

---

# OpenGL

Khang Lam  
Daniel Limas  
Kevin Castillo  
Juan Battini

---

**3035 - Professor Mark Sargent**

# What is OpenGL?

Open Graphics Library (**OpenGL**) is a cross-language, cross-platform application programming interface (API) for rendering 2D and 3D vector graphics.

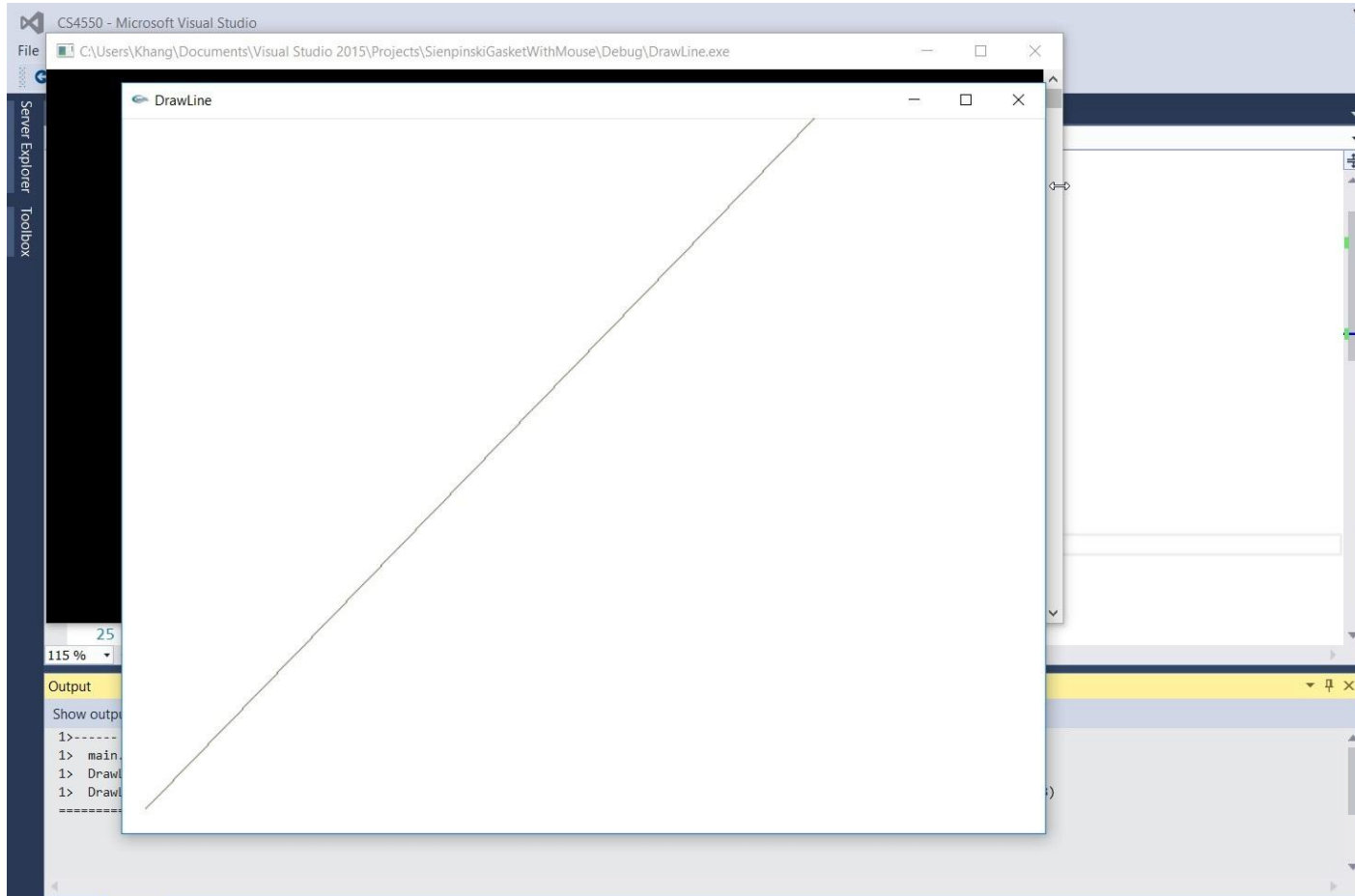
## How is OpenGL used ?

