From:	Polivka, Barbara
To:	Neelam Soundarajan
Cc:	psaday@gmail.com; Soave, Melissa; weide.1@osu.edu; gordon.1@osu.edu; sadayappan.1@osu.edu; blackwell.4@osu.edu; zerby.8@osu.edu; polivka.1@osu.edu; antani.2@buckeyemail.osu.edu; Cogdell, James
Subject:	RE: Computational Science and Engineering UG Minor - request for response
Date:	Wednesday, July 27, 2011 4:59:37 PM
Attachments:	Computational Science and Engineering Minor program.pdf

Neelan,
Thanks for your quick response and for being able to attend CAA tomorrow.

Melissa - can you post the email below and the attached to the CAA materials for tomorrow's (7/28) meeting.

Thanks

Barbara J. Polivka, RN, PhD
Associate Professor
Program Director, Nursing and Health Systems Management
Ohio State University College of Nursing
1585 Neil Ave.
Columbus, OH 43210
polivka.1@osu.edu
614-292-4902

From: Neelam Soundarajan [neelam@cse.ohio-state.edu]
Sent: Wednesday, July 27, 2011 4:28 PM
To: Polivka, Barbara
Cc: psaday@gmail.com; soave.2@osu.edu; weide.1@osu.edu; gordon.1@osu.edu; sadayappan.1@osu.edu; blackwell.4@osu.edu; zerby.8@osu.edu; neelam@cse.ohio-state.edu; polivka.1@osu.edu
Subject: RE: Computational Science and Engineering UG Minor - request for response

Hi Barbara,

Saday is out of town but I was able to talk to him over the phone.

> One of the members of our subcommittee reviewing your proposal had
> some additional questions about the minor. The minor is on the
> agenda for tomorrow's (7/28) meeting (begins at 2PM). We would
> appreciate your responding to these items and making sure a
> representative is at the meeting tomorrow. Melissa Soave - who is
> cc'd on this email - can give you specifics about the meeting.

I will be at the meeting. I understand, from a message from Melissa to Saday, that the program is likely to be discussed at about 2:30. I will try to be there at that time.

We have updated the proposal to address the points you raise and I am attaching a copy of that at the end of this message. I will also address them below so you don't have to search through the proposal:

> -Has a program plan code abbreviation been determined?

Tentatively we have decided on CMPSCENG-MN. I hope this is not considered cast in stone!

> - In the rationale for change in CH you state the maximum hours able

> to be taken for the minor is 18. It is in fact 19.

Thanks! We have fixed it.

> - You state CSE major students are allowed to take the minor because
> so many non CSE courses are offered. However, it is possible to
> complete the minor with all CSE courses, except in the simulation
> and modeling category. With this knowledge, how can you still
> justify letting CSE major students take the minor?

We have added a clause, at the top of Attachment #3, saying that at least 6 hours must be from outside the dept. of the student's major. (The problem you point out actually applies to some other students as well, for example students from the Materials Sc. and Eng. program. That is why we phrased this in a generic way rather than just for CSE majors.) We should also note that, in Engineering, it has been considered quite acceptable to have courses double counted in the major and minor, the idea being that if a student has the knowledge and skills required for a particular minor, the fact that some (or all) of those knowledge and skills are also part of the student's major should not be an issue. So requiring at least 6 hours from outside the major dept. will, we hope, address this concern.

> -Are Arts and Sciences students approved to take this minor? It
> seems it does not meet the ASC rule of a certain amount of courses
> above the 2000 or 3000 level requirement they have for minors.

We will have to take up the question of ASC students at a later stage. It may be that we will have to propose a somewhat different program that is tailored to them and meets ASC's requirements.

> -Normally, students are allowed a total of 6 hours of transfer
> credit to be applied to a minor.

That is a good point. We have added a clause, at the top of Attachment #3, allowing a maximum of 6 hours of transfer credit.

Thank you and I look forward to the CAA meeting tomorrow.

Best wishes,

--Neelam

==================================
Neelam Soundarajan
Computer Science & Engineering
Ohio State University
Columbus, OH 43210

e-mail: neelam@cse.ohio-state.edu
soundarajan.1@osu.edu

tel: (614) 292 1444
==================================

From: Soave, Melissa
To: Soave, Melissa
Subject: FW: Computational Science and Engineering Grad Minor - request for response
Date: Friday, July 22, 2011 2:01:28 PM
Attachments: Quarter-Semester-Equivalences-Computational-Science.xlsx
Coputational-Science-Advising-Sheet-Semesters.xls

From: P Sadayappan [psaday@gmail.com]
Sent: Thursday, July 14, 2011 12:03 PM
To: Polivka, Barbara
Cc: weide.1@osu.edu; gordon.1@osu.edu; Sadayappan.1@osu.edu; Marilyn Blackwell; Henry Zerby; Neelam Soundarajan
Subject: Re: Computational Science and Engineering Grad Minor - request for response

Dear Dr. Polivka,

I apologize for the delay in responding to your request. I was away on an international travel and had very limited internet access. After I got back I provided Prof. Soundarajan the requested comparison sheet with quarter and semester courses. In the process of compiling that sheet, I found that information was available for some semester courses for which we previously did not have information and one or two of the previously planned semester courses were no longer in the current list proposed by departments. I had sent to Prof. Soundarajan the comparison sheet as well as a revised minor advising sheet for semesters, so that he may forward a package to you with any additional appropriate materials. It appears that he is now on travel. I am attaching to this mail what I had sent him:

1) Side-by-side comparison sheet for the quarter and semester versions of courses for the minor
2) Revised semester version of advising sheet for the minor

Please let me know if you need any other information. I will be away on travel again starting June 24 but prior to that I would be happy to meet if desired to explain anything that may be unclear.

Regards,
-- Saday

P. Sadayappan
Professor
Department of Computer Science & Engineering

On Thu, Jul 14, 2011 at 9:49 AM, Polivka, Barbara <bpolivka@con.ohio-state.edu> wrote:
> Drs. Gordon, Weide, and Sadayappan,
> Last week I sent my second request for clarification/additional information regarding the Undergraduate minor in Computational Science and Engineering. We would greatly appreciate a response as soon as possible so that this proposal can be reviewed by CAA.
>
> Thanks
>
> Barbara J. Polivka, PhD, RN
> Associate Professor
> Specialty Track Director, Nursing and Health Systems Management The
> Ohio State University College of Nursing
> 324 Newton Hall
> 1585 Neil Avenue
> Columbus, OH 43210
> (614) 292-4902
> polivka.1@osu.edu
>
>

>
>
> -----Original Message-----
> From: Polivka, Barbara [mailto:bpolivka@con.ohio-state.edu]
> Sent: Thursday, July 07, 2011 3:10 PM
> To: weide.1@osu.edu; gordon.1@osu.edu; Sadayappan.1@osu.edu
> Cc: Marilyn Blackwell
> Subject: FW: Computational Science and Engineering Grad Minor
>
> Hello,
> A while back I sent out the email below asking a question/request regarding the Computational Science and Engineering UG minor. If someone could respond to our request we could bring the proposal to the subcommittee and then to CAA for approval.
>
> Thanks so much.
>
> Barbara J. Polivka, PhD, RN
> Associate Professor
> Specialty Track Director, Nursing and Health Systems Management The
> Ohio State University College of Nursing
> 324 Newton Hall
> 1585 Neil Avenue
> Columbus, OH 43210
> (614) 292-4902
> polivka.1@osu.edu
>
>
>
> -----Original Message-----
> From: Polivka, Barbara [mailto:bpolivka@con.ohio-state.edu]
> Sent: Friday, June 24, 2011 11:41 AM
> To: gagan agrawal
> Cc: Marilyn Blackwell; zhang.574@osu.edu
> Subject: Computational Science and Engineering Minor
>
> Henry Zerby and I are the workgroup members of CAA Subcommittee B who are reviewing the semester conversion proposal for the UG Minor in Computational Science and Engineering. We appreciate all of the information that was provided regarding this unique minor.
>
> We had one question/request for additional information.
>
> It would be helpful if you provided a table that included the approved OSU quarter-based course options and the equivalent semester based courses and some rational for any changes in course options. For example - we noticed that in Simulation and Modeling in the quarter-based Advising sheet there are 6 course options - while in the semester based advising sheet there are 7 course options. It's not clear what was added and why. Similarly - in Numerical Methods there are 7 quarter course options, but only 5 semester based options.
>
> Again - a table listing quarter and semester equivalent courses as well as a brief explanation regarding changes in course options would be helpful to us.
>
> Thank you.
>
> Barbara J. Polivka, PhD, RN
> Associate Professor
> Specialty Track Director, Nursing and Health Systems Management The
> Ohio State University College of Nursing
> 324 Newton Hall
> 1585 Neil Avenue
> Columbus, OH 43210

QUARTER EQUIVALENCES

Topic	Quarter Course	Semester Course	Notes
Calculus (Pre-Req)	Math 151	MATH 1151	
	Math 152	MATH 1152 or 1172	
	Math 153	MATH 1152 or 1172	
Simulation & Modeling		BIOMEDE 5430	FEM in BME (New)
		CHBE 5790	Model & Simulation (New)
	CEG 640		No Semester Equivalent
	CSE 778		No Semester Equivalent
	ISE 521	ISE 2010	
	ISE 704	ISE 5100	
	ME 785	MECHENG 5139	
	MSE 533	MATSCEN 2321	
Programming & Algorithms	CSE 202	CSE 1222	
	CSE 221	CSE 2221	
	CSE 294P	CSE 1221	
Numerical Methods	AAE 581	AEROENG 3581	
	CEG 406	CIVILEN 2060	
	CSE 541	CSE 5361	
	ECE 715	ECE 5510	
	Math 606		No Semester Equivalent
	Math 607	MATH 3607	
	ME 250	MECHENG 2850	
Optimization (no longer required in Semester version but allowed as an elective	CEG 776	CIVILEN 3080	
	ECE 759	ECE 5759	
	ISE 522	ISE 3200/3210/3990.3/5200	
	ME 761	MECHENG 7761	
	MSE 600	MATSCEN 4181	
Capstone Research	CE 660	CIVILEN 4000.01	
	CSE 699	CSE 4998	
	CSE H783	CSE 4999H	
	ME 564	MECHENG 4901.01	
	ME 565	MECHENG 4901.02	
	MSE 695	MATSCEN 4381	

Discipline-Specific		AEROENG 5615	Intro CFD (new)
Computationally		CHBE 5734	Mol Informatics (New)
Oriented Course	CSE 630	CSE 3521	
	CSE 655	CSE 3341	
	CSE 660	CSE 2431	
	CSE 670	CSE 3241	
	CSE 675	CSE 3421	
	CSE 680	CSE 2331	
		CSE 3461	Comp. Networking (New)
		CSE 3541	Comp. Gaming (New)
	Chem 644		New Course# Not Known
	MSE 756	MATSCEN 6756	
	Phys 780		New Course# Not Known
Elective	Math 255	MATH 2255	
	Math 415	MATH 2415	
	Math 568	Math 2568	
	Math 571	Math 2568	
	CSE 621	CSE 5441	
	CSE 694L	CSE 5544	

Attachment #1:

Computational Science and Engineering Minor Proposed Program Reqmts

• Prerequisites for minor: Calculus I and II (Math ____ and ____)
• Approved courses from other participating RRSCS
institutions may be substituted; see program director for

Core (Required): Simulation and Modeling	Course Number	Cr-hrs	Completed
Finite Element Applications in BME	BIOMEDE 5430	3	
Modeling and Simulation	CHBE 5790	3	
Systems Modeling	ISE 2010	3	
Discrete Event Simulation	ISE 5100	3	
Applied Finite Element Method	MECHENG 5139	3	
Modeling and Simulation Lab I	MATSCEN 2321	3	
Total Simulation and Modeling (one course)			

