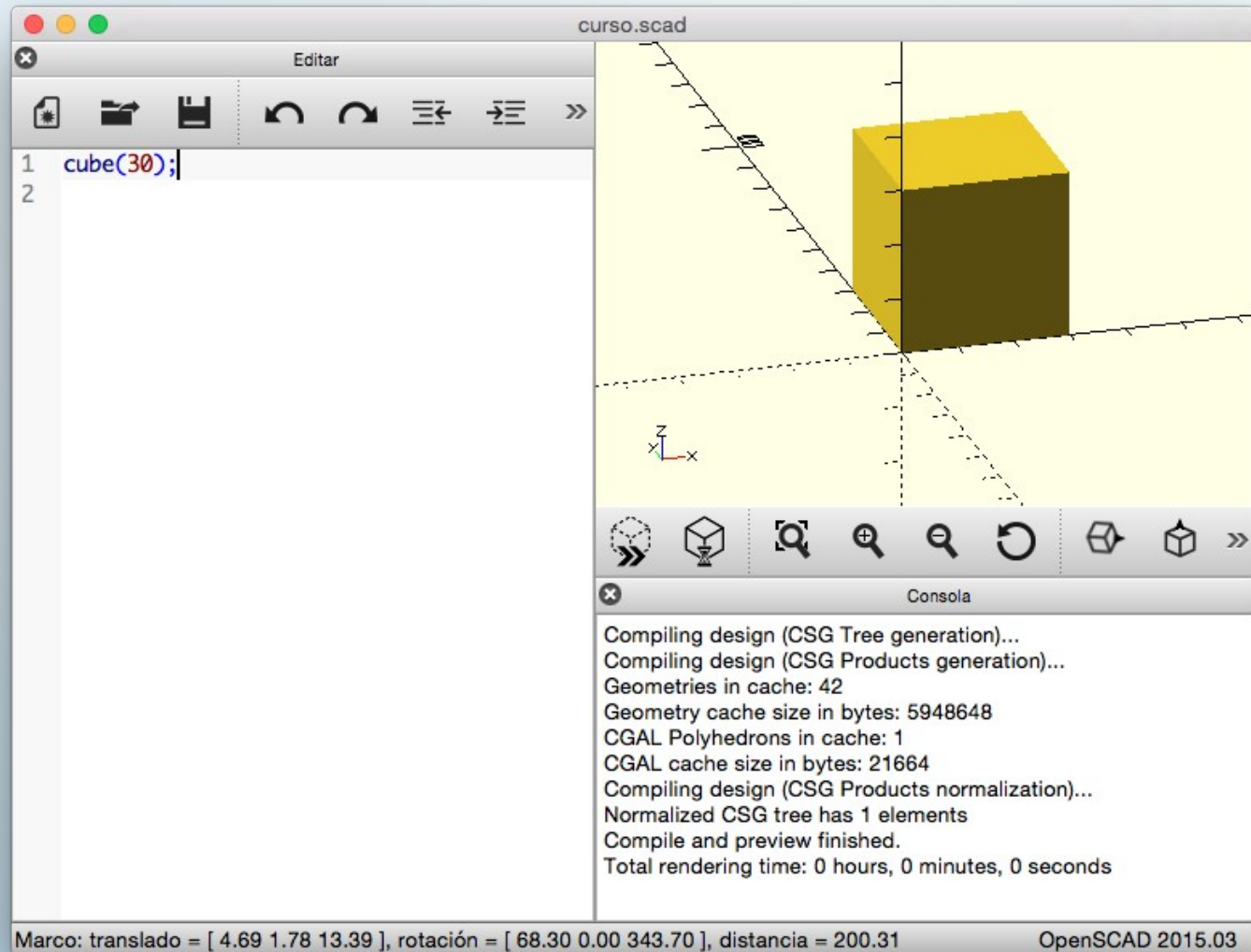


Entorno



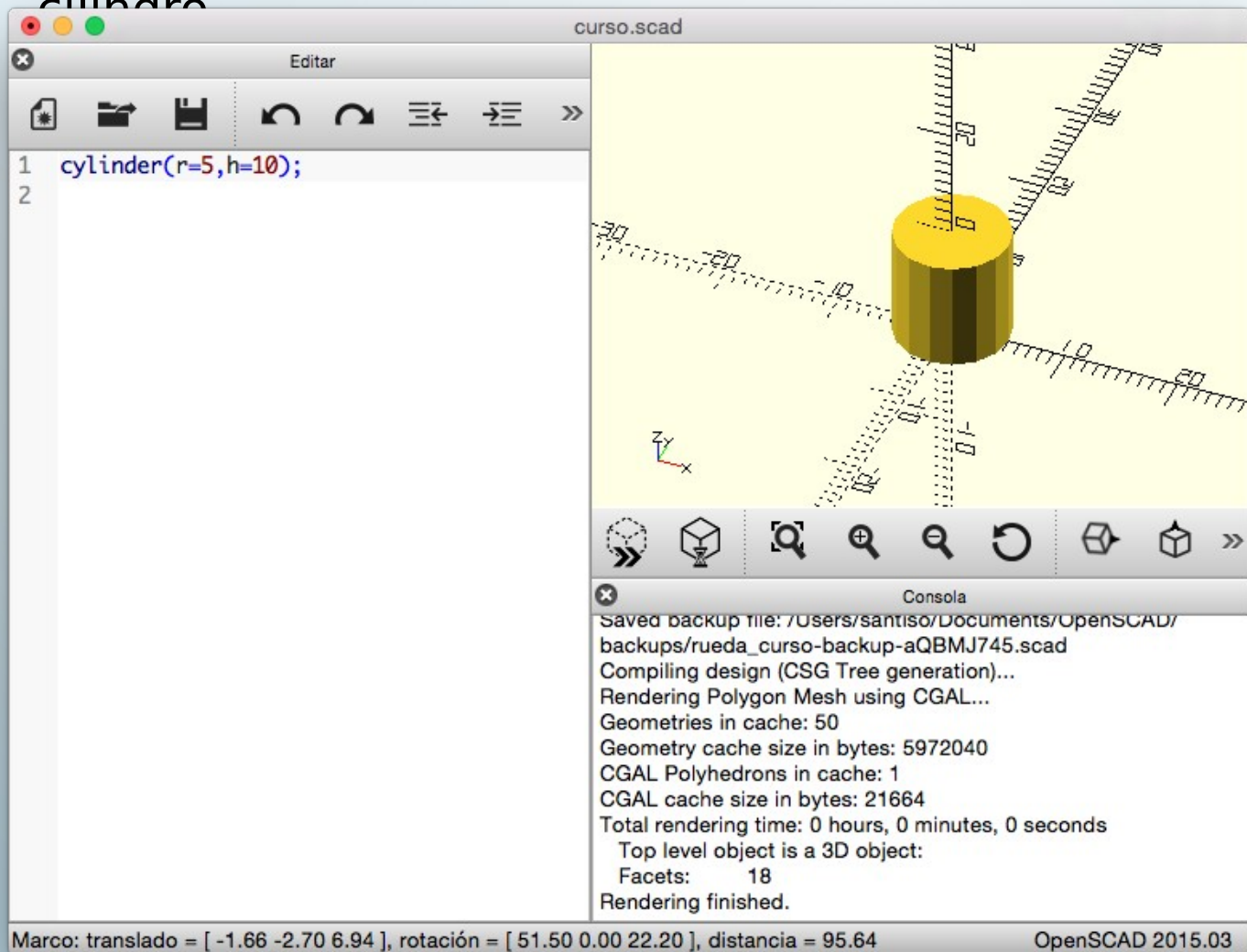