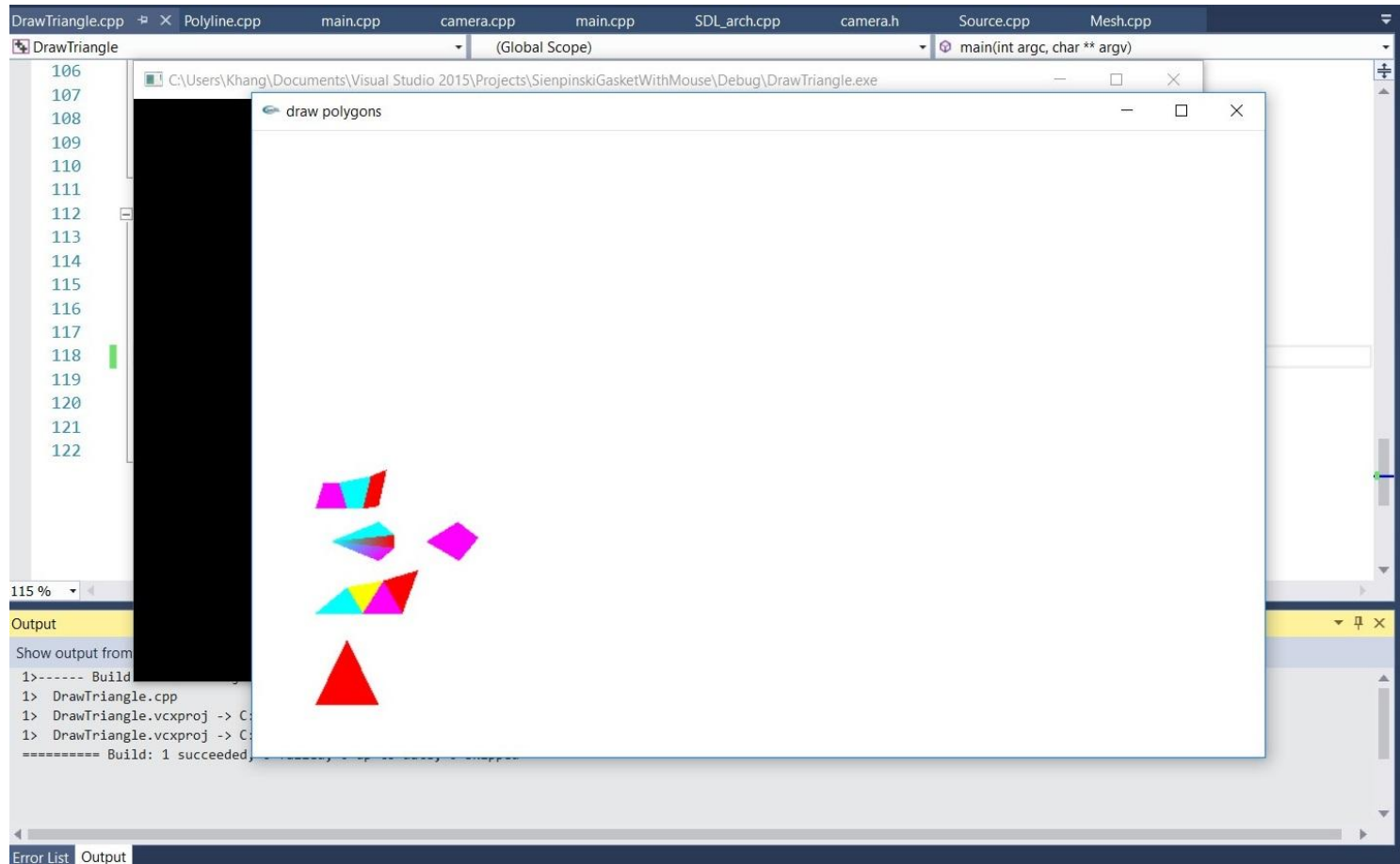
OpenGL is used extensively in the fields of computer-aided design (CAD), virtual reality, scientific visualization, information visualization, flight simulation, and video games.

## High Visual Quality and Performance

- OpenGL capabilities can be exploited by any visual computing application requiring maximum performance.
  - Example: 3D animation, CAD, visual simulation.
- Extremely used in markets such as broadcasting, CAD/CAM/CAE, entertainment, medical imaging, and virtual reality.

# Examples of some simple programs using OpenGL





# Drawing a Dinosaur

- Can use regular libraries and functions that come with the parent language.
- Read from file to draw over 300 points to make the drawing of the dinosaur.

