Reprinted from the February 1999 issue of *Computer Graphics*

A Syllabus Survey: Examining the State of Current Practice in Introductory Computer Graphics Courses

Rosalee Wolfe
DePaul University

This column is the second in a series revisiting the introductory computer graphics course for undergraduate computer science majors. It has been ten years since the last formal discussions resulted in a list of topics for *Curriculum 91*[1]. Given the great changes that have occurred in computer graphics during that time, the SIGGRAPH Education Committee is re-examining this issue.

At SIGGRAPH 98, several computer graphics educators met to compare syllabi and as a result of the discussion that ensued, decided to solicit syllabi from educators at a variety of institutions across the country. Scott Grissom, Lew Hitchner, Bill Jones, Susan Reiser and I collected syllabi from 23 educators. For a list of the instructors who contributed syllabi, please see the Education column in the last issue of *Computer Graphics*[2]. Of the 23 collected, two were strictly for graduate students and one was primarily an image-processing course. This column examines the remaining 20 syllabi.

Level

As can be seen in table 1, instructors universally believe that an introductory computer graphics course requires a certain degree of maturity. No courses were targeted for students below the junior year and several were crosslisted as graduate-level courses.

Junior	4
Upper Level	10
Senior	2
Upper Level/Grad	3
Senior/Grad	1

Table 1: Level of Course

Previous Experience

One of the factors that places an introductory computer graphics course at an upper level is the extensive amount of background required for the course. Table 2 lists the immediate prerequisites for the surveyed courses. Most courses had more than one prerequisite. All of them required at least a year of programming courses or "programming fluency". In addition, almost every course required some background in mathematics, but the range in requirements is very wide, ranging from analytic geometry to multivariate calculus (Calculus III). However, the most prevalent math prerequisite was Linear Algebra.

Algorithms and Data Structures	12
CS2	4
Programming fluency	2
Software Engineering	2
Linear Algebra	11
Calculus I	2
Calculus II	1
Calculus III	1
Discrete Math	1
Analytic Geometry	1

Table 2: Immediate Prerequisites

Textbook, Software

A few textbooks featured prominently, namely Angel's *Interactive Computer Graphics: A top-down approach with OpenGL,* Hearn and Baker's *Computer Graphics,* and the classic Foley, van Dam, Feiner and Hughes' *Computer Graphics: Principles and Practice.* Two instructors did not use a text of any kind but created their own notes for classroom use.

Angel	8
Hearn and Baker	8
Foley, van Dam, Feiner, Hughes	2
notes	2

Table 3: Textbooks used

There was also quite a bit of consensus on software selection, as Table 4 suggests. Quite a few instructors mentioned that the choice of computer and operating system was not important as long as students had access to OpenGL. Several instructors mentioned that they used OpenInventor or GLUT in addition to OpenGL. Two instructors failed to answer this question, which is why the total number of responses is less than 20.

OpenGL	12
C/C++	2
custom	2
VRML/Renderman/Java	1
Pascal	1

Table 4: Software used

Topics

The topics listed in Table 5 appear in the contributed syllabi. The range and number (38) of topics attest to the rich variety of backgrounds and interests of computer graphics instructors and include such diverse subjects as time-critical applications, animations, fractals, virtual reality and scientific visualization. In general, instructors tend to present these more exotic topics near the end of the course.

Although there are a large number of topics, five are mentioned in 70% or more of the courses, and three of these draw heavily on mathematics. The most prevalent, mentioned in 95% of the syllabi, is viewing transformations. (The thrust of the one course not mentioning this topic is building a ray tracer, which can obviate the need for a matrix representation of viewing transformations.) Another topic, 3D transformations, relies on the same matrix operations as does the viewing operations. Three quarters of the courses cover lighting or illumination models, which also requires a substantial math background.

Two additional topics are present in 70% or more of the syllabi. Most courses have an opening discussion covering graphics hardware and basic terminology, and they also include some discussion of the principles and implementation of interactive techniques.

The topic of object representation occurs in over half of the courses and includes a range of items, from representing points, line segments and polygons to representing surfaces of revolution. Although instructors mention curves and smooth surfaces fairly often, many add the proviso "as time permits." Rasterization topics range from Bresenham's line- and circle-drawing algorithms to polygon fills. Another topic occurring fairly frequently is data structures that support graphics. This encompasses everything from display lists and polygon meshes to scene graphs, scene hierarchies and octrees.

viewing transformations/ camera	19	math foundations/review	5
hardware/terminology	15	2D transformations	4
lighting models	15	3D rendering	4
3D transformations	14	fractals	4
user interaction	14	coordinate systems	3
object representation	11	photorealistic methods	3
shading models	11	antialiasing	2
color/color models	10	hidden-line removal	2
curves	10	pipeline details	2
hidden-surface removal	10	virtual reality	2
rasterization	9	VRML	2
implementation particulars	7	2D ray tracing	1
supporting data structures/models	7	computational geometry	1
texture mapping	7	global illumination	1
clipping	6	interactive internet applications	1
ray tracing	6	projections	1
surfaces	6	scientific visualization	1
2D drawing	5	shadows	1
animation	5	time-critical app	1

Table 5: Course Topics

What's next?

How does this list of topics compare to the course you teach? Have we omitted anything important? Are there topics listed here that you feel are no longer relevant? Let us know what you find important. Email from you is happily read at wolfe@cs.depaul.edu.

As a next step, Lew Hitchner, Steven Cunningham, Scott Grissom and I will present a panel at SIGCSE 99 in New Orleans. This is the annual conference for the ACM Special Interest Group on Computer Science Education. Check the website at www.cs.virginia.edu/SIGCSE for details. Entitled *Computer Graphics: The Introductory Course Grows Up,* the panel incorporates the feedback received thus far and will describe a philosophical basis for changing the introductory undergraduate course. It will also give some examples of courses that are responding to that change. In addition, we are sponsoring a Birds-Of-a-Feather session to gather feedback on drafts of sample syllabi. We hope to see you there.

References

[1] ACM/IEEE-CS Joint Curriculum Task Force. *Computing Curricula 1991*. February, 1991.

[2] OpenGL: Agent of Change or Sign of the Times? *Computer Graphics,* November 1998, 29-31.