Core (Required): Programming and Algorithms	Course Number	Cr-hrs	Completed
Intro to Computer Prog in MATLAB for Engrs and Scientist	CSE/ENGINEER 1221	2	
Intro to Computer Prog in C++ for Engrs and Scientists	CSE/ENGINEER 1222	3	
Software I: Software Components	CSE 2221	4	
Total Programming and Algorithms (one course)			

Core (Required): Numerical Methods	Course Number	Cr-hrs	Completed
Numerical Methods in Aerospace Engineering	AEROENG 3581	3	
Numerical Analysis Methods for Civ/Env Eng Apps	CIVILEN 2060	4	
Numerical Methods	CSE 5361	3	
Introduction to Computational Electromagnetics	ECE 5510	3	
Beginning Scientific Computing	MATH 3607	3	
Introduction to Numerical Methods	MECHENG 2850	3	
Total Numerical Methods (one course)			

iscipline-Specific (Required): Capstone Research/Internsh	Course Number	Cr-hrs	Completed
Computationally oriented capstone course(s) or indiv research			
Total CapstoneResearch/Internship Experience (≥ 2 cr-hrs)			

Discipline-Specific (Required): Computational Study	Course Number	Cr-hrs	Completed
Introduction to Computational Aerodynamics	AEROENG 5615	3	
Molecular Informatics	CHBE 5734	3	
Computational Chemistry	Chem 644(sem)	3	
Foundations II: Data Structures and Algorithms	CSE 2331	3	
Systems II: Introduction to Operating Systems	CSE 2431	3	
Introduction to Database Systems	CSE 3241	3	
Principles of Programming Languages	CSE 3341	3	
Introduction to Computer Architecture	CSE 3421	3	
Computer Networking and Internet Technologies	CSE 3461	3	
Survey of Artificial Intelligence I: Basic Techniques	CSE 3521	3	
Computer Game and Animation Techniques	CSE 3541	3	
Computational Materials Modeling	MATSCEN 6756	2	
Total Computational Study (one course)			

Electives	Course Number	Cr-hrs	Completed
Economic Evaluation and Optimization in Civ/Env Eng	CIVILEN 3080	3	
Introduction to Parallel Computing	CSE 5441	3	
Introduction to Scientific Visualization	CSE 5544	3	
Optimization for Static and Dynamic Systems	ECE 5759	3	
Optimization for Enterprise Systems	ISE 3200	3	
Optimization for System Design	ISE 3210	3	
Engineering Optimization	ISE 3990.3	2	
Linear Optimization	ISE 5200	3	
Differential Equations and Their Applications	MATH 2255	3	
Ordinary and Partial Differential Equations	MATH 2415	3	
Linear Algebra	MATH 2568	3	
Total Electives (one course)			

THE OHIO STATE UNIVERSITY

College of Engineering

122 Hitchcock Hall
2070 Neil Avenue
Columbus, OH 43210-1278

Phone 614-292-2651
FAX 614-292-9379
E-mail engosu@osu.edu

Date: 6 April 2011

To: Randy Smith
Vice Provost, Office of Academic Affairs

From: Ed McCaul
Secretary, College of Engineering Committee on Academy Affairs (CCAA)

Subject: Semester Conversion Proposal for the Undergraduate Minor in Computational Science and Engineering

Attached is a letter from Xiaodong Zhang, Department Chair of Computer Science and Engineering, as well as a semester conversion proposal for their Undergraduate Minor in Computational Science and Engineering. Their proposal was reviewed by a subcommittee of CCAA. After reviewing the proposal and having some changes made to it the subcommittee recommended to the full committee that it be approved. After a discussion, CCAA approved the proposal on the 5th of April 2011 and requested that I forward it to you for consideration by CAA. If you have any questions concerning this proposal please let me know.

To: Engineering College Committee on Academic Affairs

From: Xiaodong Zhang, CSE Department Chair

Date: 9 March 2011

Re: Semester Proposals for ***Minor in Computational Science and Engineering***

The faculty of Computer Science and Engineering have worked diligently to prepare the attached proposal. The CSE Semester Task Force comprising about fifteen CSE faculty members, academic advising staff, and undergraduate and graduate students, began meeting weekly at the start of Au09 to plan the semester conversion for all our programs. Data collected during these deliberations included historical feedback from CSE graduates (compiled as part of accreditation-based assessment processes over the past 10+ years), input from the CSE Department Industrial Advisory Committee, a survey of all CSE faculty on various issues related to the transition, the Undergraduate Forum (an annual open meeting with undergraduate students), and comparisons with about a dozen computer science and engineering, computer science, and similarly named programs at major peer institutions. Given the nature of the *Minor in Computational Science and Engineering* program, key input about all aspects of the program was also provided by Dr. Steve Gordon of OSC.

The faculty have voted to approve the attached proposal as our semester plans for the ***Minor in Computational Science and Engineering***, and I also recommend approval. The vote of the CSE faculty members on the Minor in Computational Science and Engineering proposal was obtained electronically following our procedures for such votes; 39 were in favor, 0 opposed, 0 abstentions. Some aspects of the proposal have been revised based on feedback received from the CCAA subcommittee that considered the proposal, of the one we had originally submitted. The feedback enabled us to streamline and simplify the program. We also apologize for the errors in the previous submission dated Feb. 28 caused by cut-and-paste errors; and thank CCAA's Subcommittee A for its diligent work in pointing out the errors. We hope we have corrected them all in this version.

____

Xiaodong Zhang

Robert M. Critchfield Professor, and CSE Department Chair

Minor in Computational Science and Engineering

Primary Contacts: Neelam Soundarajan (soundarajan.1, 292-1444), Steven Gordon (sgordon@osc.edu, 292-6082), and P. Sadayappan (sadayappan.1, 292-0053)

1. *Fiscal Unit / Academic Organization*

Department of Computer Science and Engineering (1435)

2. *Administering College / Academic Group*

College of Engineering / Department of Computer Science and Engineering (CSE)

3. *Co-administering College / Academic Group*

Not applicable

4. *Semester Conversion Designation*

c. Converted with minimal changes to program goals and/or curricular requirements

5. *Program / Plan Name*

Minor in Computational Science and Engineering

6. *Type of Program*

Undergraduate minor

7. *Program Plan Code Abbreviation*

CMPSCENG-MN

8. *Degree Title*

Not applicable

9. *Specializations / Sub-plans*

Not applicable

10. *Program Learning Goals*

Not required at this time for minors

11. *List of Semester Courses*

See Attachment #1: Computational Science and Engineering Minor Proposed Program Requirements.

12. *Program Rationale*

The Computational Science and Engineering Minor was approved in Sp 2010. As the program has just commenced, minor changes are proposed along with those necessitated by converting course requirements to semesters.

The basic requirements of this minor program are guided by a document associated with the Ralph Regula School of Computational Science, which was established as a “virtual school” by the Ohio Board of Regents in December 2005. RRSCS is a run from the Ohio Supercomputer Center and does not offer degrees of its own but serves to organize and coordinate statewide efforts to integrate computational science programs into the curricula at participating institutions, one of which is OSU. Computational science and engineering is an interdisciplinary field with expertise scattered among different departments and different institutions—though OSU offers expertise and coursework in all these areas. The intention of RRSCS is to sponsor shared, inter-institutional programs that take advantage of the existing expertise, make it widely available, and limit the duplication of effort and expense where possible. For completeness, and to answer any questions that might arise about this unusual aspect of this program, the full proposal for the just-approved minor program is provided as Attachment #4. One change in the previously approved minor is to make the “optimization” course an elective rather than a requirement. This keeps the total credit hours upon conversion to semesters within the desired range.

13. Quarters Curriculum Advising Sheet

See Table 1: Current Advising Sheet, which shows the minor program proposal as previously approved as in the original document (Attachment #4) and presently serves as the Computational Science and Engineering Minor Advising Sheet.

14. Semesters Curriculum Advising Sheet

See Table 2 and Attachment #3: Proposed Advising Sheet. Attachment #5 provides a side-by-side comparison of the quarter and semester versions of courses for the minor.

15. Curricular Map

Not applicable

16. Associated Pre-Major or Area of Interest

Not applicable

17. Credit-Hour Changes

	Number of qtr-cr-hrs in current program[1]	Calculated result for 2/3 of current qtr-cr-hrs	Sem-cr-hrs required for proposed prog.	Change in cr-hrs
Total minimum cr-hrs required for completion of program	22	14.7	15	+0.3
Required cr-hrs offered by the unit	4 - 19	2.7 - 12.7	2 - 12	–0.7
Required cr-hrs offered outside of the unit	3 - 18	2.0 - 12.0	3 - 13	+1.0 - +1.0

[1] Numbers in this column are computed by considering the OSU courses listed in Attachment #2 that would be needed to satisfy the current requirements in all competency areas of the minor coursework. The official overall minimum for the minor is listed as 20 cr-hrs rather than 22 cr-hrs, as some courses at other institutions might qualify as substitutes for these OSU courses.

Required prerequisite cr-hrs not included above	15	10.0	10	0.0

18. Rationale for Significant Change in Credit Hours

The minimum number of semester credit hours for the minor program is 15 credits. However, the actual number of credit hours for students to complete the minor is likely to be in the range of 15-19 credits, because most of the courses in the various required categories for the program (please see Attachment 3) are 3-credit courses. Simply because of the "breakage" in cr-hrs as OSU courses are converted to semesters in different units *in the best interest of majors in those units* (i.e., without regard to the impact on this interdisciplinary minor program), a direct conversion of the minor program to meet the overall coverage requirements of the RRSCS curriculum entailed an increase in the minimum from an equivalent of 14.7 sem-cr-hrs to 18 sem-cr-hrs, with the average expected credits to be even higher. Therefore, the originally required "Optimization" course was moved to the elective category, resulting in the lowered credit requirement as shown above.

19. Transition Policy

No student who begins the Computational Science and Engineering Minor under quarters will have progress toward completion impeded by the transition to semesters. Computational Science and Engineering Minor requirements beginning Summer 2012 will be those in force for students under semesters; but *every* quarter-credit-hour that would have counted toward a Computational Science and Engineering Minor under the quarter-based program will count (as 2/3 of a semester-credit-hour) toward the requirements for the semester version. If necessary, a revision of specific requirements will be worked out for any Computational Science and Engineering Minor student who is caught in the transition, in consultation with the CSE Associate Chair.

