ARCH 481: 3D MODELING AND RENDERING

http://courses.washington.edu/arch481

When:	Tu/Th, 3:30 to 4:50	**Where**:	Digital Commons (GLD 007F)
Credits:	3, Graded	**Prerequisite**:	Arch 380 (or permission)

Instructor:	Brian Johnson	**Assistant**:	Elham Soltani
E-mail to:	brj@uw.edu	**E-mail to**:	elhams@uw.edu
Office:	Arc G55	**Office**:	DMG
Office Hours:	by appointment	**Consulting Hours**:	tbd
Office phone:	206.543.2132		

This is a course about architectural *modeling* and *rendering* with computer graphics tools. There are two important things to keep in mind as you start this study, and they are encapsulated by these two quotes:

> *What remains hard is modeling. The structure inherent in three-dimensional models is difficult for people to grasp and difficult too for user interfaces to reveal and manipulate. Only the determined model three-dimensional objects, and they rarely invent a shape at the computer, but only record a shape so that analysis or manufacturing can proceed. The grand challenges to three-dimensional graphics are to make simple modeling easy and make complex modeling accessible to far more people.*
>
> *-Robert Sproull,*
> *keynote speech, SIGGRAPH 1990*

Robert Sproull is one of the founding fathers of computer graphics. The "Special Interest Group on Graphics" (SIGGRAPH) annual conference is the biggest computer graphics conference in the world. This guy knew what he was talking about in 1990, and it's still true today.

> *A journey of a thousand miles must begin with a single step.*
> *-Lao-tzu*

Lao-tzu was a Chinese philosopher. China is a big country. He knows what he's talking about too.

Goals of the Course

This course has two main goals:

- To develop a "critical eye" with regard to computer graphic (CG) images by developing your understanding of and experience with 3D modeling & rendering. To this end, we will examine at a conceptual level a number of different rendering algorithms, and see first-hand the kinds of images they produce. Understanding how they work will help you judge what are reasonable and unreasonable expectations of CG technology in your own work.
- To remind you that images are communicative tools, linked together and used to "tell a story" to someone else. Models are not objective reality from which rendering programs simply produce precise snapshots. They are *authored*–created and manipulated to accomplish a particular communication task. There are overlaps between digital techniques and rendering using traditional media–different visual results, but similar authorship goals, responsibility and opportunity. You should THINK about what story you are telling with each image.

Course Philosophy

This course is about *beginnings*, in preparation for regular use of 3D computing during your professional career. It is **not** a course about a particular software package, though you'll sometimes swear that it is. You may not use the same programs in a few months or years as you use this quarter, not because they aren't good, but because change happens. From this point of view it is not so important what programs or commands you learn as it is *what you learn from using* those programs and commands. On the other hand, it's very hard to master this subject matter purely as "book learning". You need to *do* modeling and rendering as well as learn the concepts. The combination of exercises

and lectures is intended to provide an enriched environment on which your modeling and rendering skills can develop and expand.

There is a lot to learn (and it keeps increasing!). We use a fast-paced, "learning by doing" project-based approach. Each week lectures and demonstrations during the class sessions will help you to undertake a related modeling and/or rendering task on your own. Online readings will explain many of the technical issues surrounding the exercises.

The weekly exercise is the central component of the course. The lectures support and explain the exercises by providing material that may not be available in the readings and by providing an opportunity to ask questions. However, completing the exercise provides most of the actual learning experience. The exercises take time (the course evaluations say the median is 10.6 hours/week) to complete and write up. That time must be spent on campus unless you have the software installed on your own computer (see below). In addition, most help is available during the day and early evening. Please consider these things as you set up your schedule for the quarter.

U-pick software?

While this isn't a course about use of just *one* piece of software, you will need to *use* at least one to complete the course. It would be nice if it were powerful and easy to use, was useful in other courses and studios, and got you a job later, but given the variety of programs out there and the speed of change, that's not the most important issue. Access is. In that regard, you may prefer to use a Mac or a PC, but the computers in 007 are Windows boxes, and not all software runs on both. Further, the public studio and lab computers in the college, and the laptops in the CBE Library, have software installed, so we are predisposed to use it.

For in-class demos we will be using the **Rhino and V-Ray** combination (plus Adobe **Premiere** for post-production). However, if you already have or want to use other software and are prepared to "go it on your own" as far as direct help is concerned…. talk to me and we'll sort something out. In the past students have used Blender, 3DS Max, and form-Z, with varying results.

Selecting Your Project

You will need to select a project to work on for the quarter. Old studio projects often work well, but if you don't have a project to work from, I have drawings for a couple of different buildings that you can use. You will find the drawings on the course website. If you pick a project from a previous studio, I recommend that you don't resurrect an old digital model, and do ***NOT*** take on an historical building (Barcelona Pavilion, etc). It is important that you have reasonably detailed drawings for the building, so that this doesn't turn into a guessing exercise. It is also recommended that the building NOT be large or highly complex, allowing you to focus on the process rather than raw productivity and minimizing limitations due to hardware and time (some renderings may take hours, even with modest models).

Weekly Exercises

Each week we will cover a different cluster of topics, building from the simple to the more complex. In-class lectures and demonstrations, combined with readings, provide the conceptual information you need, plus tips and tricks related to the particular program or problem. Weekly exercises provide an opportunity to test comprehension as well as develop and demonstrate skill. They also provide a means of incrementally developing your model.

The lecture topics/exercises are shown in the schedule and detailed in the exercise workbook. For most exercises there is an associated lecture and hands-on class session, but a few "extra-credit" exercises are also offered for independent challenge. **NOTE: due to variation in software, some of the more esoteric exercises cannot be completed with all software combinations.** I'll try to alert you to particular issues with software as we progress, and we can always discuss substitute projects if necessary.