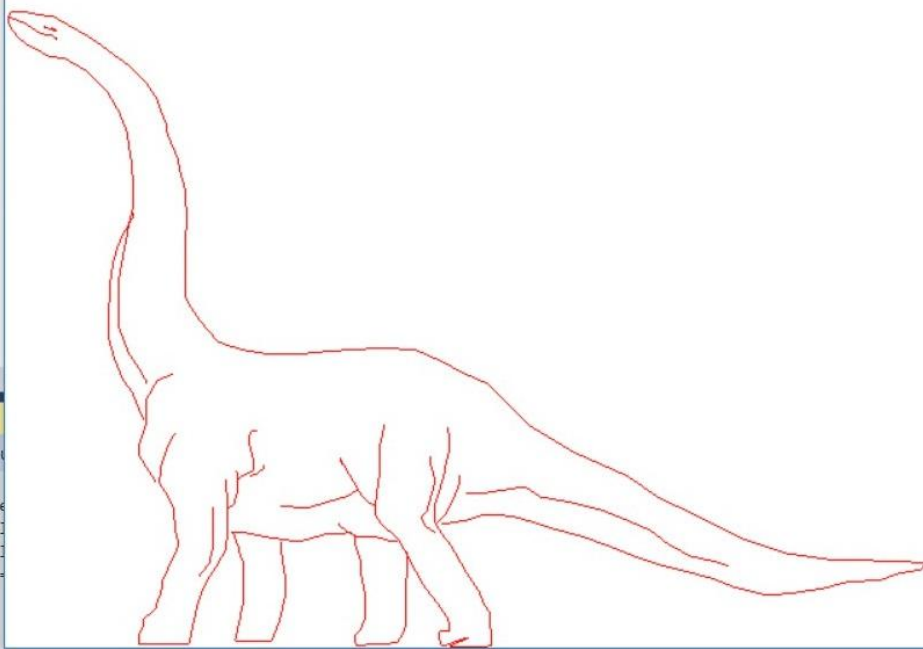


Polyline.cpp | main.cpp | camera.cpp | main.cpp | SDL\_arch.cpp | camera.h | Source.cpp | Mesh.cpp | TeapotRotate.cpp

Polyline (Global Scope) | main(int argc, char\*\* argv)

40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

dat dino



115 %

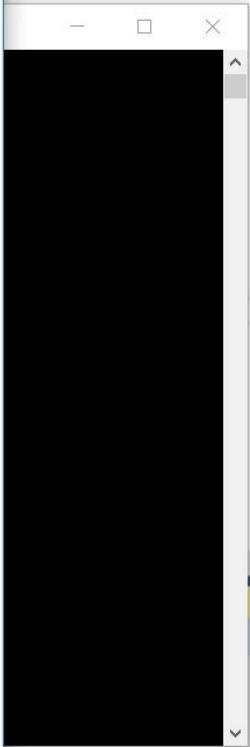
Output

Show output

1>-----  
1> Singe  
1> Sing  
1> Sing  
-----

Error List

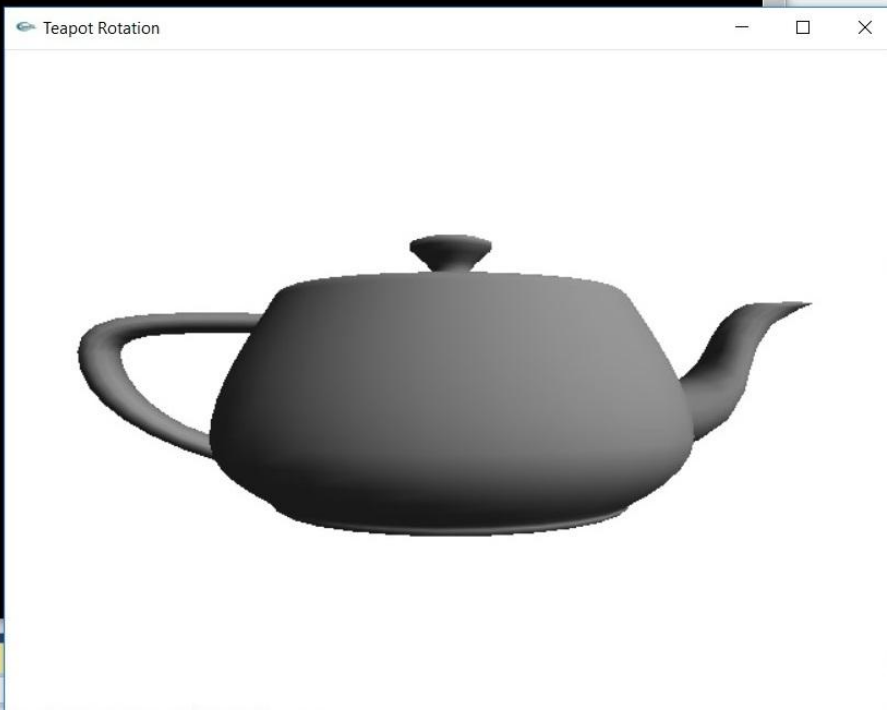
Output



# Making a TeaPot

- One possible difficulty of this application is that it requires previous knowledge of linear algebra like matrices.
- Rotating a TeaPot requires to constantly redraw a teapot at different points on the screen to give the illusion of rotation.