— Xiaodong Zhang, CSE Department Chair

20. Assessment Practices

Not applicable

Table 1: Current Requirements for Undergraduate Computational Science Minor Current Advising Sheet (Quarter Version)		
Topic	**Courses**	**Required/Elective**
Prerequisites		
Calculus	Math 151, 152, 153 (or equivalent, e.g. Math 161, 162 or Math H190, H191)	
Core Courses		
Simulation and Modeling	One of: CEG 640, CSE 778, ISE 521, 704, ME 785, MSE 533	Required
Programming and Algorithms	One of: CSE 202, CSE 294P, CSE 221	Required
Numerical Methods	One of: AAE 581, CEG 406, CSE 541, ECE 715, Math 606, Math 607, ME 250	Required
Optimization	CEG 776, ECE 759, ISE 522, ME 761, MSE 600	Required
Discipline Specific Courses		
Capstone Research/Internship Experience (3 credits minimum)	CE 660, CSE 699 or H783,ME 564 or 565, MSE 695 or other approved individualized research credits	Required
Discipline-specific Computationally oriented Course	CSE 630, 655, 660, 670, 675, 680, Chem 644, MSE 756, Phys 780	Required
Elective: Choose at least one course from the following		
Differential Equations and Discrete Dynamical Systems	Math 255, Math 415, Math 568, Math 571	Elective
Parallel Programming	CSE 621	Elective
Scientific Visualization	CSE 694L	Elective

Table 2: Undergraduate Computational Science Minor Requirements (Semester Version)

	Topic	Course	Cr. hrs	Semesters offered	Req/Elective
Prerequisites	Calculus	MATH 1151	5	Au, Sp	
		MATH 1152 or 1172	5	Au, Sp	
Core Courses	Simulation and Modeling (One of the Courses)	BIOMEDE 5430	3	Au	Required
		CIVILEN 5680	3	Sp	
		CHBE 5790	3	TBD	
		ISE 2010	3	Sp	
		ISE 5100	3	Au, Sp	
		MECHENG 5139	3	Au	
		MATSCEN 2321	3	Sp	
	Programming and Algorithms (One of the Courses)	CSE 1222	3	Au, Sp	Required
		CSE 1221	2	Au, Sp	
		CSE 2221	4	Au, Sp	
	Numerical Methods (One of the Courses)	AEROENG 3581	3	Au	Required
		CIVILEN 2060	4	Sp	
		CSE 5361	3	Au, Sp	
		MATH 3607	3	Sp	
		MECHENG 2850	3	Au, Sp	
Discipline Specific Courses	Capstone Research/Internship Experience (minimum 2 credits)	CIVILEN 4000.01	2	Au, Sp	Required
		CSE 4998, or MECHENG 4901.01, or 4902.01, or other approved research credits	2-5	Au, Sp, Su	
	One discipline-specific Computationally oriented Course	AEROENG 5615	3	Au	Required
		CHBE 5734	3	TBD	
		CHEM 644 (sem)	3	TBD	
		CSE 2331	3	Au, Sp	
		CSE 3241 or 3341	3	Au, Sp	
		CSE 3421 or 3521	3	Au, Sp	
		CSE 3461	3	Au, Sp	
		CSE 3541	3	Au, Sp	
		ECE 5510	3	Au	
		MATSCEN 6756	3	Au, Sp	
Elective: Choose at least one elective course from the following	Differential Equations and Discrete Dynamical Systems	MATH 2255	3	Au, Sp	Elective
		MATH 2415	3	Au, Sp	
	Parallel Computing	CSE 5441	3	Au	Elective
	Scientific Visualization	CSE 5544	3	Sp	Elective
	Optimization	CIVILEN 3080	3	Sp	Elective
		ECE 5759	3	Au	
		ISE 3200	3	Au, Sp	
		ISE 3210	3	Au, Sp	
		ISE 3990.3B	2	TBD	
		ISE 5200	3	Au	
		MATSCEN 4181	2	Au	
		Total minimum credit hours: 15			

Attachment 2: Current Course Requirements for Computational Science Minor

Topic	**Courses**	**Required/Elective**
Prerequisites		
Calculus	Math 151, 152, 153 (or equivalent, e.g. Math 161, 162 or Math H190, H191)	
Core Courses		
Simulation and Modeling	One of: CEG 640, CSE 778, ISE 521, 704, ME 785, MSE 533	Required
Programming and Algorithms	One of: CSE 202, CSE 294P, CSE 221	Required
Numerical Methods	One of: AAE 581, CEG 406, CSE 541, ECE 715, Math 606, Math 607, ME 250	Required
Optimization	CEG 776, ECE 759, ISE 522, ME 761, MSE 600	Required
Discipline Specific Courses		
Capstone Research/Internship Experience (3 credits minimum)	CE 660, CSE 699 or H783,ME 564 or 565, MSE 695 or other approved individualized research credits	Required
Discipline-specific Computationally oriented Course	CSE 630, 655, 660, 670, 675, 680, Chem 644, MSE 756, Phys 780	Required
Elective: Choose at least one course from the following		
Differential Equations and Discrete Dynamical Systems	Math 255, Math 415, Math 568, Math 571	Elective
Parallel Programming	CSE 621	Elective
Scientific Visualization	CSE 694L	Elective

Attachment 3: Undergraduate Minor in
Computational Science and Engineering

Computing has become the key enabler of fabulously rapid advances across nearly all disciplines of the academy and throughout all segments of society. Many recent advances of this kind are the result of scientists and engineers building detailed models of physical systems and then simulating their behavior, or solving relevant systems of equations numerically, using powerful hardware and software. Students who understand how to do this are becoming increasingly valuable to employers. Moreover, looking at problems through the lens of "computational thinking" can bring new insights to students' future research, for those interested in graduate education.

The ***Computational Science and Engineering Minor***, administered by the OSU Department of Computer Science and Engineering (CSE) and offered under the auspices of the State of Ohio's Ralph Regula School of Computational Science, is designed to provide such knowledge and skill. It educates undergraduate students in both conceptual and practical computational aspects of science and engineering.

Impact for the Student

Completion of the program leads to a transcript designation that can and should be advertised to prospective employers. Candidates with not only discipline-specific knowledge in computational science and engineering but also a clear conception of computational thinking are very attractive to most employers and prospective graduate schools.

Curriculum

The Computational Science and Engineering Minor consists of six courses (comprising a minimum of 15 cr-hrs) in five required competency areas plus one elective area. Rules of the College of Engineering allow majors in Engineering to "double-count" any of these courses in their major. Because the minor program is offered under the auspices of the Ralph Regula School of Computational Science—a "virtual" school run through the Ohio Supercomputer Center—many courses to meet the requirements are also available from other Ohio institutions who are partners in the RRSCS.

Prerequisites

All students earning the Computational Science and Engineering Minor must complete the first two semesters of calculus (for engineers and scientists). Most of the courses that could count toward this minor are academically accessible to undergraduate students majoring in Engineering, Mathematics, Physical Sciences, and some areas of the Biological Sciences, based only on courses already required in those majors or other courses counted in the minor.

Getting Started

Prospective students should begin by contacting the CSE Department's Undergraduate Advising Office (DL 374, 2-1900).

Course options listed on reverse.
For more details, please see:
http://www.cse.ohio-state.edu/ugrad/cseminor.shtml

Computational Science and Engineering Minor Proposed Program Requirements

• Prerequisites for minor: Calculus I and II (Math 1151.xx and (Math 1152.xx or Math 1172))
• Approved courses from other participating RRSCS institutions may be substituted; see program director for details.
• Total minimum semester hours required – 15
• No more than 6 credit hours may be *transfer* credits.
• Students must take at least 6 credit hours from *outside* their *major* department.

Core (Required): Simulation and Modeling	**Course Number**	**Cr-hrs**
Finite Element Applications in BME	BIOMEDE 5430	3
Modeling and Simulation	CHBE 5790	3
Systems Modeling	ISE 2010	3
Discrete Event Simulation	ISE 5100	3
Applied Finite Element Method	MECHENG 5139	3
Modeling and Simulation Lab I	MATSCEN 2321	3
Total Simulation and Modeling (one course)		
Core (Required): Programming and Algorithms	**Course Number**	**Cr-hrs**
Intro to Computer Prog in MATLAB for Engrs and Scientists	CSE/ENGINEER 1221	2
Intro to Computer Prog in C++ for Engrs and Scientists	CSE/ENGINEER 1222	3
Software I: Software Components	CSE 2221	4
Total Programming and Algorithms (one course)		
Core (Required): Numerical Methods	**Course Number**	**Cr-hrs**
Numerical Methods in Aerospace Engineering	AEROENG 3581	3
Numerical Analysis Methods for Civ/Env Eng Apps	CIVILEN 2060	4
Numerical Methods	CSE 5361	3
Introduction to Computational Electromagnetics	ECE 5510	3
Beginning Scientific Computing	Math 3607	3
Introduction to Numerical Methods	MECHENG 2850	3
Total Numerical Methods (one course)		
Discipline-Specific (Required): Capstone Research/Internship	**Course Number**	**Cr-hrs**
Computationally oriented capstone course or individual research		
Total CapstoneResearch/Internship Experience (≥ 2 cr-hrs)		
Discipline-Specific (Required): Computational Study	**Course Number**	**Cr-hrs**
Introduction to Computational Aerodynamics	AEROENG 5615	3
Molecular Informatics	CHBE 5734	3
Computational Chemistry	Chem 644(sem)*	3

Foundations II: Data Structures and Algorithms	CSE 2331	3
Systems II: Introduction to Operating Systems	CSE 2341	3
Introduction to Database Systems	CSE 3241	3
Principles of Programming Languages	CSE 3341	3
Introduction to Computer Architecture	CSE 3421	3
Computer Networking and Internet Technologies	CSE 3461	3
Survey of Artificial Intelligence I: Basic Techniques	CSE 3521	3
Computer Game and Animation Techniques	CSE 3541	3
Introduction to Computational Electromagnetics	ECE 5510	3
Computational Materials Modeling	MATSCEN 6756	2
Total Computational Study (one course)		

Electives	**Course Number**	**Cr-hrs**
Economic Evaluation and Optimization in Civ/Env Eng	CIVILEN 3080	3
Introduction to Parallel Computing	CSE 5441	3
Introduction to Scientific Visualization	CSE 5544	3
Optimization for Static and Dynamic Systems	ECE 5759	3
Optimization for Enterprise Systems	ISE 3200	3
Optimization for System Design	ISE 3210	3
Engineering Optimization	ISE 3990.3	2
Linear Optimization	ISE 5200	3
Materials Selection	MATSCEN 4181	2
Differential Equations and Their Applications	Math 2255	3
Ordinary and Partial Differential Equations	Math 2415	3
Total Electives (one course)		

*Chem 644 has not yet been renumbered but will be offered as a semester course with a new number

Attachment #4

Minor Program in Computational Science

Introduction

Computational science describes the application of computing, especially high performance computing, to the solution of scientific and technical problems. Computational scientists use computers to create mathematical models that help them simulate and understand the operation of natural and mechanical processes, as well as to visualize the operation and results of these models.

Computational science (i.e., science in-silico) has become a third way of advancing knowledge along with the traditional methods of theory and experimentation. In-silico simulations and modeling afford the opportunity to "see" the unattainable – phenomena that are too small (atoms and molecules), too large (galaxies and the universe), too fast (photosynthesis), too slow (geological processes), too complex (automobile engines), or too dangerous (toxic materials). In recent years, computational studies have produced enormous advances in almost all fields of scientific and engineering inquiry, including DNA sequencing, behavioral modeling, global climatic predictions, drug design, financial systems, and medical visualization.

In recognition of the importance of computational science, the Ohio Board of Regents created the Ralph Regula School of Computational Science (RRSCS) in December 2005. RRSCS is a "virtual" school housed at the Ohio Supercomputer Center that will not offer degrees of its own but will serve to organize and coordinate statewide efforts to integrate computational science programs into the curricula at participating institutions. This effort recognizes that the field is an interdisciplinary field with expertise scattered among different departments and different institutions. The intention of the school is to sponsor shared, inter-institutional programs that take advantage of the existing expertise, make it widely available, and limit the duplication of effort and expense where possible.

The first such program is a minor program in computational science intended initially for majors in science and engineering programs at the participating institutions. The program is being initiated as a part of a National Science Foundation grant to nine Ohio higher education institutions, the Ohio Learning Network, and the Ohio Supercomputer Center that pledged to cooperate in the creation of the program. The participating institutions are shown in Table 1. **Each of these institutions is separately approving the minor program so that it is officially established as a part of their degree programs. This is the proposal to do so at OSU.**

A minor is the most appropriate approach to this area at the undergraduate level because domain expertise extending from a field of science, mathematics, or engineering is required to understand and implement computational modeling. The minor program also will require fewer resources to implement while helping to organize the required inter-institutional agreements and operating procedures for the program.

Table 1: Institutions Participating in the NSF CI-TEAM Grant for Computational Science
Capital University
Central State University
Columbus State Community College
Kent State University
Ohio Learning Network
Ohio Supercomputer Center
Sinclair Community College
The Ohio State University
University of Cincinnati
Wittenberg University
Wright State University

Overview of the Minor Curriculum

The minor curriculum was created through a collaborative effort of the faculty at the participating institutions. Because of its inter-institutional nature, the curriculum is competency-based, defining learning outcomes as the basis for courses to allow for easier evaluation and approval of proposed courses. The competencies were also reviewed by a business advisory committee including major, prospective employers of our graduates. They helped to revise the proposed structure to best suit the need for graduates in the current market. The full description of the competencies is provided in Appendix A.