Programación

- Diseño de un cubo



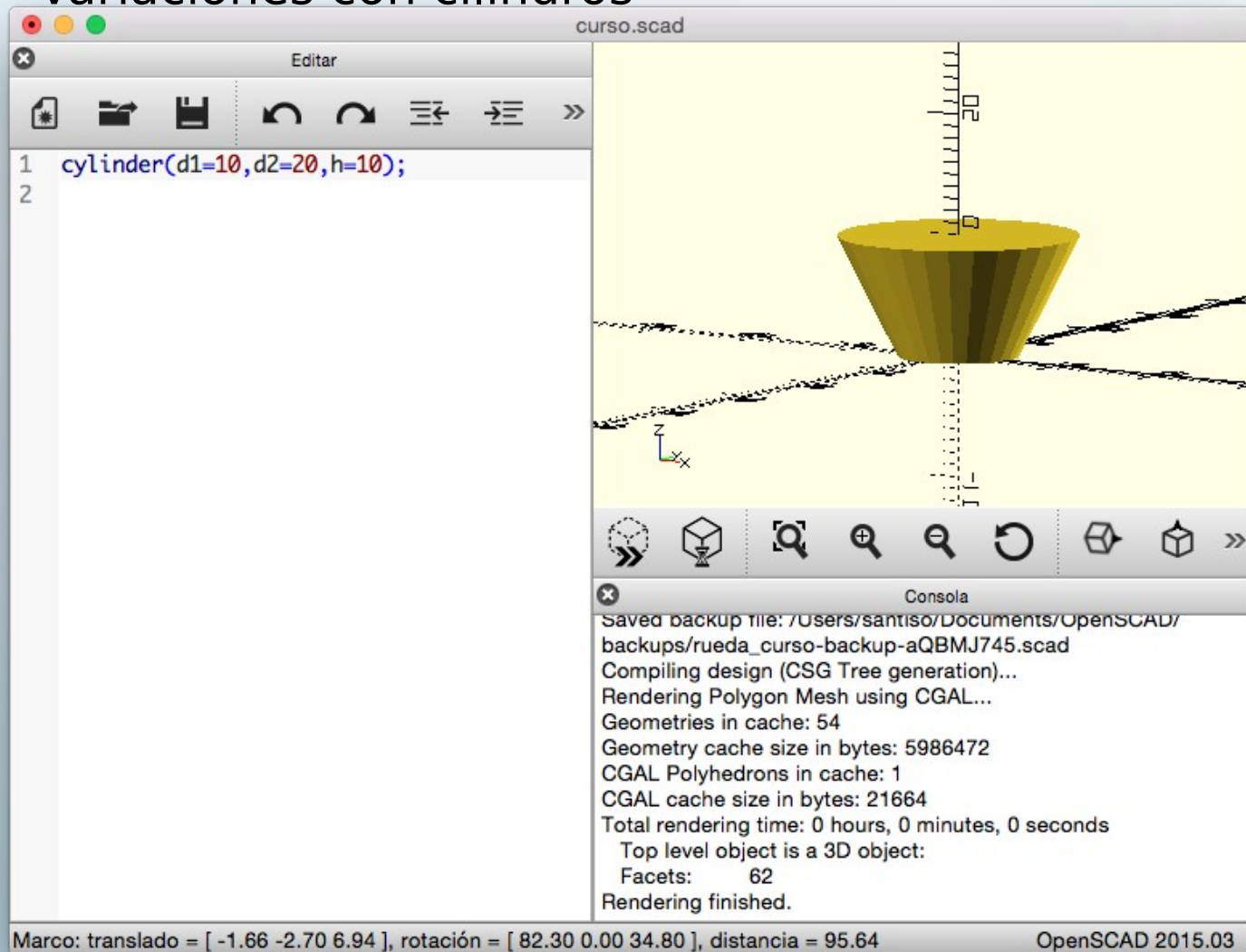
