Arch 486: Computer Graphics
Programming for Design

Qtr/SLN/Inst: Spring, 2009; (SLN: 10319)  
When/Where: M-W-F: 10:30-11:20 a.m.  
Brian Johnson  
Gould 007F

Office/Hours/Email: Arc G55; M & F: 11:30-12:00 or by appt.; brj@u.washington.edu
Course web site: http://quicksilver.be.washington.edu/courses/arch486
Prerequisites: Computer literacy comparable to Arch 380.
Goals: To provide a stimulating and fun introduction to programming via the Java programming language; to empower you to create your own applications, whether for serious or light-hearted purposes.

Through a series of programming exercises you will explore concepts related to programming (objects, data types, loops, branching, variables, events, etc.) and graphics (pixels, lines, arcs, color, mouse input, windows, clipping, animation, etc.). You will learn fundamental programming concepts and terminology through the (freely available) Java language, with a focus on playful, designerly responses to the opportunities placed before you. In addition to a number of 2D applets (programs for use in web pages), you will be exposed to, and have an opportunity to explore, 3D issues and client/server programming concepts.

Expectations: That each student will actively engage in the exploration of the material covered in the course, attend class, share discoveries with classmates, speak up when confused or bored, submit exercises on time, and challenge themselves to explore creative opportunities in exercises.

Exercises: There will be up to 9 required exercises, in addition to which students will design and execute a term project for presentation at the end of the quarter.

Grading: Grading will be based on evaluations of each project completed during the quarter (40%), a personal term project (30%), a final exam (20%), and on attendance and participation in class exchanges and presentations (10%).

Individual project evaluations will consider both the technical (code) and expressive (creative) character of the work.

Extra Credit: will be available as a "challenge" extension to each applet exercise. Completed extra-credit work will generally raise the grade on the related exercise. In addition, you may complete more than one of the "optional applet" exercises (#9) for extra credit. Extra "optional applet" projects will be used to replace the grade of low-scoring required applet projects.
Exercises: Each exercise includes a “seed” applet distributed and discussed in class. Your job is to implement the “seed” and then extend or enhance the applet in some relevant but personal and creative way (we can discuss ideas). Results will be presented using web pages.

Exercises are named for the new concepts introduced in them:

“Hello World” – a traditional getting-started exercise to work out process kinks and get going.

“2D Graphics” – fundamental drawing actions, coordinates, colors; used to draw a simple graphic.

“Flow Control” – condensing and controlling repetitive action via loops, branches, classes, etc.

“GIN” – Java event model, basic mouse input methods; create a paint-like drawing environment. (GIN comes from “Graphic INput”)

“OOP” – “objects,” instances, packaging data and action (OOP stands for Object Oriented Programming)

“Animation” – Animation and motion; as easy as bouncing a ball.

“Arrays” – combining animation with multiple instances of a class, launch a number of balls into a space and watch them bounce.

“GUIs” – Using simple interface elements, construct a rudimentary drawing program. (GUI comes from “Graphical User Interface”)

“Options” (a “one of” option) – This week I’ll talk about three different extensions to previous content. Your job is to pick one and run with it. Possibilities: 3D wireframe; Applet Input and output; Clients & Servers (make your own shared whiteboard?)

Term Project: Each student (or possibly a small group) will develop a term project, the goal of which is to design and implement some more complete and complex operation using a Java applet. The applet, completely or substantially of your own creation, might provide (a) an interactive game, (b) a teaching experience, (c) a toolkit for display of dynamic building information, (d) a design-utility of some sort, (e) a web-presentation utility. We will discuss ideas, design, and implementation issues throughout the quarter.

Games: Connect-4, network tic-tac-toe, life, go, 9-square, ...

Didactics: Thermal behavior of materials, browse weather data, ...

Toolkit: Various charting applets for weather, temperature, flow, over differing time domains, etc.

Design Utility: Graphical Bulletin Board, Sun-shade designer, ..?

Web Presentation: pano or object viewers (ala QTVR), magic lens magnifier, ...

You tell me: something that strikes you as interesting and doable...