The proposed minor curriculum consists of six required courses and at least one elective. These are shown in Table 2. All students are required to take a year of calculus, which is part of the current requirements for the target major fields. Calculus could be taken concurrently with the introductory modeling and simulation course, but is a prerequisite for the other courses. The courses are described in the table of competencies shown in Appendix A.

In addition to the approval of the proposed minor program in Computational Science through the formal process at The Ohio State University, the courses proposed as a part of the curriculum will be reviewed by an inter-disciplinary, inter-institutional committee of computational science faculty from around the state to ensure that they meet the required competencies. The creation of course materials and conversion of existing courses to match the competencies and offer some courses at distance is being subsidized by an NSF grant and other funding at the RRSCS. The materials produced for those courses will be retained in a repository at RRSCS so that faculty can draw upon those shared materials to expand the number of courses offerings.

Coursework Requirements

The Computational Science minor requires the completion of at least 20 credits of approved coursework, including a required course in each of six designated areas and one elective, as summarized in Table 2.

Because of the inter-institutional nature of the program, students have two options for the minor courses. They can take the courses at OSU, if they are offered here, or they can take the courses at distance from one of the other participating institutions. Transfer credits from courses at other institutions will be handled in the usual way by the registrar's office under the collaborative agreement (details may be found in Appendix A, page 30). Table 3 shows a list of pertinent courses for the 2008-2009 academic year offered at different institutions. Thus OSU is not obligated to teach every course in the program; also some of the courses could be taught less frequently. In the current proposal, OSU courses are identified (Table 2) that meet or exceed the minimum competencies associated with the topics in the minor program. For some of the upper division courses that exceed the competencies, other pre-requisites are required (listed in Appendix B).

Required Courses

1) Simulation and Modeling: An introductory course on the use of models (continuous and discrete) and simulation in science and engineering. The overall goal is to introduce the need for modeling and simulation as an integral part of science and engineering practice, provide an introduction to modeling concepts, review the mathematics underlying deterministic models, build an understanding of both the computational and estimation errors associated with models, introduce stochastic or Monte Carlo simulations, and demonstrate an ability to formulate, use, and explain a project model. Greater detail on the list of competencies is available in Appendix A (page 14-16). A general introductory course in simulation and modeling is currently not offered at OSU. Students may take this course remotely at another institution (offerings for 2008-2009 are listed in Table 3; there are no prerequisites for the distance courses at other institutions as the level of expertise and mathematics required can be met by most college freshman upon entry), or use a more advanced domain-specific course that focuses on modeling and simulation. Table 2 lists suitable courses at OSU; details on pre-requisites and quarters of offering are provided in Appendix B.

2) Programming and Algorithms: A first course on computational thinking, use of a programming language for problem solving, and development of algorithms. In addition to several existing courses at OSU that would satisfy this requirement, a new Matlab-based course on computational thinking in the context of science and engineering (CSE 294P) is being developed by the CSE Department (pilot in Spring 2009), with science/engineering students as the target audience.

3) Numerical Methods: An applied introduction to use of numerical methods in solving linear and nonlinear equations, interpolation, numerical solution of differential equations. Because many engineering students either take a linear algebra course or are introduced to linear algebra concepts in numerical methods and the modeling and simulation course, we do not require a linear algebra course.

4) Optimization: Courses on the topic of optimization are being developed at some other institutions in support of a Computational Science minor. A course is also being considered for development by the IWSE department at OSU. Several departments have domain-specialized courses that focus on discrete and/or continuous optimization techniques, which may also be used to satisfy this requirement

5) Capstone Research/ Internship Experience: Each student must complete a guided research project or internship on a computational topic. The mechanism used to satisfy this requirement may differ across departments – e.g. computationally oriented senior design project or honors thesis, or a computationally oriented independent research study with a faculty member at Ohio State. The Ralph Regula School also plans to create a list of computationally oriented projects contributed from faculty at the participating universities; such a guided project can also be used to satisfy the capstone requirement. At OSU, the CSE Department will maintain a list of computational projects from faculty at Ohio State, and coordinate the matching of interested students with faculty guiding the projects. A student from any department may register for CSE 699 if they choose one of these computational projects for satisfying the computational capstone requirement.

6) Discipline-Specific Course: Any approved computationally oriented course from the student's major discipline. The current list of approved courses is listed in Table 2.

Elective Courses

At least one elective course must be completed, chosen from any of the following topics.

a) Differential Equations and Dynamical Systems: A course on numerical solution of differential equations, which are fundamental in modeling physical and biological systems. Many engineering majors are required to take differential equations as are those in some of the physical sciences. We did not make this a requirement in the minor program to accommodate majors in the biological sciences and fields where other mathematical or analytical approaches are more prevalent.

b) Parallel Computing: An introduction to parallel programming, parallel algorithm development, implementation and optimization. Courses offered at distance in this area generally require an introductory computer algorithms course or related programming course as a prerequisite.

c) Scientific Visualization: An introduction to tools and techniques for visualization of large-scale data produced by computational simulations. A version of this course will be offered by OSU CSE in Spring 2009.

Administration and Advising

The minor in Computational Science will be open to any undergraduate student at the Ohio State University. The Computer Science and Engineering department will serve as a primary agent for the proposed program, but a Computational Science Advisory Committee will be formed, with broader membership. The Committee will be chaired by a faculty member from CSE (appointed by the Department Chair), and it will include an additional faculty member from the College of Engineering (appointed by the Associate Dean of Undergraduate Education and Student Affairs), one faculty member from the College of Mathematical and Physical Sciences (appointed by the Associate Dean), and the Director of the Ralph Regula School of Computational Science. The Committee will meet at least once every year to review the minor curriculum and course offerings, and make revisions as necessary, including approval of new discipline-specific computationally oriented courses.

The undergraduate counseling office of the Department of Computer Science will assist with student advising for the Computational Science minor. It will make available a form that students may use to declare the Computational Science minor. Any petitions for substitution of alternative courses (taken at Ohio State or another university) to satisfy the requirements of the minor must be approved by the Computational Science Advisory Committee.

Because this minor program is unique and generally outside the required CSE curriculum, CSE majors will be allowed to take it.

For the capstone course, we expect the committee to create a set of general guidelines that will be given to the faculty with potentially eligible courses. The guidelines will ensure that approved courses or internships have sufficient computational applications to qualify as a capstone for this minor program. Faculty will be asked to sign a form indicating that a particular course meets the guidelines and to submit the final summary reports from the projects annually to the minor program committee. This will allow the committee to evaluate whether the capstone courses are continuing to meet the guidelines as well as provide important information on the overall effectiveness of the program.

Table 2: Requirements for Undergraduate Computational Science Minor		
Topic	**Courses**	**Required/Elective**
Prerequisites		
Calculus	Math 151, 152, 153 (or equivalent, e.g. Math 161, 162 or Math H190, H191)	
Core Courses		
Simulation and Modeling	One of: CEG 640, CSE 778, ISE 521, 704, ME 785, MSE 533	Required
Programming and Algorithms	One of: CSE 202, CSE 294P, CSE 221, EG 167	Required
Numerical Methods	One of: AAE 581, CEG 406, CSE 541, ECE 715, Math 606, Math 607, ME 250	Required
Optimization	CEG 776, ECE 759, ISE 522, ME 761, MSE 600	Required
Discipline Specific Courses		
Capstone Research/Internship Experience	CE 660, CSE 699 or H783,ME 564 or 565, MSE 695 or other approved individualized research credits	Required
Discipline-specific Computationally oriented Course	CE/ECE 675, CSE 630, 655, 660, 670, 675, 680, Chem 644, MSE 756, Phys 780	Required
Elective: Choose at least one course from the following		
Differential Equations and Discrete Dynamical Systems	Math 255, Math 415, Math 568, Math 571	Elective
Parallel Programming	CSE 621	Elective
Scientific Visualization	CSE 694L	Elective

Table 3: 2008-2009 Academic Year Courses Offered at Collaborating Institutions

Course Number and Title	Institution / Instructor	Quarter or Semester*	Credit Hours
PHY 212 Introduction to Modeling and Simulation	Sinclair Community College / Art Ross & Bob Chaney	Winter quarter 2009	4
PHY 220 Introduction to Computational Physics	Sinclair Community College / Art Ross	Spring quarter 2009	5
CSAC 245 Introduction to Computational Science	Capital University / Dr. Patrick Shields	Fall semester 2008	3
BSCI-50/70195 BSCI 40195/ BTEC-40220 ST: Bioinformatics	Kent State University / Dr. Helen Piontkivska	Fall semester 2008	3
20-CS-668 Parallel Computing	University of Cincinnati / Dr. Fred Annexstein	Fall Quarter 2008	3
Chem 644 Computational Chemistry	The Ohio State University / Dr. Richard Spinney	Fall Quarter 2008 Classroom only	3
CSE 694L Data and Information Visualization Description	The Ohio State University / Dr. Raghu Machiraju	Spring Quarter 2009	4
COMP 345 Optimization	Wittenberg University / Eric A. Stahlberg	Fall Semester 2008 Classroom only	3
MATH 299 Special Topics: Differential Equations and Discrete Dynamical Systems	Columbus State Community College / John Nedel	Spring quarter 2009 Classroom only	6
CS/PBIO 416/516 Problem Solving with Bioinformatics Tools	Ohio University / Dr. Lonnie Welch and Sarah Wyatt	Spring Quarter 2009	4
CS 412 Parallel Computing	Ohio University / Dr. Frank Drews	Spring Quarter 2009	5
CST120 Computational Science Methods	Stark State Community College / Robert Berens	Fall Semester 2008 Classroom only	3
CST121 Introduction to Modeling and Simulation	Stark State Community College / Karen Hardesty	Spring Semester 2009 Classroom only	3
CSA 443 High Performance Computing and Parallel Programming	Miami University / Dr. Dhananjai M. Rao	Spring Semester 2009	3
15PHYS410 Computational Physics	University of Cincinnati / Richard Gass	Spring Quarter 2009	3

*All courses offered at distance unless otherwise indicated.

Some of the courses will be given temporary course numbers for the first year as they go through the approval process for a permanent course number at the participating institutions.

The RRSCS will maintain a central website that students and faculty can use to see what courses and being offered. The site will also show how to register for courses being offered at other institutions. The proposed mechanisms for these procedures are currently being discussed and are discussed briefly below. The proposed inter-institutional agreements have been discussed at all of the participating institutions and we are working with the appropriate offices to implement those agreements. The agreement is shown as Appendix A.

Inter-institutional Arrangements

As part of the NSF project, a subcommittee of representatives of academic affairs officers and registrars has been meeting to discuss the inter-institutional arrangements for the minor program. They have agreed in principle to several things:

- The registration process for students should be as easy as possible.
- RRSCS should maintain a list of approved courses by year so it is relatively easy to conduct degree audits.
- Transfer of credits from other institutions can be handled by the registrars at the institutions as those courses are completed.
- Tuition will be collected at the home institution of the student based on the total number of credit hours taken including hours taken from other institutions and charged at the home institution rate.
- There should be some compensation given to the institutions teaching courses to students at other institutions in lieu of tuition.
- Petitions for exceptions to the required curriculum and other individual advising will remain solely at the home institution.
- Mechanisms for changes and additions to the minor curriculum should be undertaken by the RRSCS and a statewide committee of faculty.