C:\Users\Khang\Documents\Visual Studio 2015\Projects\SienpinskiGasketWithMouse\Debug\Project\_TeapotRotate.exe



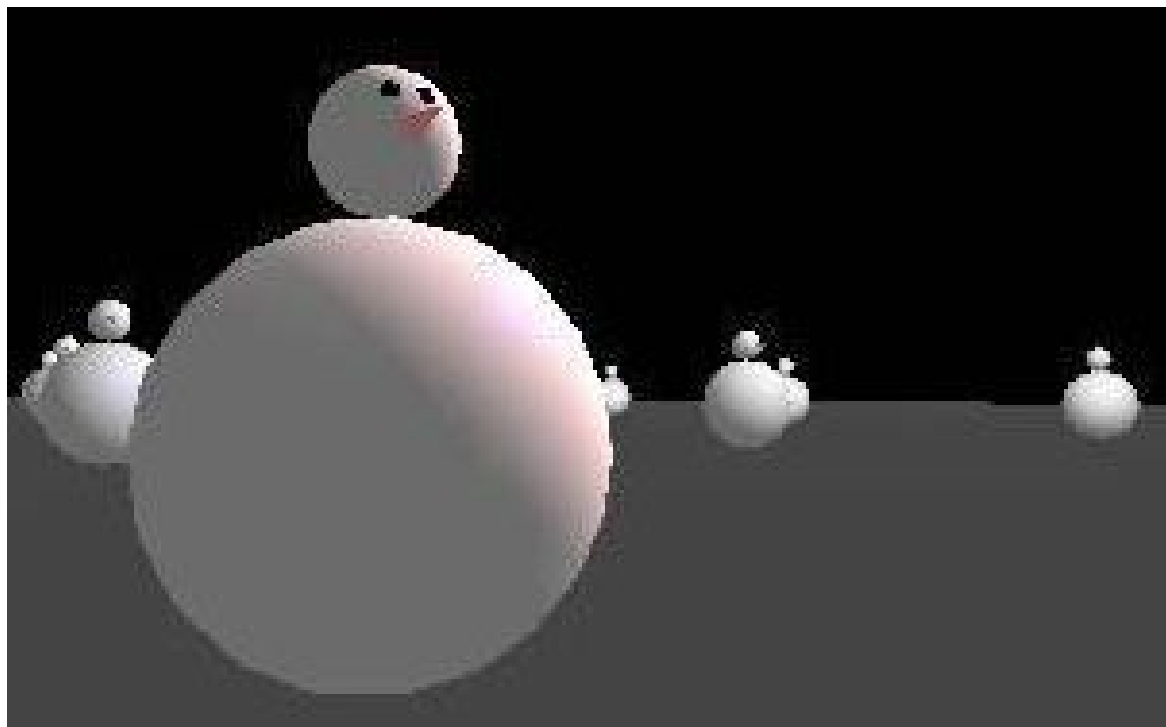
## Output

Show output from: Build

```
1>----- Build started: Project: Project_TeapotRotate, Configuration: Debug Win32 -----
1> TeapotRotate.cpp
1> Project_TeapotRotate.vcxproj -> C:\Users\Khang\Documents\Visual Studio 2015\Projects\SienpinskiGasketWithMouse\Debug\Project_TeapotRotate.exe
1> Project_TeapotRotate.vcxproj -> C:\Users\Khang\Documents\Visual Studio 2015\Projects\SienpinskiGasketWithMouse\Debug\Project_TeapotRotate.pdb (Partial PDB)
***** Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped *****
```

# Adding camera Movement

- OpenGL has many useful functions that can meet anyones needs.
- Has a camera class to allow the user to move around in his masterpiece.



# Advantages of using OpenGL

Why is OpenGL widely used ?

# Industry standard

- OpenGL is the only truly open, vendor-neutral, multi-platform graphics standard.
- Broad industry support: an independent consortium, the OpenGL Architecture Review Board, guides the OpenGL specification.