Programación

- Diseño de un cilindro



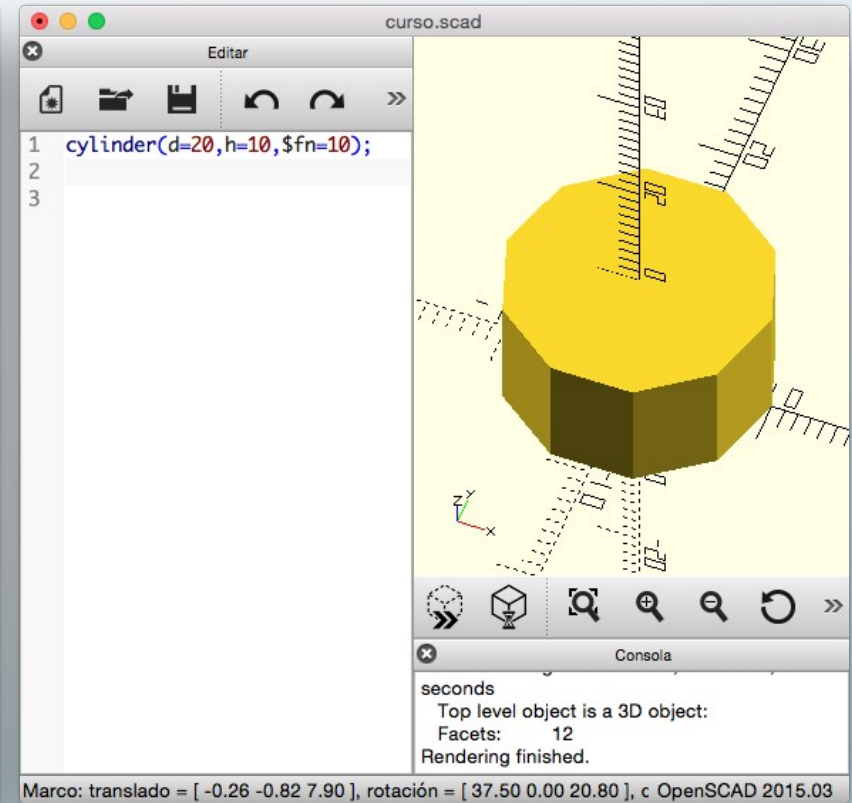
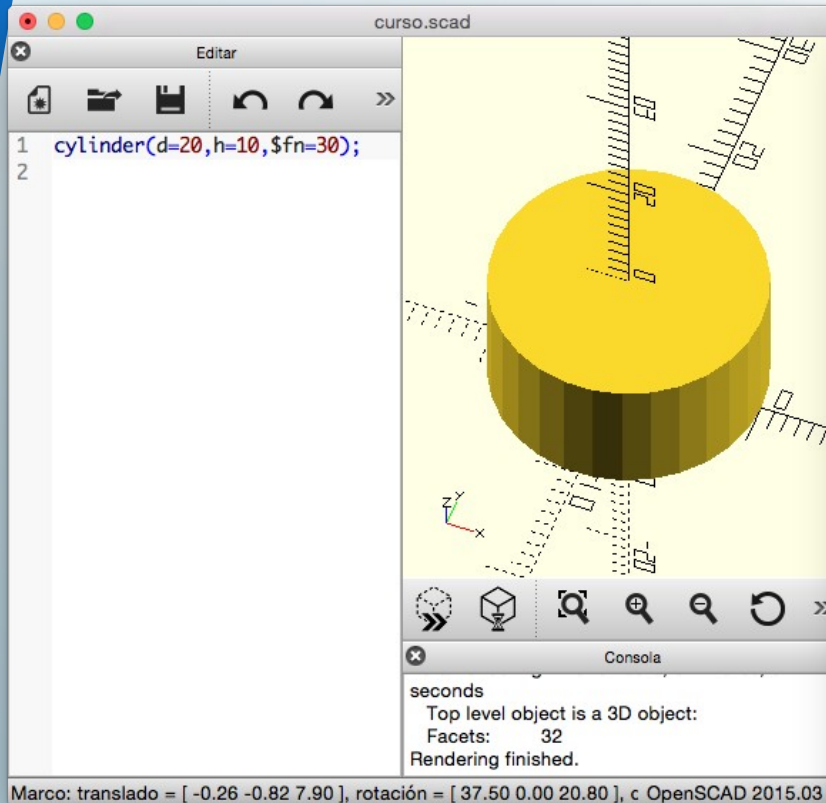
Programación