Appendix A

Computational Science Minor Program
Detailed Program Description

Ralph Regula School of Computational Science
Cyberinfrastructure Team Project
Summary of Undergraduate Minor Program Requirements
September 2006

The minor in computational science will be offered by a number of higher education institutions in Ohio. Degrees will be offered by the students "home" institution but it will possible for the student to take minor program materials from other, participating institutions. We have started with a minor because we believe that each student needs some domain expertise in a major field before being able to complete computationally-based projects in related areas.

There is a committee reviewing the various options for the sharing of materials, instruction, and revenues associated with cross-registered students. They will make some recommendations to the participating institutions in the coming months on the institutional procedures.

Because of the inter-institutional nature of the program, we have designed it to be competency-based. The instructional materials associated with each competency or subset of competencies can then be embedded either in existing courses on each campus, in new courses at one or more campuses, or as stand-alone modules that might be taken at distance and at the time that each student is ready for them. Competency will then be demonstrated through one or more assessments of the student's abilities on a combination of exams and projects. The competency-based approach is preferred because it gives curriculum flexibility to participating programs and gives employers some assurance of the working knowledge of each graduate.

The draft competencies created by the participating faculty has been reviewed by a business advisory committee. They confirmed the majority of the recommended competencies at a meeting on May 31, 2006 and offered some advice on topic emphasis and breadth. This document reflects their comments and is the basis on which faculty are currently proposing instructional modules that meet the competencies.

The computational science minor will enable science and engineering majors to apply computational tools to the problems in their discipline. As such, the minor breaks into three broad categories: Prerequisites, the Computational Science Core Competencies and the Discipline Specific Competencies. At this stage, we are focused on the competencies that comprise the Computational Science Core. The goal of the Core competencies is to establish a foundation that can be leveraged in the Discipline Specific Competencies that are directly applicable to the practice of the student’s chosen profession. The competencies are shown in Table 1.

Our remaining decisions involve deciding what competencies are associated with each broad area, how many elective competencies are required for the minor, and the specific competencies for each discipline-oriented course. Those will be addressed in Autumn 2006 as we begin to pre-test the materials in classrooms around Ohio.

Table 1: Competencies and Requirements for Undergraduate Computational Science Minor		
Topic	**Subtopics**	**Required/Optional**
Prerequisites		
Calculus 1 and 2		Required
Computational Science Courses		
Simulation and Modeling	Should use one of the major symbolic programs Maple, MATLAB, Mathematica; need for local, hands-on support at outset of course	Required
Programming and Algorithms	Logic of programming whether a traditional computer science or programming with a symbolic language like Maple, MATLAB, Mathematica	Required
Differential Equations and Discrete Dynamical Systems	Depending upon major; linear algebra may also be needed by some	Elective
Numerical Methods	Need for a project-based course which touches the most important topics rather than a standard math course	Required
Optimization	An important topic; could be part of a modeling course or integrated with numerical methods	Required
Parallel Programming		Elective
Scientific Visualization		Elective
Discipline Specific Courses		
Capstone Research/Internship Experience		Required
Discipline Oriented Courses		One required; probably only one per field for awhile

Minor Program in Computational Science
Competency/Topic Overview
Area 1: Simulation and Modeling

Competency/Descriptors
Explain the role of modeling in science and engineering **Descriptors:** Discuss the importance of modeling to science and engineering Discuss the history and need for modeling Discuss the cost effectiveness of modeling Discuss the time-effect of modeling (e.g. the ability to predict the weather) Define the terms associated with modeling to science and engineering List questions that would check/validate model results Describe future trends and issues in science and engineering Identify specific industry related examples of modeling in engineering (e.g., Battelle; P&G, material science, manufacturing, bioscience, etc.) Discuss application across various industries (e.g., economics, health, etc.)
Analyze modeling and simulation in computational science **Descriptors:** Identify different types of models and simulations Describe a model in terms of iterative process, linking physical and virtual worlds and the science of prediction Explain the use of models and simulation in hypothesis testing (e.g. scientific method)
Create a conceptual model **Descriptors:** Illustrate a conceptual modeling process through examples Identify the key parameters of the model Estimate model outcomes Utilize modeling software and/or spreadsheets to implement model algebraic equations (e.g. Vensim, Excel, MATLAB, Mathematica) Construct a simple computer visualization of the model results (e.g. infectious disease model, traffic flow, etc.) Validate the model with data Discuss model quality and the sources of errors
Examine various mathematical representations of functions **Descriptors:** Describe linear functions Define non-linear functions (e.g., polynomials, exponential, periodic, parameterized, etc.) Visualize functions utilizing software (e.g. Excel, Function flyer, etc.) Determine appropriate functional form to fit the data Demonstrate essential mathematical concepts related to modeling and simulation
Analyze issues in accuracy and precision **Descriptors:** Describe various types of numerical and experimental errors Explain the concept of systematic errors

Explain the concept of data dependent errors Illustrate calculation and measurement accuracy Identify sources of errors in modeling and approaches to checking whether model results are reasonable
Understand discrete and difference-based computer models **Descriptors:** Explain the transition of a continuous function to its discrete computer representation Represent “rate of change” using finite differences Cite examples of finite differences Explain derivatives and how they relate to model implementation on a computer Write pseudo-code for finite difference modeling
Demonstrate computational programming utilizing a higher level language or modeling tool (e.g. Maple, MATLAB™, Mathematica, other) **Descriptors:** Describe the system syntax (e.g., menus, toolbars, etc.) Define elementary representations, functions, matrices – arrays, script files, etc. Explain programming and scripting processes (e.g., relational operations, logical operations, condition statements, loops, debugging programs, etc.) Create tabular and visual outputs (e.g., 2-D and 3-D subplots) Translate the conceptual models to run with this system and assess the model results (e.g. traffic flow and/or “spread of infectious disease”) Illustrate other people’s models utilizing the modeling program
Assess computational models **Descriptors:** Assess problems with algorithms and computer accuracy Discuss techniques and standards for reviewing models Verify and validate the model Discuss the differences between the predicted outcomes of the model and the computed outcomes and relevance to the problem Discuss the suitability and limits of the model to address the problem for which the model was designed
Build event-based models **Descriptors:** Describe event-based modeling (e.g. SIMULINK™; Extend, ARENA) Run existing models Translate conceptual models (e.g., traffic flow utilizing SIMULINK™)
Complete a team-based, real-world model project **Descriptors:** Identify a problem, create mathematical model and translate to computational modeling Organize and present project proposal Document model development and implementation Collaborate with team members to complete the project
Demonstrate technical communication **Descriptors:** Demonstrate technical writing skills in the comprehensive report

Demonstrate verbal communication skills in an oral presentation Create and present visual representation of model and results Address all components of a comprehensive technical report Respond to peer review

Minor Program in Computational Science
Competency/Topic Overview
Area 2: Programming and Algorithms

Competency/Descriptors
Describe the fundamentals of problem solving **Descriptors:** Understand Top-Down thinking and program design Discuss breaking up a problem into its component tasks Understand how tasks acquire data Describe how tasks should be ordered Represent tasks in a flow-chart style format Understand the difference between high-level languages (for example Mathematica, Maple or MATLAB), medium level languages (for example FORTRAN or C) and low-level languages (assembler) and when each should be used.
Understand and write Pseudo code **Descriptors:** List the basic programming elements of Pseudo code Explain the logic behind an if/then/else statement Understand the iterative behavior of loops Describe the difference between several looping constructs Write Pseudo code to solve basic problems Understand how to represent data flow in and out of subprograms.
Use subprograms in program design **Descriptors:** Describe how logical tasks can be implemented as subprograms Understand the logical distinction between functions and subroutines Explain the control flow when a function is called Define dummy and actual arguments Discuss the different relationships dummy and actual arguments Explain how function output is used Understand how languages handle passed data into functions and subprograms, especially one and two dimensional arrays.
Write code in a Programming language **Descriptors:** Understand the concept of syntax in a programming language Describe the syntax of the programming language constructs List the type of subprograms available in the language Explain the concepts of argument pass-by-value and pass-by-reference Understand what a compiler and linker do Understand the difference between a compiled and interpreted language Understand the difference between a typed and an un-typed language Understand the difference between a source file and an executable file Write and run basic programs in the language of choice Understand how to de-bug code and how to"sanity check" code. Understand the importance of user-interfaces: clear input instructions including physical

units if needed and clearly formatted and labeled output Understand the numerical limits of various data types and the implications for numerical accuracy of results.
Use different approaches to data I/O in a program **Descriptors:** Explain the advantages and disadvantages of file I/O Describe the syntax for file I/O in your programming language Compare binary and ASCII file I/O Write code using file I/O and keyboard/monitor I/O
Understanding and use of fundamental programming Algorithms **Descriptors:** Explain an algorithm as an ordered series of solution steps Describe an algorithm for a simple programming problem Learn and use “classic” programming algorithms from a field of interest to the student. If possible, these should be algorithms used in the student’s discipline. Describe what a software library is Understand how library functions implement algorithms Write code to implement your own version of “classic” algorithm Compare with code using a library function Understand data flow into library functions and implications of selecting any “tuning parameters” or options that may be required.
Explain various approaches to Program Design **Descriptors:** Describe Functional decomposition (Top-down Problem Solving) Be familiar with different programming styles (e.g. function, procedural, rule based) Understand how to modularize code Understand the benefits of code re-use Explain the operation of a Boss-Worker design Compare designs based on Global Variables vs. self-contained functions Define Object-Oriented Programming (OOP) Contrast OOP with functional decomposition Explain the power of Inheritance in OOP Understand how to document code Understand how to write and when to use stubs and drivers.

Minor Program in Computational Science
Competency/Topic Overview
Area 3: Differential Equations and Discrete Dynamical Systems

Competency/Descriptors
Describe the solution methodology for first order linear differential and difference equations **Descriptors:** Analyze modeling problems with first order differential equations and present their solution methodology (e.g. liner, homogeneous, exact) Analyze modeling problems with first order difference equations and present their solution methodology (e.g. homogeneous, non-homogeneous). Analyze long term behavior
Describe the solution methodology for systems of linear first order differential and difference equations **Descriptors:** Describe modeling problems with systems of first order differential equations and present their solution methodology (e.g., homogeneous with constant coefficients, variation of parameters) Describe modeling problems with systems of first order difference equations and their solution methodology (e.g., homogeneous with constant coefficients)
Describe the solution methodology for higher order differential and difference equations **Descriptors:** Describe modeling problems with higher order differential equations analyze their solution methodology (e.g., homogeneous, non-homogeneous, undetermined coefficients, variation of parameters) Describe modeling problems with higher order difference equations analyze their solution methodology (e.g., homogeneous, non-homogeneous). Analyze the long-term behavior.
Describe the solution methodology for differential equations using the Laplace Transforms **Descriptors:** Discuss the Laplace transformation of (e.g., continuous , discontinuous, delta and convolution) functions Describe modeling problems with differential equations and present their solution methodology using Laplace transformations (use of CAS, Maple, Mathematica)
Describe the solution methodology for non-linear differential equations **Descriptors:** Describe the concept of an equilibrium point Model with non-linear differential equations and present the phase –portrait analysis Understand and demonstrate how chaos is generated in the solution process of non-

linear differential equations.
Describe the solution methodology for non-linear difference equations **Descriptors:** Describe the method of linearization Describe the concepts of Logistic and Henon Maps Model with non-linear difference equations and demonstrate understanding of fundamental concepts from Bifurcation theory (e.g., fixed, periodic points, chaos) Describe techniques for controlling chaos Understand concepts of numerical accuracy applied to each solution approach

Minor Program in Computational Science
Competency/Topic Overview
Area 4: Numerical Methods

