



Computer Science 112

Art & Science of Computer Graphics

The College of Saint Rose
Fall 2015

Topic Notes: Introduction

What is Computer Graphics?

Computer graphics is, in short, the field that is concerned with visual output from a computer. It is a relatively young but very rich field. Active research is ongoing, but many of the fundamentals have remained the same for decades.

Computer graphics is fundamental to our interactions with many modern computers. Computer graphics is at the core of the *graphical user interfaces (GUIs)* we all use on our desktop, laptop, and mobile computers.

We can see a (very) brief look at the history of computer graphics at <http://www.fiegggen.com/graphics/history.htm> (linked from today's lecture page).

Of course, our focus is on graphics on modern computers. We will talk more later about issues like display resolution and image sizes and formats.

Modern computer graphics is applied in many areas, beyond the standard user interfaces we work with on a regular basis.

- Gaming
- Photorealistic images and video (including those for television, movies, *etc.*)
- Scientific visualization

So what can we do in a computer graphics course with no programming prerequisites?

We will focus on creating images (and later, animations, hopefully) of objects in a scene. At first, these will be simple geometric shapes with simple colors and material properties. Later, we will work toward making our images more photorealistic. I will show you how to use the tools and you need to bring your artistic abilities.

Along the way, you will learn how to write *computer programs* that describe the scenes to the computer and specify how to create images of those scenes. You will learn some of the science behind the techniques that are used to create, store, and display these images from our scene descriptions. And the *computer programming* techniques you learn will be useful in many other contexts, as well. Programming is the act of creating a computer program, which is, in general, a set of instructions that can be understood by a computer.

But more importantly, computer programming and the way of thinking needed to do it successfully are a fundamental tool for problem solving in our world. Whatever meaningful work you plan to do in nearly any field, that work can probably be seen as a series of problems to be solved, and some ability to program a computer will be an essential tool for you to be able to solve those problems.

Why Study Programming?

“I’m not a Computer Science major, why should I have to learn anything about about programming?”

- Programming skills may be applied in other areas.
 - The experience of programming helps develop problem solving skills, in particular, the ability to deal with complexity.
 - Computer technology is pervasive and computational thinking is pervading society. For example, biologists are beginning to model cells as distributed computing devices that communicate over membrane boundaries. Nearly every field of science now has a “computational” branch of that science which is becoming as important as the experimental and theoretical branches. Moreover, the impact of the computer on society is huge and will continue to accelerate over the years.
 - Most programming is done by non-computer scientists. Customizing of tools is possible by those who understand programming. This can range from programming complicated queries on databases to running complex simulations with spreadsheets. The techniques you learn in any of these contexts can usually be applied to others.
 - It’s a fascinating challenge to teach the computer how to solve hard problems.
-

The Ambrosia Modeling System

We will spend most of our time this semester learning one particular tool that can be used to generate images and animations: *The Ambrosia Modeling System*.

Ambrosia will allow us to describe the objects we want in the scene, the properties of those objects, the lighting, and the position of the camera. From this description, Ambrosia will generate either a single, static image or a series of images that it will glue together into an animation.

Ambrosia is itself a Python-based front end to the popular Persistence of Vision rendering engine. Ambrosia is being developed (present tense) by Duane Bailey at Williams College, specifically for use in a course there with the same name as ours. He has graciously made his software available for us to work with here. Python is a general purpose programming language that has gained extensive popularity in recent years.

As the semester goes on, we will learn to use more and more features of Ambrosia and the Python programming language in general that will allow us to build more complex scenes that will be

the subjects of our images and animations. And, much of what you learn about Python will be applicable to many domains outside of computer graphics.