Exercise Workbook

The weekly exercises are described in an exercise workbook (pdf) on the course website. It is your responsibility to read the appropriate problem statement and review the printed and online textual material prior to the lecture. Most exercises include a list of recommended readings available on the course website.

Turning in Work

For each exercise you will generally be expected to turn in two things: a *set* of four or more image *files* showing your visual results, and printed *hardcopy* of your answers to the exercise questions from the workbook. Files will be turned in using the "Collect It" Catalyst tool. The URL is

https://catalysttools.washington.edu/collectit/dropbox/summary/brj/36229

(which is kind of long, so there's a link from the course website too). Dues dates are shown on the schedule.

Due Dates & Late Work

This class requires regular work, and it gets much harder to do the exercises if you aren't keeping up, and I want you to get feedback with time to use it. So, I will **not accept** a pile of catch up work at the end of the quarter. On the due-date shown in the schedule each exercise may be turned in for full credit. After that your grade declines by 0.1 for each *week or fraction of a week* that it is late. Allowances will be made for illness and similar problems (*if* you stay in touch).

Get Out of Jail Free cards

"Stuff" happens to everyone now and then, so everybody gets a "get out of jail free card". This entitles you to a single one-week penalty-free extension to use as you wish or need (but you must tell me you're using it if & when you do), and they don't extend the quarter. That final due date is the end.

Grading of individual exercises

Exercise grades are based on *COMPLETION* (production and quality of results), and *CONCEPTION* (demonstrated understanding of the principles behind the software). COMPLETION includes consideration of the "story telling" *content* of your renderings as well as the *technical control* you demonstrate in the work. CONCEPTION is addressed through exercise questions that address specific issues. If there are substantial deficiencies in either area you may be asked to resubmit the project. **Your exercise will generally be given a score between 0.3 and 0.6, with 0.4 to 0.5 being average.** Extremely weak work may be returned ungraded.

Extra Credit

Some exercises include one or more clearly labeled "extra credit" activities designed to stretch your understanding and/or challenge your mastery of the software. Adequate completion of the extra credit activity will add *up to* 0.1 to your grade on the exercise. Proposing and completing additional independent exercises is another way to earn points.

Course Grading

Your grade for the course is simply the **arithmetic sum** of the grades you receive on the individual exercises you turn in. There is no *penalty* if you do not do all the exercises (other than the lost opportunity to earn points). There are enough regular exercises so that if you get "OK" grades (in the 0.3 to 0.5 range) on each exercise it should be fairly straightforward for you to earn a high grade in the course.

Textbook

In the interest of keeping your costs down, there is *no required book* for this course (but see next section).

The *Tapestry* Web Pages

Past student evaluations have suggested that greater emphasis be placed on in-class demonstration and hands-on work. Because the pace of such demonstrations is necessarily much slower than that of lectures, responding to this suggestion means additional out-of-class time for you. To help, I have converted much of the "lecture" content into web pages, which you can consult instead of listening to two lectures each week. This material should be considered required reading. Readings are linked to the course web page (see URL on page 1). Read them. These are intended to help you complete the exercises and understand what the software is doing. If they aren't helping, please let me know.

Backups & Other Costs

You will likely want *at least one* USB "jump drive" or similar to store your work on. Protect your work and your sanity by making back-ups of your project in at least two places. See the workbook for more on making backups.

Getting Help

The most important thing to remember about working with any computing software, especially unfamiliar software, is ***DON'T charrette***. If you start your project early there will be plenty of time to get help. If you wait until the last minute you are almost certain to get hung-up for hours by something that turns out to be trivial. Obviously, working around those blockages is one of the secondary skills you need to acquire, but *adequate time* is one of the key ingredients to minimal frustration.

There are a variety of places you may find help:

- **Yourself**. Take a deep breath and get calm. If you're looking at a dialog box or prompt that you don't understand, the computer isn't doing any damage to your file. If you ***do*** something, you might. You can often figure out error messages by just taking a minute to *stop and think* about what you just did and why it might have "gone wrong".
- **One-another**. Learning from and helping each other is an important skill. Use the course email list, as well as face-to-face opportunities in studio and labs, to expand *and share* what you know. When you do, make sure you understand the answer (don't let others simply do things to your data for you).
- The exercise **statement** in the workbook may contains hints and suggestions, so *read* it carefully. If it contains errors, please let me know.
- Program **user-guides**, online **tutorials**, and built-in **help systems** can actually be helpful. Check under the program's "**HELP**" menu, visit related online forums, or try an online search engine for guidance.
- **Me**. I don't have lots of posted *office hours* because I try to be available to students *at all times*. The best way to contact me is via e-mail, but knocking on my door works too! Be prepared to describe or demonstrate the problem, or error message if there was one (this is another useful skill).

About Shared Work

You may certainly work together, meaning that you work *in the same place* and at *the same time*, but not *on the same data*. Feel free to share ideas, tricks, knowledge, etc. You are graded individually, so there's no competition—just be sure you understand what you are doing, and always use your own words in the write-ups (or cite the source of the information). Plagiarism (copying the work of another and claiming it as your own) will *cost* you points and may be grounds for academic disciplinary action. This includes copy and paste from the Tapestry pages. I know you can use a search engine; I want to know if you understand the work you are doing.

Feedback

Each year I make changes in response to prior course evaluations, but you don't need to wait for the end of the quarter—if there is something you think I should know about the course, please take the time to tell me while there is still time to do something about it. It won't always be possible, but a conversation is a good place to start.