- Variaciones con cilindros



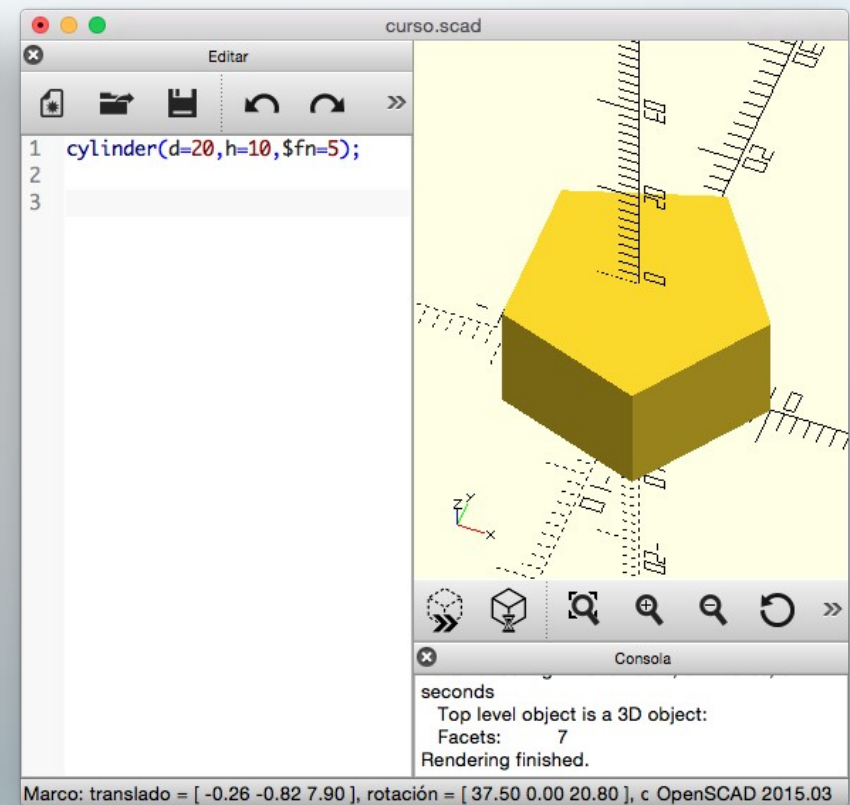
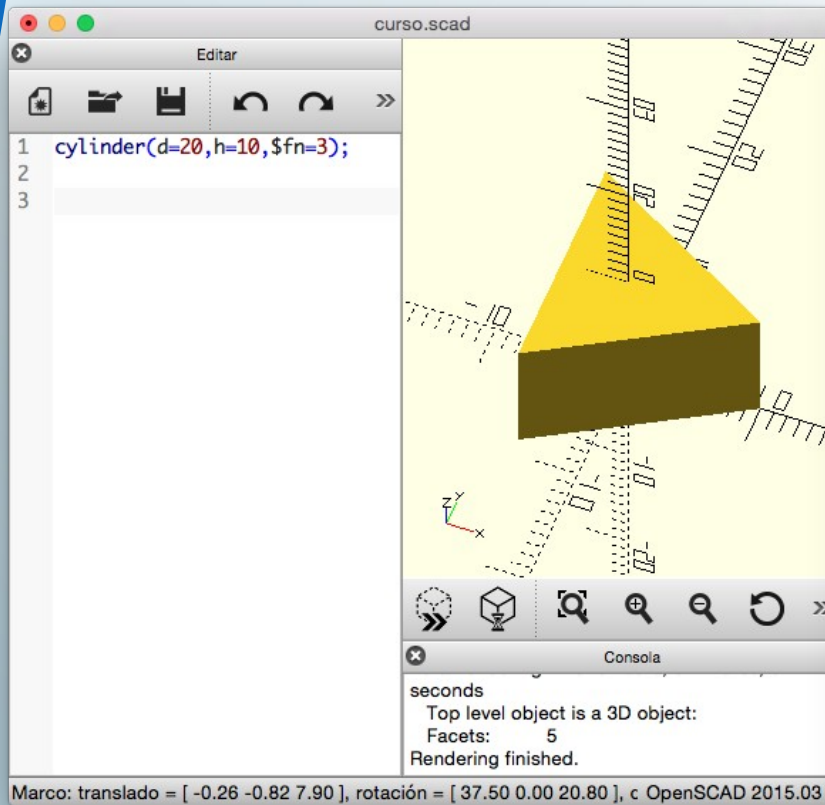
Programación

- Variaciones con cilindros. Resolución \$fn.