# Stable

- Available for more than seven years on a wide variety of platforms.
- Specifications are well controlled
- Proposed updates are announced in time for developers to adopt changes.
- Backward compatibility requirements ensure that existing applications do not become obsolete.



## Reliable and portable

Produce consistent visual display results on any OpenGL API-compliant hardware, regardless of operating system or windowing system

# Evolving

- OpenGL allows new hardware innovations to be accessible through the API via the OpenGL extension mechanism.
- Allows application developers and hardware vendors to incorporate new features into their normal product release cycles.

# Scalable

- OpenGL can run on various kinds of systems: mobile devices, PCs, workstations, and supercomputers.
- Applications can scale to any class of machine that the developer chooses to target.

# Easy to use

- OpenGL is well structured with intuitive design and logical commands.
- Applications can be written in fewer lines of code compared to other graphic libraries.
- OpenGL drivers encapsulate information about the underlying hardware, freeing the application developer from having to design for specific hardware features.

## Well-documented

- Numerous books about OpenGL have been published
- Sample code is readily available

**How**

**OpenGL Simplifies Software Development, Speeds  
Time-to-Market ?**

# 1.

- OpenGL routines simplify the development of graphics software—from rendering a simple geometric point, line, or filled polygon to the creation of the most complex lighted and texture-mapped NURBS curved surface.

## 2.

- OpenGL gives software developers access to geometric and image primitives, display lists, modeling transformations, lighting and texturing, anti-aliasing, blending, and many other features.



### 3.

- Every conforming OpenGL implementation includes the full complement of OpenGL functions.
- Language bindings: C, C++, Fortran, Ada, Java.
- OpenGL implementations come from a single specification and language binding document
- Maximize programmer productivity and shorter time-to-market.

## 4.

- All elements of the OpenGL state can be obtained by an OpenGL application.
- OpenGL supports visualization applications with 2D images treated as types of primitives that can be manipulated just like 3D geometric objects.

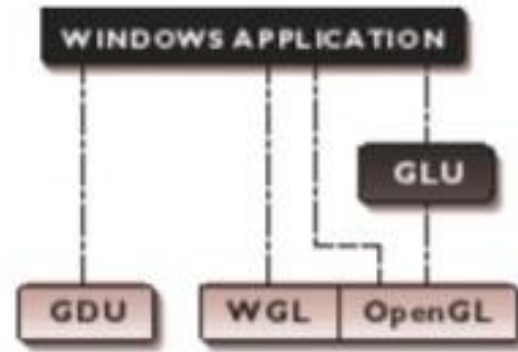
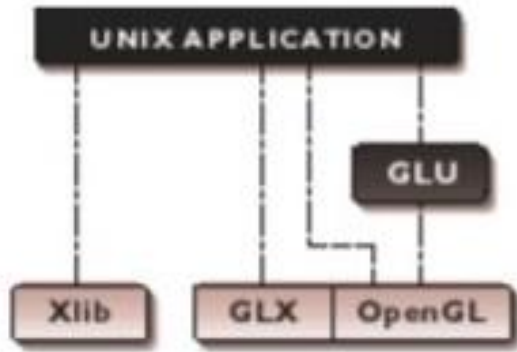
# OpenGL Is Available Everywhere

## OpenGL:

- Supports UNIX® workstations, Windows 95/98/2000/NT and MacOS PC
- Runs on operating systems: Mac OS, OS/2, UNIX, Windows 95/98, Windows 2000, Windows NT, Linux, OPENStep, and BeOS
- Works with windowing system: Win32, MacOS, Presentation Manager, and X-Window System.
- Callable from Ada, C, C++, Fortran, Python, Perl and Java and offers complete independence from network protocols and topologies.

# API Hierarchy

- OpenGL applications use the window system's window, input, and event mechanism
- GLU supports quadrics, NURBS, complex polygons, matrix utilities, and more



# Flexibility and Differentiation

- OpenGL hardware acceleration is widely available on everything from low-cost PCs to high-end workstations and supercomputers.
- Application developers are assured consistent display results regardless of the platform implementation.
- Hardware developers can differentiate their products by developing extensions that allow software developers to access additional performance and technological innovations.
- Many OpenGL extensions, as well as extensions to related APIs like GLU, GLX, and WGL, have been defined by vendors and groups of vendors.

# Continued Innovation

- Formal revisions occur at periodic intervals, and extensions.
- Allowing application developers to access the latest hardware advances through OpenGL are continuously being developed.
- OpenGL extensions become widely accepted allowing OpenGL to evolve in a controlled yet innovative manner.

**Thank you for listening!**