Competency/Descriptors
Understand number representation and computer errors **Descriptors:** Understand the pros and cons of floating point and integer arithmetic Describe various kinds of computing errors (e.g., round-off, chopping) Describe absolute, relative error and percent error Discuss error propagation Describe loss of significance – methods to avoid loss of significance
Analyze methods for solving non-linear equations **Descriptors:** Discuss and contrast fixed point methods (e.g., bisection, secant, Newton's) for a single equation Describe a fixed point method for a system of equations (e.g., Newton's)
Describe techniques for solving systems of linear equations **Descriptors:** Describe the naïve Gauss elimination and the partial pivoting method Understand the concepts of condition number and ill-conditioning problems Discuss and contrast factorization methods (e.g., LU, QR, Cholesky, SVD) Discuss and contrast iterative methods (e.g., Jacobi, Gauss Siedel) Describe convergence and stopping criteria of iterative methods
Analyze techniques for computing eigenvalues—eigenvectors (Optional) **Descriptors:** Describe and give examples of eigenvalue –eigenvector problems using specific, applied examples and their significance Describe canonical forms of matrices Describe and contrast direct methods for computing eigenvalues (e.g., power method, inverse power method) Describe and contrast transformation methods (e.g., QR algorithm)
Describe interpolation and approximation methods **Descriptors:** Describe and contrast interpolation methods (e.g., Lagrange, Chebyshev, FFT) Describe interpolation with spline functions (e.g., piecewise linear, quadratic, natural cubic) Discuss approximation using the method of least squares (linear .vs. non-linear)
Describe numerical methods for Ordinary Differential Equations **Descriptors:** Describe and compare basic methods for IVPs (e.g., Euler, Taylor, Runge-Kutta) Describe and compare predictor-corrector methods Describe and compare multistep methods Discuss and contrast numerical methods for BVPs (e.g., shooting method, finite difference method) Compare the accuracy, memory requirements, and precision of each of the approaches

Describe numerical methods for Partial Differential Equations **Descriptors:** Describe and compare numerical methods for parabolic PDEs (e.g., finite difference, Crank-Nicolson) Describe numerical methods for hyperbolic PDEs Describe numerical methods for elliptic PDEs (e.g. finite difference, Gauss-Seidel) Discuss the finite element method for solving PDEs
Describe Monte Carlo Methods Describe applications of Monte Carlo models with examples Discuss algorithms for Monte Carlo methods

Minor Program in Computational Science
Competency/Topic Overview
Area 5: Optimization

Describe and use Optimization techniques **Descriptors:** Describe and contrast unconstrained optimization methods (e.g., Golden section search, Steepest descent, Newton's method, conjugate gradient, simulated annealing, genetic algorithms) Describe and contrast constrained optimization methods (e.g., Lagrange multiplier, quasi-Newton, penalty function method
Implement linear and non-linear programs Analyze linear programming methods (e.g., simplex method) Describe non-linear programming methods (e.g., interior, exterior, mixed methods) Demonstrate ability to correctly use software systems (e.g., Matlab, IMSL, NAG) to solve practical optimization problems

Minor Program in Computational Science
Competency/Topic Overview
Area 6: Parallel Programming

Competency/Descriptors
Describe the fundamental concepts of parallel programming and related architectures **Descriptors:** Describe the differences between distributed and shared memory architectures Describe the difference between domain and functional decomposition in parallel Describe a parallel programming approach to an introductory problem Compare parallel, distributed, and grid computing concepts
Demonstrate parallel programming concepts using MPI **Descriptors:** Describe the MPI programming model Create, compile, and run an MPI parallel program Create MPI programs that utilize point-to-point communications Create an MPI program that uses point-to-point blocking communications Create an MPI program that uses point-to-point non-blocking communications Create an MPI program that uses collective communications Create an MPI programs that use parallel I/O Create MPI programs that use derived data types Create MPI programs that use vector derived data type Create MPI programs that use structure derived data type
Demonstrate knowledge of parallel scalability **Descriptors:** Use mathematical formulas to determine speed-up and efficiency metrics for a parallel algorithm. Demonstrate the use of graphical systems such as MATLAB to display speed-up and efficiency graphs
Demonstrate knowledge of parallel programming libraries and tools **Descriptors:** Demonstrate the use of performance tools for profiling programs (e.g., GNU GPROF or MATLAB profiler) Create parallel programs with calls to parallel libraries (e.g. BLAS, BLACS, ScaLAPACK or FFTW) Demonstrate the use of MPI tracing tools (e.g., VAMPIR) to determine parallel performance bottlenecks

Minor Program in Computational Science
Competency/Topic Overview
Area 7: Scientific Visualization

Competency/Descriptors
Define SciVis needs; relationships to human visualization; basic techniques Define Scientific Visualization (Sci Vis) Discuss needs of SciVis (in the framework of a large variety of possible application areas) Survey different platforms for Visualization (e.g. AVS, VTK, OpenGL, VRLM) Discuss the different techniques and visualization methods used in SciVis Explain the human visualization system – capabilities and perceptions Explain the different steps in the visualization pipeline Discuss different sources of data for SciVis and explain the terms applied to data types (i.e. scalar, vector, normal, tensor) Discuss different types of grids (e.g., regular vs. irregular grids) Discuss the different methods used to gather data Describe and explore the use of different file formats for sharing data (netCDF, XML, TIFF, GIF, JPEG, Wavefront OBJ) Discuss limitations of different methods Discuss future applications in emerging fields Metadata needs for graphics libraries
Overview of computer graphic concepts **Descriptors:** Overview of SciVis concepts (pixels, rgb colors, 3D coordinate system, mapping 3D data to a 2Dscreen, continuous vs. discrete) Discuss polygonal representation Discuss lighting/shading Overview of classification/segmentation and transfer functions Discuss concept of rendering pipeline (no details about matrices) Discuss hardware rendering (mainly for polygonal models, few specialized volumetric hardware cards) Identify terms used in virtual space and in graphics elements Navigate in virtual space and manipulate primitive objects ▪ Transform: scale, rotate, translate) ▪ Manipulate surface ▪ Manipulate lighting and camera Explore colormaps and examine conceptual definitions for different color maps (pertaining to color spaces HSV, RGB, etc.) as related to representing data and relationships to perception
Describe approaches to visualization for different scientific problems **Descriptors:** Examine different computational solutions to scientific problems Explain the different techniques used in visualization (i.e. glyphs, iso-contours, streamlines, image processing, volume-data) Examine the application of problems to visualization techniques

Utilize software tools to implement visual image of a solution Discuss the use of time in animation
Utilize software to implement grid representations of data **Descriptors:** Identify the various cell representations (i.e. points, polygons, 3d geometries) Discuss the application to different grid types (i.e. structured, unstructured, random) Discuss raycasting methods and texture mapping Examine the details of raycasting sampling (FAT(low resolution sampling), interpolation techniques). Examine algorithms: Direct Composite, SFP, use of transparency. Identify the grid representation and data(color reps.) (regular grids, 1, 8, 24 and 32 bit color information) Discuss algorithms for manipulating images, distortion, fft's, enhancement, restoration, frequency domains Utilize software to implement different grid types Discuss limitations of grids

Use visualization software to display an isosurface **Descriptors:** Discuss different data types used: scalar vs. vector data Discuss the different grid types Discuss the different algorithms (Marching Cubes etc) Introduce details of the system being used in a course (e.g., VTK, AVS, etc.) Apply the system to extract and display an isosurface of some data set (could be tailored towards the teacher's and student's interests/application areas) Discuss limitations of these methods
Use visualization software to complete a volumetric rendering Descriptors: Discuss direct volumetric rendering (raycasting and texture mapping) and its advantages/disadvantages vs. surface rendering Discuss segmentation/classification and transfer functions Discuss and illustrate how to use a system (VTK, AVS, etc.) to do volumetric rendering) Using the system, visualize a data set using raycasting Using the system, visualize a data set using texture mapping Discuss limitations of this method
Utilize visualization software to visualize a vector dataset **Descriptors:** Discuss vector data Discuss different methods for vector visualizing (particles, stream ribbons, vector glyphs, etc.) Discuss the use of structured grid types: (ir)regular, cylindrical, spherical Discuss application areas for vector visualization (air flow, etc.) Using a system (VTK, AVS, etc.), visualize a vector data set Discuss limitations of this method
Explore examples of image processing **Descriptors:** Discuss basic steps and goals in image processing Discuss variety of data sources of images and how they can be represented Discuss algorithms used for image processing Explore examples of image processing (e.g., noise reduction, image enhancement, feature extraction etc) Discuss challenges and limitations in image processing
Use advanced techniques applied to a real problem **Descriptors:** To be chosen by instructor based on instructor/student interest. Among suggested topics are: - visualizing an irregular grid; - visualizing a data set specific to the area of interest (see 3.2 and 3.3 for specific examples) - writing a segmentation tool - implementing a visualization algorithm from scratch (such as marching cubes or raycasting)
Examine SciVis problems for Biological Sciences – "OMICS" applications

Descriptors: Examine different problems existing in 'OMICS sciences that require visualization solution (overview) Discuss challenges of representing biomedical/biological data (i.e., representing protein structure or genomic sequence with all their attributes as a visual metaphor) Discuss challenges associated with visualization of scattered data such as text information and bioinformatics data (e.g., phylogenetic information) Gene finding in genomic sequences - Examine different components of a gene structure - Visualize genomic structure of an individual gene - Build a comparison between genomic features from several genomes o Visualize (and examine) similarities and differences o Discuss goal-dependent options of parsing the results to be explored elsewhere (e.g., as plain text, XML-marked) Protein folding and protein structure prediction o Discuss the differences between protein folding and protein structure prediction o Explore different methods used in protein folding and structure prediction o Apply different methods of protein structure prediction and compare the results o Construct, visualize and examine structure-based protein alignment Biological networks (e.g., protein-protein or protein-DNA interaction networks) Visualization of various types of expression data o Discuss and contrast different types of expression data – e.g., microarray gene expression data, protein expression data o Discuss different visualization (and analyses) techniques used for expression data o Apply (and compare outcomes) hierarchical clustering and k-means clustering to the same gene microarray expression data o Discuss pros and cons of different clustering methods, their shortcomings, and ways to access the quality of clusters Discuss potential applications of SciVis techniques in biomedical and drug design fields Utilize MATLAB to implement/solve the above problems
Explore SciVis techniques in BioMedical applications **Descriptors:** Explore a variety of biomedical applications of SciVis to explore large datasets such as MRI and confocal microscopy data Overview of volume visualization techniques in biomedical problems Examine and different ways MRI (Magnetic Resonance Imaging) data can be visualized (e.g., 2-D image versus isocontour slices). Discuss potential applications (interpretation) of each of the techniques. Utilize software tools (MATLAB, VTK) to apply the techniques above

RALPH REGULA SCHOOL OF COMPUTATIONAL SCIENCE CONSORTIAL AGREEMENT

REGULA SCHOOL CURRICULUM

Institutions participating in the Ralph Regula School of Computational Science will offer a multi-institutional, interdisciplinary undergraduate minor in computational science with courses starting in the fall of 2007. The effective date of this agreement will be August 15, 2007. Standardized certificate programs to create workforce knowledge and skills valued by industry also are under development. Graduate courses will be added by 2009. The undergraduate minor curriculum offering for 2007-2008 is as follows:

Course Title and Number	Institution/Instructor	Quarter or Semester*	Credit Hours
CS 399 Selected Topics in Computational Science Programming and Algorithms	Wright State Ronald Taylor, Computer Science	Fall quarter 2007	4
Special Topics Introduction to Modeling and Simulation Part 1	Sinclair Art Ross Physics	Fall quarter 2007	2
Special Topics Introduction to Modeling and Simulation Part 2	Sinclair Art Ross Physics	Winter quarter 2008	2
Computational physics	Art Ross Physics	Spring quarter 2006	4
CSAC 245 Introduction to Modeling and Simulation	Capital University John Philips and Pat Shields	Fall semester 2007	3
Bioinformatics	Kent State Helen Piontkivska	Fall semester 2007	3
Parallel and Distributed Computing	Univ. Cincinnati Fred Annexstein	Fall quarter 2007	3
Computational Physics 15PHYS410	Univ. Cincinnati Richard Gass	Spring 2008	3
Computational Thinking in Context CSE294 (Programming and Algorithms)	OSU, CSE P. Sadayyapan	Spring, 2008 Classroom only	4
Chem 644 Computational Chemistry	Chris Hadad OSU	Spring, 2008 Classroom only	3
Optimization	Wittenberg	Still pending	5
CPS3465 Parallel Programming	Robert Marcus Central State	Spring Semester 2008	3
Differential Equations Using Computation, MATH299	Columbus State	Summer quarter 2008	6

*All courses offered at distance unless otherwise indicated.