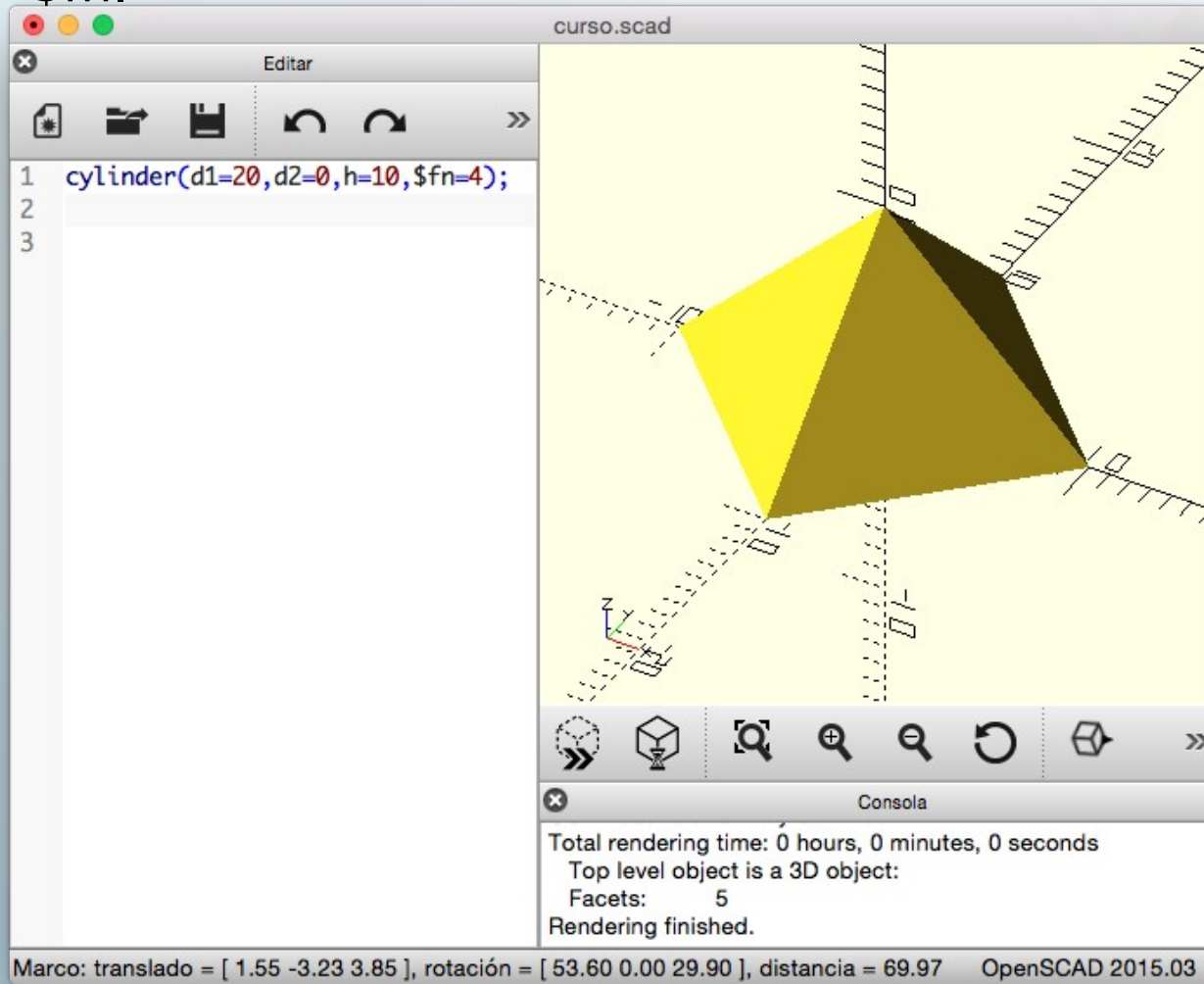
Programación

- Variaciones con cilindros. Resolución \$fn.



