COURSES AND PROJECTS TAUGHT AT SEATTLE UNIVERSITY
William Bricken
June 2001

COURSES

Quarter	Course		Students	Course type
W96	SE560	Human-Computer Interaction	24	new elective (Adjunct)
Su96	SE561	Programming GUIs	19	new elective (Adjunct)
A96	SE592C	Client-Server Architectures	30	new elective
W97	SE560	Human-Computer Interaction	19	repeat elective
	SE500	Data Structures and Algorithms	7	new core required
Sp97	SE561	Programming the Interface	17	new elective
	SE543	Applied Formal Methods	6	new elective
A97	SE502	Mathematical Foundation	17	new core required
	SE553	Artificial Intelligence	15	new elective
W98	SE543	Applied Formal Methods	9	repeat elective
S98	SE500	Data Structures and Algorithms	20	repeat core required
A98	SE560	Human Computer Interaction	11	repeat elective
	SE502	Math Foundations	23	repeat core required
W99	SE561	Programming the Interface	5	repeat elective
S99	SE564	Computer Graphics	8	new elective
	SE500	Data Structures and Algorithms	12	repeat core required
A99	SE502	Mathematical Foundations	17	repeat core required
	SE553	Artificial Intelligence	8	repeat elective
W00	SE543	Applied Formal Methods	9	repeat elective
S00	SE500	Data Structures and Algorithms	17	repeat core required
	SE514	Programming Methods	25	new core required
A00	SE502	Mathematical Foundations	16	repeat core required
	SE560	Human Computer Interaction	20	repeat elective
W01	SE593	Computer Ethics	12	new elective

YEAR LONG CAPSTONE PROJECTS (ALL REQUIRED)

Year	Project		Sponsor	Students
96	SE585-6-7	KAoS Java	Boeing	5
	SE585-6-7	KAoS DCOM	Boeing	5
97	SE585-6-7	KAoS	Boeing	5
	SE585-6-7	WebSET	Rockwell Collins	5
	CS487-8-9	Undergrad KAoS	Boeing	5
98	SE585-6-7	KAoS	Boeing	5
	CS487-8-9	Undergrad KAoS	Boeing	5
99	SE585-6-7	KAoS	Boeing	5
00	SE585-6-7	Boundary Numbers	Bricken	3
	SE585-6-7	Wireless Listings	Elektrorbit	5

MSE capstone projects are one year (three quarters) long, although most teams begin in the previous summer. Supervision includes client liaison and assurance, meeting with students especially during difficult times, reviewing all project materials, providing resources and administrative troubleshooting for the team, technical guidance when needed, and in general serving as a source of stability and morale.

INDEPENDENT STUDY

Quarter	Course		Students	Course type
Su97	SE596C	Independent Study	2	new elective
A97	SE596C	Independent Study	1	new elective
Su98	SE596C	Independent Study	1	new elective
98-99	SE585-6-7	Independent Study Project	1	required capstone
S99	CS496	Independent Study	2	undergrad elective
S99	SE596	Independent Study	1	new elective
S00	SE596	Independent Study	2	new elective
Su00	SE596	Independent Study	1	new elective
A00	SE596	Independent Study	3	new elective

I am deeply committed to providing MSE students with an academic study option. Therefore, I eagerly support independent study students, under three conditions, 1) not too time consuming, 2) I know something about the subject, and 3) the student exhibits a burning desire to do individual work. All these projects are voluntary and not included in my teaching load. This effort has been very rewarding; I have learned a lot from the research projects, students are thrilled, and often this option permits a student to graduate at the end of a year.

COMMENTARY ON COURSEWORK

Both SE502 (Mathematical Foundations) and SE543 (Applied Formal Methods) have developed into stimulating and innovative courses, with a surprising interest in formal methods from the enrolled students. Formal methods is a strong stream for our program, although enrollment numbers are marginal.

The HCI class, SE560, has begun to be difficult to teach, due to the complete revision of course content over the last four years. The internet has redefined the meaning and tools of HCI, and within a few years, even the desktop metaphor (and probably the concept of personal computers) will be obsolete. HCI itself is in turmoil, again because the HCI of five years ago is largely irrelevant to interface projects and questions today. My own specialization in VR is largely irrelevant to non-entertainment applications.

SE500 too is a teaching challenge. Conventional algorithm analysis addresses toy domains (sorting, searching) which have been over-analyzed, and are now simply functions in several languages. Pointer chasing implementations, so widely stressed in algorithms texts, are obsolete and bias toward antiquated procedural programming approaches. Object-oriented and functional techniques are seen as fields separate to data structures and algorithms. In sum, the content and materials of this topic are substantially obsolete, yet this fact is ignored in conventional course materials. I have been unable to find an appropriately modern textbook.

SE561, Programming the Interface, has shown low enrolment interest. After polling the eligible students, I found that 90% would prefer a course on *Computer Ethics*. I am currently developing such a course for Winter 2001.

SE564 (Computer Graphics) and SE553 (Artificial Intelligence) are both specialty fields that I have actively participated in. Teaching these courses is a joy, since I can bring to bear so much personal experience. The students greatly appreciate being taught a subject by someone who has worked intimately in the field.

My latest course, SE514, Programming Methods, is one I have wanted to teach for quite a while. The first pass at organizing the course was enjoyable, although I am eager to rebuild that curriculum based on the classroom teaching experience. This topic has room for three valid courses over an entire year.

The capstone project teams have surprised me on how much social interaction can dominate technology development, and on the level of emotional/professional immaturity of some of our students. The projects thus provide an excellent context for issues not covered in classrooms.

I am pleased with the initial growth of independent study projects, and hope to continue to support special individual needs of students.