The full curriculum description can be found at:

http://www.rrscs.org/docs/competencyfinal.pdf

Regula School Initial Participating Institutions
Capital University
Central State University
Columbus State Community College
Kent State University
Ohio State University
Sinclair Community College
University of Cincinnati
Wittenberg University
Wright State University

Terms

- **Home Institution:** The institution where the student is admitted as a student. Home students "visit" other institutions through Regula School courses
- **Host Institution:** The institution where the Faculty of Record (primary course instructor) is employed. Host institutions "teach" the Regula course.
- **Participating Institutions:** Colleges and universities that elected to participate in sharing courses through the Regula School.
- **Faculty Coordinators:** At least one faculty member at each participating institution responsible for assisting students to register for courses, shepherding courses for approval by deans, department heads and curriculum committees, arranging course technical support, as needed, and communicating with their Home Registrar and the Regula School.
- **Registrar Contacts:** The staff in the Office of the Registrar of participating institutions who are responsible for sending and/or receiving Regula -related information to the Faculty Coordinators, the Host or Home Registrars Contacts, and the Regula School
- **Technical Contacts:** The staff at participating institutions responsible for providing various related technical support (video conferencing, student support services, etc.)
- **Visiting Students:** Students from institutions other than the Host Institution, who participate in Regula School courses.

REGULA SCHOOL START UP SUPPORT

To ensure that students have continual access to Regula School courses, the School will assist institutions with start-up costs. Each institution will provide the agreed upon course with the faculty-approved competencies for $3000 one time for an offering of the course. The courses and contracted institutions are listed above.

REGISTRATION PROCESSES

Regula School course information and schedule is made available through the Regula School website and the websites and print catalogs of all participating institutions (http://www.rrscs.org).

Once a course has been approved through its local curriculum review process and/or designated for a Regula course, that information will be posted at the host institution on their website, on-line catalog, and course management system (if available) and shared with the Regula School. If faculty approval is required by a student interested in registering for the Regula course, the Faculty Coordinator at each institution will secure approval for the student.

Supplementary Regula School course information will be available for student viewing on the Regula School web site (http://www.rrscs.org/minorcourses/index.shtml)

The Regula School will maintain an informational web site for students to gain knowledge about the course and curriculum, internships, and job possibilities. The Regula School will link to each participating institution's policies related to class calendars and dates for adds, drops, withdrawals, grade translations. This web site will stress that students register for courses through their Home Institution and not through this web site. Faculty Coordinators will provide Regula course information to the Regula School for publication on the RRSCS web site.

Students register and pay tuition for Regula course(s) at their Home Institution. To provide the information required by the registrars to track cross-registering students, students will be asked to fill out a common form for each course enrollment as a visiting student. The procedure used will be:

1. When a student fills out the special common form (Application for Host Institution Class Enrollment attached to this agreement) registrars at each participating home institution will add a specially tagged course number or a placeholder class on their schedule to enroll them in the class. A long-term goal of the consortium will be to cross-list all Regula School courses in the regular catalogs to facilitate registration and grade transfer procedures. The RRSCS website will have information about the course calendar for each such course and students will need to comply with the calendar at the host institution. The home institution will inform the student that, in cases where the home and host institutions differ in academic calendar systems (i.e., semester vs. quarter), the term associated with the home institution's placeholder course may not correspond to the actual term associated with the course taught at the host institution.
2. Students will register for the course at their home institution using regular registration procedures.
3. In addition, students will fill out the Application for Host Institution Class Enrollment, sign it, and turn it in at their home institution registrar's office. When faculty approval is required for registration, Faculty Coordinators will obtain approval for registration. Based on the local enrollment systems, each Regula School course will reserve one half of the enrollments in each class for the visiting students up to 30 days prior to the start of the class and after that time, class enrollment should become open to all students. To assist this process, registrars and Faculty Coordinators will meet online once per term.
4. Once the Application for Host Institution Class Enrollment has been received, the home registrars will fax the completed form to the host registrar who will, in turn, add the student to their local database and register the student for the hosted course, and notify the home institution of

completion of the registration process. This process will be facilitated electronically when those capabilities are in place through the Ohio Board of Regents.

5. The host institution will then notify the student that they are registered and transmit the procedures for them to obtain access to course materials and other host institution systems. Students will be given the same privileges as local students.   **As required, network IDs and Passwords are mailed to Regula students by the appropriate administrative office at the host university.**

6. Host/Teaching and Home Registrar Contacts will distribute course rosters. The Host/Teaching Registrar Contact will send the complete course roster (including visiting students and regular students) to the Faculty/Instructor of Record per usual systems. The Home Registrar Contact will send a course roster of only the visiting students from that institution to their local Faculty Coordinator. All participating Registrars will send an email to the Regula School that summarizes the institution’s final student enrollment numbers (undergraduate and graduate) in the shared course. This email to the Regula School should NOT include specific student information, only numbers of undergraduate and graduate students enrolled and their home institutions

7. **Drops/adds/withdrawals will be managed and communicated by Host/Teaching and Home Registrar Contacts.** Students will agree to abide by the calendar of the teaching institution in term of dates for withdrawal, penalties, and final grades. This information will be posted on the RRSCS website and the teaching institution site. Home Institution procedures and policies are followed regarding any administrative fees that are charged for add/drop activity, late enrollment processing, etc. Withdrawals should be made through both the home and host institutions. As a secondary measure, the home institution also will communicate the student’s withdrawal to the host institution’s registrar. Mechanisms to track drops will need to put into place to allow this information exchange.

GRADES AND TRANSCRIPTS

Grading Policies

Grade conversions will happen at the Home Institution. The Faculty of Record will submit all student grades (visiting and regular) to the Host/Teaching Registrar Contact using the grading scale of the Host/Teaching Institution.

Upon completion of the academic term, the host institution will notify the home institution of the final grade in the course and number of credit hours completed. The home institution will be responsible for all grade conversions and course grading scale adjustments. As these are completed, the home institution will provide a list of students and the courses they completed.

Grade changes will follow Home Institution policies and procedures

Local policies and procedures of the student’s Home Institution regarding late, missing, incorrect grades, and their impact on graduation will apply. In cases where graduations at Home Institutions are earlier than the Host/Teaching Institution, students will work with Faculty Coordinators to resolve any issues that might impact graduation.

STATE SHARE OF INSTRUCTION

State Share of Instruction will remain with the home institution.

Public colleges and universities offering Regula courses will report student enrollments through regular HEI reporting mechanisms. State Share of Instruction (SSI) will flow to the home institution by agreement. Independent non-for-profit institutions will not receive state support in the form of State Share of Instruction (SSI).

FUNDING MODELS

There are several aspects to the funding model under this collaborative agreement. To assist in the implementation of the program, the Ralph Regula School of Computational Science will offer a one-time subsidy of $3000 per course for each of the courses taught in the 2007-2008 academic year.

Each institution will collect tuition from the students taking courses taken through the Regula School. There will be no differences in course funding with native students. The home institution will collect the tuition and fees and state subsidy for those students.

Revenue will be shared with the Regula School and the institutions hosting visiting students. Tuition will be collected at the home institution for those students using their current tuition and fee structure. Payment will be made to the Regula School at a rate of $250 per quarter credit hour for each student hosted by other institutions. Public institutions will continue to collect the State Share of Instruction for those students.

The Regula School will use 25% of the collected fees for additional course development, marketing, and administration. The balance of the fees collected for hosted students will be passed to the hosting institution.

CURRICULUM APPROVAL AND INSTITUTIONAL INVOLVEMENT

All institutions participating in this consortium are expected to play an active role in on-going curriculum development, review, and delivery.

A) Each participating institution will offer at least one Regula School class at the institution in each academic year.
B) Each institution will appoint one faculty representative to the statewide curriculum review committee. That committee will meet quarterly (if needed) but at least twice per year to review and approve new courses for existing consortium members or proposals for courses by new member institutions. The committee also will review the competencies required in the minor program and revise them as necessary in response to changes in the science and engineering fields.
C) Member institutions can propose changes to the curriculum or to the consortium agreement. The changes will be considered by the statewide committee and a recommendation made to the consortium for their approval.

CURRICULUM EVALUATION

Curriculum and course reviews are the hallmarks of quality processes in higher education. Due to the nature of the content, courses in the Ralph Regula School of Computational Science have a shorter 'shelf live' than some other academic disciplines. A continuous improvement process is necessary for the strongest curriculum and the best trained students. The process shall be as follows:

A) The curriculum for the minor in computational science will be reviewed at the end of the first full year of all courses taught. A review team of faculty, students, and industry representatives will evaluate the relevance of the total curriculum and make recommendations for additions, deletions or revisions at the course level.

B) At the course level, courses and modules funded by RRSCS will be evaluated by a faculty and student peer review team on an annual basis after the first offering and then reviewed every other year The Ohio Learning Network's rubrics for online courses will be used as part of the evaluation process.

C) Courses taught by RRSCS faculty, but not funded by the School, will be examined as part of the overall curriculum review, and will be evaluated every other year by a faculty and student peer review team. This team shall contain some members from the review teams created in paragraph B above.

STUDENT SERVICES EVALUATION

An annual assessment will be made of student services related to the Ralph Regula School. The School will coordinate this effort with participating campuses to field regular surveys of students and campus administrative staff to ensure that students advising, registration, and other procedures are working satisfactorily from the student's viewpoint.

SIGNATORIES

The undersigned academic officers agree to the terms of this consortial agreement for the Ralph Regula School of Computational Science.

____
Ronald L. St. Pierre
Acting Provost
Capital University

____
Joseph A. Alutto
Interim Provost
The Ohio State University

____
Toy Caldwell-Colbert
Provost
Central State University

____
Anthony J. Perzigian
Provost
University of Cincinnati

____
Kay Adkins
Provost
Columbus State Community College

____
Kenneth W. Bladh
Provost
Wittenberg University

____
Robert G. Frank
Provost
Kent State University

____
Steven R. Angle
Provost
Wright State University

____
Helen Grove
Provost
Sinclair Community College

Ralph Regula School of Computational Science
Application for Host Institution Class Enrollment

Complete all items below. Sign and return the form to the registrar's office of your **home** institution. You will be registered for the class(es) at the host institution you have indicated below. You are responsible for observing all host institution registration, drop/add and withdrawal deadlines. **If enrolling in classes at more than one host institution, you must complete and submit a separate form for each**.

Name:____ **Social Security Number**: __ __ __/__ __ /__ __ __ __
Last First Middle

Other name(s) used previously: ____ **Date of Birth:** __ __/__ __/__________
Month Day Year

Permanent Mailing Address: ____
(Number and street. If P.O. Box, number and street also required)

____
City State/Country Zip

Telephone:_________________________ **E-mail Address**: ____
(include area code)

Gender: ☐ Female ☐ Male **Country of Citizenship**: ____

Check if this applies:: I am a ☐ Permanent resident alien of the U.S. **or** ☐ Refugee **or** ☐ Asylee

Alien/File # A ____ Date approved: ___________________________
Month Day Year

My most recent Visa type is: ______________________________ Date issued:____________________________
Month Day Year

Residency: Students enrolled in a public institution will retain their residency status as determined by their home institution. Students enrolled in a private institution will be reviewed for a residency determination.