Programación

- Variaciones con cilindros. Resolución \$fn.



Programación

- Objetos base

2D

`circle(radius | d=diameter)`

`square(size,center)`

`square([width,height],center)`

`polygon([points])`

`polygon([points],[paths])`

3D

`sphere(radius | d=diameter)`

`cube(size)`

`cube([width,depth,height])`

`cylinder(h,r|d,center)`

`cylinder(h,r1|d1,r2|d2,center)`

`polyhedron(points, triangles, convexity)`



Programación

- Sintaxis

Syntax

```
var = value;  
module name(...) { ... }  
name();  
function name(...) = ...  
name();  
include <...scad>  
use <...scad>
```



Programación

- Transformaciones

Transformations

```
translate([x,y,z])  
rotate([x,y,z])  
scale([x,y,z])  
resize([x,y,z],auto)  
mirror([x,y,z])  
multmatrix(m)  
color("colorname")  
color([r, g, b, a])  
hull()  
minkowski()
```



Programación

- Operadores booleanos

Boolean operations

`union()`

`difference()`

`intersection()`

- Variables especiales

Special variables

`$fa` minimum angle

`$fs` minimum size

`$fn` number of fragments

`$t` animation step



Programación

- Otros

Other

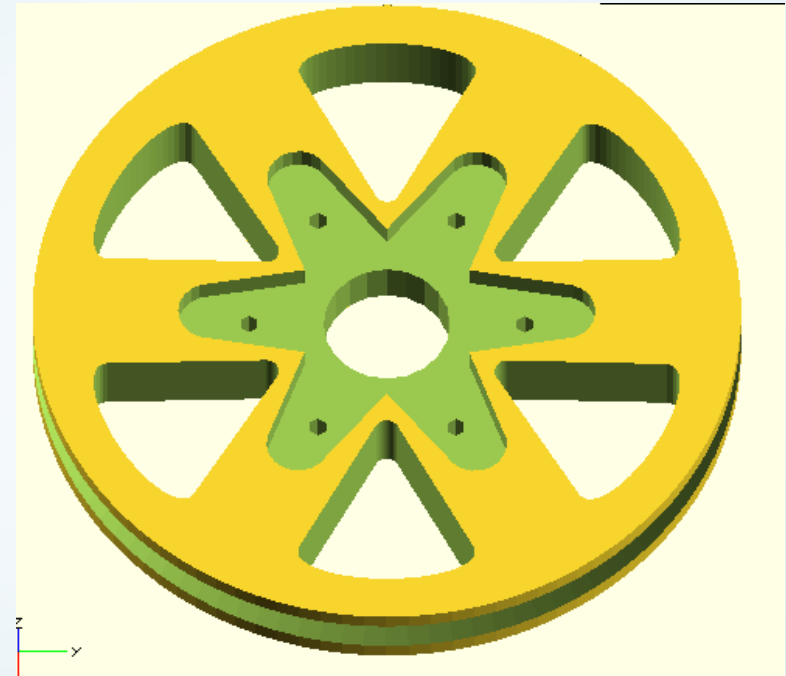
```
echo(...)  
for (i = [start:end]) { ... }  
for (i = [start:step:end]) { ... }  
for (i = [...,...,...]) { ... }  
intersection_for(i = [start:end]) { ... }  
intersection_for(i = [start:step:end]) { ... }  
intersection_for(i = [...,...,...]) { ... }  
if (...) { ... }  
assign (...) { ... }  
import("...stl")  
linear_extrude(height,center,convexity,twist,slices)  
rotate_extrude(convexity)  
surface(file = "...dat",center,convexity)  
projection(cut)  
render(convexity)  
children([idx])
```

Mathematical

```
abs  
sign  
sin  
cos  
tan  
acos  
asin  
atan  
atan2  
floor  
round  
ceil  
ln  
len  
log  
pow  
sqrt  
exp  
rands  
min  
max
```

Programación

- Ejercicios
 - Rueda del robot



Ejemplos

- Rueda del robot
 - Maciza
 - Surco triangular
 - Surco circular
 - Adaptador servo
 - Radios
 - Parametrizar