Most recent high school attended: _______________________________________ **Graduation date**: __________________
Name of High school City State

Prior College Degree & Institution (if applicable): ____

Current Home Institution: ____
Institution for your primary registration and degree program

Class(es) in which you are enrolling:

1) _________________________ _________________________ _________________________ _________________________
Home Course Number and Class Section **Host** Course Number and Class Section **Host** Institution Term & Year **Host** Institution Credit hours

2) _________________________ _________________________ _________________________ _________________________
Home Course Number and Class Section **Host** Course Number and Class Section **Host** Institution Term & Year **Host** Institution Credit hours

I affirm that the information I have provided on this application is complete and accurate. Pursuant to the Family Educational Rights and Privacy Act of 1974, as amended (FERPA), I hereby authorize both my home institution and the host institution from which I am taking this course to exchange registration and required financial aid information regarding my enrollment in the class(es) noted on this form. I also authorize the host institution to release an official copy of my host institution transcript at the end of the term to my home institution. I understand my home institution will add the course(s) to my transcript and charge me the appropriate tuition and fees for the additional credit hours. I also agree to abide by the class schedule and course drop and/or withdrawal dates associated with the host institution from which I am taking this course.

____ ____________________
Applicant's Signature **Date**

To Be Completed by the Home Institution

Home institution residency status: ☐ In-state ☐ Out-of-state

I hereby certify that student named above is in academic good standing with this institution and is authorized to enroll in the host institution class(es) as indicated. Institution: ______________________________

Certifying Official:____ Title: ___________________________
Please Print

Telephone:_________________________ Fax: ____________________ E-mail Address: ______________________

Signature: ____ Date: ___________________________

Appendix B

OSU Course Offerings and Prerequisites

Course Number and Title	Prerequisites	Quarters offered	Credits	Topic
Math 151: Calculus and Analytic Geometry I	Mathematics 150 (with grade C- or better) or satisfactory score on Ohio State Math Placement Test	Autumn, Winter, Spring, Summer	5	Calculus
Math 152: Calculus and Analytic Geometry II	Mathematics 141 or 151	Autumn, Winter, Spring, Summer	5	
Math 153: Calculus and Analytic Geometry III	Mathematics 152	Autumn, Winter, Spring, Summer	5	
Math 161: Accelerated Calculus with Analytic Geometry I	Math 151 or permission of dept.	Autumn	5	
Math 162: Accelerated Calculus with Analytic Geometry II	Mathematics 161 or written permission of Math Counseling Office	Winter	5	
Math H190: Elementary Analysis I	Permission of dept.	Autumn	5	
Math H191: Elementary Analysis II	Mathematics H190 with a grade of C or better or written permission of Honors Committee chairperson	Winter	5	
ChBE 790 Process Modeling and Simulation	Permission of instructor	Autumn	3	Simulation & Modeling
CEG 640 Civil and Environmental Systems Engineering	Permission of instructor	Autumn	4	
CSE 778 Computer-Aided Design and Analysis of VLSI Circuits	CSE 560; ECE 561; 601; 675 or ECE 662	Autumn	4	
ISE 521 Operations Research I: Simulation of Production Systems	ISE 500; Stat 427 and 428, or equiv	Au, Wi	5	
ISE 704 Introduction to Discrete System Simulation	Stat 426 or 428	Winter	4	
ME 785 Modeling, Simulation and Control of Hybrid-Vehicles	ME 784 or permission of instructor	Winter (Offered even numbered Years)	4	
MSE 533 Modeling of Materials Processing Methods	MSE 525 and 526, Matsc&en major or permission of instructor	Spring	3	

Course Number and Title	Prerequisites	Quarters offered	Credits	Topic
CSE 202 Introduction to Programming and Algorithms for Engineers and Scientists	Math 151	Su, Au, Wi, Sp	4	Programming & Algorithms
CSE 221 Software Development Using Components	Math 151 or 161/H161 or H190; 201 or 202 or 203 or 204 or En Graph 167 or Engineer H192 or CS&E Placement Level A	Su, Au, Wi, Sp	4	
CSE 294P Computational Thinking in Context: Science and Engineering	Math 151 or equivalent	Sp	4	
EG 167 Problem Solving through Programming for Engineering Calculations and Computer Graphics	Math 151 or equivalent	Su, Au, Wi, Sp	4	
AAE 581 Numerical Methods in Aerospace Engineering	En Graph 167, 580	Autumn	3	Numerical Methods
CEG 406 Professional Aspects of Civil and Environmental Engineering	Civil en or Env Eng major; must be taken as soon as possible upon entering the major	Wi, Sp	1	
CSE 541 Elementary Numerical Methods	CSE 221/H221 or 230 or 502; Math 153	Su, Au, Wi, Sp	3	
ECE 715 Introduction to Numerical Methods for Electromagnetics	ECE 301, and Math 568 or 571; or grad standing	Autumn	3	
Math 606 Introduction to Numerical Analysis of Partial Differential Equations	Math 512 and 572 or equiv with permission of instructor	Sp	3	
Math 607 Essentials of Numerical Analysis	Math 548 or 652 or permission of the Graduate Studies Committee	Wi	5	
ME 250 Numerical Methods and Analysis in Mechanical Engineering	Math 415 or 255; and enrollment in engineering major	Au, Wi, Sp	4	
CBE 781 Chemical and Biomolecular Engineering Optimization	En Graph 167 or equiv or permission of instructor	Sp	3	Optimization
CEG 776 Network Algorithms in Transportation Systems	CEG 405, 540, and EG 167 or CSE 201 or 221 or equiv.	Wi	5	
ECE 759 Numerical Optimization for Electrical Engineers	ECE 352	Au	3	
ISE 522 Operations Research II: Fundamentals of Linear Optimization with Applications	ISE 500, Math 254, 415 or 255, and 568 or 571. Working knowledge of Excel	Au, Wi	3	
ME 761 Optimization in Mechanical Design	ME 562 or 563 or permission of instructor	Spring	3	
MSE 600 Materials Selection and Performance I	Sr standing in MatSc&Eng or Ceram Eng or Metal Eng or permission of the instructor	Wi	3	

Course Number and Title	Prerequisites	Quarters offered	Credits	Topic
CE 660 Civil Engineering Capstone Design	Sr standing. Must be taken as close to graduation as possible	Au, Wi, Sp	4	Capstone
CSE 699 Undergraduate Research in Computer Science and Engineering		Su, Au, Wi, Sp	1-5	
CSE H783 Honors Research	Honors standing; permission of instructor	Su, Au, Wi, Sp	1-5	
ME 564 Senior Design Group Project	ME 510; a second writing course and prereq or concur: 563	Au, Wi, Sp	3	
ME 565.01 Mechanical Engineering Design ME 565.02 Mechanical Engineering Design ME 565.03 Mechanical Engineering Design	ME 562 and 2nd writing course, 3.4 or higher GPA or permission of instructor ME 565.01 ME 565.02	Au Wi, Sp Au, Sp	3 3 3	
MSE 695.01 Senior Design Project I MSE 695.02 Senior Design Project I I MSE 695.03 Senior Design Project I I I	Sr standing in MSE or the physical sciences MSE 695.01 Sr standing in MatSc&En and MatSc&En 695.02	Au, Wi, Sp Au, Wi, Sp Au, Wi, Sp	1 1 1	
CE/ECE 675 Instrumentation, Signals, and Control in Transportation Applications	Elec Eng 301 and Math 415; or Civil En 570; or grad standing in elec eng or civil eng	Au	3	Discipline Specific Courses
CSE 630 Survey of Artificial Intelligence I: Basic Techniques	CSE 222/H222 or 230 or 502; Math 366	Au, Wi, Sp	3	
CSE 655 Introduction to the Principles of Programming Languages	CSE 560 and 625	Su, Au, Wi, Sp	4	
CSE 660 Introduction to Operating Systems	CSE 560; 675 or ECE 662; Stat 427	Su, Au, Wi, Sp	3	
CSE 670 Introduction to Database Systems I	CSE 314 or 222 or 230 or 502; Math 366	Su, Au, Wi, Sp	3	
CSE 675.01 Introduction to Computer Architecture CSE 675.02 Introduction to Computer Architecture	360 or ECE 265; Math 366; ECE 261 360 or ECE 265; Math 366	Au, Wi, Sp Su, Au, Sp	3 4	
CSE 680 Introduction to Analysis of Algorithms and Data Structures	CSE 560 or 668 or ECE 668; Stat 427; Math 566	Su, Au, Wi, Sp	3	
Chem 644 Computational Chemistry	Chem 252	Au	3	
MSE 756 Computational Materials Modeling	Permission of instructor	Au	3	
Phys 780				

Course Number and Title	Prerequisites	Quarters offered	Credits	Topic
CSE 621 Introduction to High-Performance Computing	CSE 541; Math 568 or Math 571 or Math 601. Course is well suited to grad students from science/engineering in addition to CS&E students	Au	3	Elective
CSE 694L Scientific Vizualization		Su, Au, Wi, Sp	1-5	
Math 255 Differential Equations and Their Applications	Math 254	Su, Au, Wi, Sp	5	
Math 415 Ordinary and Partial Differential Equations	Math 254	Su, Au, Wi, Sp	4	
Math 568 Introductory Linear Algebra	Math 254 or equiv with written permission of dept	Su Term 1, Au, Wi, Sp	3	
Math 571 Linear Algebra for Applications I	Math 254	Su Term 1, Au, Wi, Sp	3	

Attachment #5

Topic	Quarter Course	Semester Course	Notes
Calculus (Pre-Req)	Math 151	MATH 1151	
	Math 152	MATH 1152 or 1172	
	Math 153	MATH 1152 or 1172	
Simulation &		BIOMEDE 5430	FEM in BME (New)
Modeling		CHBE 5790	Model & Simulation (New)
	CEG 640		No Semester Equivalent
	CSE 778		No Semester Equivalent
	ISE 521	ISE 2010	
	ISE 704	ISE 5100	
	ME 785	MECHENG 5139	
	MSE 533	MATSCEN 2321	
Programming &	CSE 202	CSE 1222	
Algorithms	CSE 221	CSE 2221	
	CSE 294P	CSE 1221	
Numerical Methods	AAE 581	AEROENG 3581	
	CEG 406	CIVILEN 2060	
	CSE 541	CSE 5361	
	ECE 715	ECE 5510	
	Math 606		No Semester Equivalent
	Math 607	MATH 3607	
	ME 250	MECHENG 2850	
Optimization	CEG 776	CIVILEN 3080	
(no longer required	ECE 759	ECE 5759	
in Semester version	ISE 522	ISE 3200/3210/3990.3/5200	
but allowed as an elective	ME 761	MECHENG 7761	
	MSE 600	MATSCEN 4181	
Capstone Research	CE 660	CIVILEN 4000.01	
	CSE 699	CSE 4998	
	CSE H783	CSE 4999H	
	ME 564	MECHENG 4901.01	
	ME 565	MECHENG 4901.02	
	MSE 695	MATSCEN 4381	
Discipline-Specific		AEROENG 5615	Intro CFD (new)
Computationally		CHBE 5734	Mol Informatics (New)
Oriented Course	CSE 630	CSE 3521	
	CSE 655	CSE 3341	
	CSE 660	CSE 2431	
	CSE 670	CSE 3241	
	CSE 675	CSE 3421	
	CSE 680	CSE 2331	
		CSE 3461	Comp. Networking (New)
		CSE 3541	Comp. Gaming (New)
	Chem 644		New Course# Not Known
	MSE 756	MATSCEN 6756	
	Phys 780		New Course# Not Known
Elective	Math 255	MATH 2255	
	Math 415	MATH 2415	
	Math 568	Math 2568	
	Math 571	Math 2568	
	CSE 621	CSE 5441	
	CSE 694L	CSE 5544	