Computational Science Revision of the Undergraduate Major

The College at Brockport, College Senate

Follow this and additional works at: https://digitalcommons.brockport.edu/senate_resolutions

Part of the Higher Education Commons

Repository Citation
The College at Brockport, College Senate, "Computational Science Revision of the Undergraduate Major" (2008). College Senate Resolutions. 1370.
https://digitalcommons.brockport.edu/senate_resolutions/1370

This Resolution is brought to you for free and open access by Digital Commons @Brockport. It has been accepted for inclusion in College Senate Resolutions by an authorized administrator of Digital Commons @Brockport. For more information, please contact kmyers@brockport.edu, digitalcommons@brockport.edu.
TO: Dr. John R. Halstead, College President

FROM: The College Senate: March 24, 2008

RE:  
I. Formal Resolution (Act of Determination)
II. Recommendation (Urging the Fitness of)
III. Other, For Your Information (Notice, Request, Report, etc.)

SUBJ: Computational Science Revision of the Undergraduate Major
routing #68 06-07 UC

Signed: ________________________________
(P. Gibson Ralph, 2007-2008 College Senate President)

Date: ____________

Please fill out the bottom portion and follow the distribution instructions at the end of this page.

TO: P. Gibson Ralph, The College Senate President

FROM: John R. Halstead, College President

RE:  
I. Decision and Action Taken on Formal Resolution (circle choice)
   a. Accepted
      Implementation Effective Date: ____________
   b. Deferred for discussion with the Faculty Senate on ____________
   c. Unacceptable for the reasons contained in the attached explanation

II. III. Response to Recommendation or Other/FYI
   a. Received and acknowledged ____________

Signed: ________________________________
(Dr. John R. Halstead, President, SUNY College at Brockport)

Date: ____________

Comment: ____________________________________________

DISTRIBUTION

PRESIDENT'S OFFICE COPIES: Provost, Vice Presidents, College Senate, Other: ____________
PROVOST & VICE PRESIDENT(S) COPIES: Assistant Provost, Dean(s), Academic Advisement, Registrar,
Other: ____________________________________________

DEAN(S) COPY: Department Chair(s), Other: ____________________________________________

COLLEGE SENATE COPIES: Originator, College Senate Website, Other: ________________________
DEADLINE FOR SUBMISSIONS: FEBRUARY 28

INSTRUCTIONS – please, no multiple attachments – each proposal must be submitted as one document:
- Submit only complete proposals. Include support letters from department chair and dean.
- Proposals must be prepared individually in Word format using committee guidelines (guidelines online).
- Fill out this cover page for each proposal and insert it electronically as the front page of your document. (available online at www.brockport.edu/collegesenate)
- Email whole proposal with cover page as one attachment to senate@brockport.edu and facprez@brockport.edu .
- All updates must be resubmitted to the Senate office with the original cover page including routing number.
- Questions? Call the Senate office at 395-2586 or the appropriate committee chairperson.

1. PROPOSAL TITLE: Please be somewhat descriptive, ie. Graduate Probation/Dismissal Proposal rather than Graduate Proposal.

Revision of undergraduate Computational Science degree program

2. BRIEF DESCRIPTION OF PROPOSAL:
We wish to add courses to the core requirements in order to ease the transition between sophomore and junior/senior level courses and to teach several computational methods and programming concepts in greater depth. We also propose to add a mathematics course (differential equations) to the major requirements. Elective requirements will be reduced so as to not impact total credits.

3. ANTICIPATED EFFECTIVE DATE: Fall 2008

4. SUBMISSION & REVISION DATES: PLEASE PUT A DATE ON ALL UPDATED DOCUMENTS TO AVOID CONFUSION.

<table>
<thead>
<tr>
<th>First Submission</th>
<th>Updated on</th>
<th>Updated on</th>
<th>Updated on</th>
</tr>
</thead>
<tbody>
<tr>
<td>12/15/06 (rec’d 2/7/07 at Senate Office)</td>
<td>10/31/07</td>
<td>2/21/08</td>
<td></td>
</tr>
</tbody>
</table>

5. SUBMITTED BY: (contact person)

<table>
<thead>
<tr>
<th>Name</th>
<th>Department</th>
<th>Phone</th>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robert E. Tuzun</td>
<td>Computational Science</td>
<td>395-5368</td>
<td><a href="mailto:rtuzun@brockport.edu">rtuzun@brockport.edu</a></td>
</tr>
</tbody>
</table>

6. COMMITTEES TO COPY: (Senate office use only)

<table>
<thead>
<tr>
<th>Standing Committee</th>
<th>Forwarded To</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>__ Enrollment Planning &amp; Policies</td>
<td>To Committee for approval</td>
<td>2/7/07, 11/1/07, 2/21/08</td>
</tr>
<tr>
<td>__ Faculty &amp; Professional Staff Policies</td>
<td>Committee Chair Sign Here When Passed</td>
<td></td>
</tr>
<tr>
<td>__ General Education &amp; Curriculum Policies *</td>
<td>To Executive Committee</td>
<td>2/25/08</td>
</tr>
<tr>
<td>__ Graduate Curriculum &amp; Policies</td>
<td>GED to Vice Provost</td>
<td>NA</td>
</tr>
<tr>
<td>__ Student Policies</td>
<td>To Senate</td>
<td>3/3/08, vote 3/24/08</td>
</tr>
<tr>
<td>__ Undergraduate Curriculum &amp; Policies</td>
<td>To College President</td>
<td>4/8/08</td>
</tr>
</tbody>
</table>

* follow special Gen Ed procedures for submission of General Education proposals at “How to Submit Proposals” on our Website.

**Use routing number and title in all reference to this proposal.

COMMITTEE CHAIR:
- WHEN YOUR COMMITTEE APPROVES A PROPOSAL, PLEASE SIGN WHERE INDICATED ABOVE. GENERAL EDUCATION PROPOSALS MUST ALSO BE SIGNED ON THE SIGNATURE PAGE PROVIDED IN THE PACKET. YOUR SIGNATURE WILL BE UNDER THE DEAN’S SIGNATURE. RETURN ALL PROPOSALS TO SENATE OFFICE FOR PROCESSING.
2. Side by side comparison of the old and new core curriculum in Computational Science.

<table>
<thead>
<tr>
<th>Current Prerequisites</th>
<th>Cr</th>
<th>Proposed Prerequisites</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTH201 (Calculus I)</td>
<td>4</td>
<td>MTH201 (Calculus I)</td>
<td>4</td>
</tr>
<tr>
<td>MTH202 (Calculus II)</td>
<td>4</td>
<td>MTH202 (Calculus II)</td>
<td>4</td>
</tr>
<tr>
<td>MTH281 (Discrete Math I)</td>
<td>3</td>
<td>MTH281 (Discrete Math I)</td>
<td>3</td>
</tr>
<tr>
<td>CSC120 (Intro. Computer Science)</td>
<td>3</td>
<td>CSC120 (Intro. Computer Science)</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Current Core</th>
<th>Proposed Core</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPS201 (Computational Tools I)</td>
<td>3</td>
</tr>
<tr>
<td>CPS202 (Computational Tools II)</td>
<td>3</td>
</tr>
<tr>
<td>CPS303 (High Performance Computing)</td>
<td>3</td>
</tr>
<tr>
<td>CPS304 (Simulation and Modeling)</td>
<td>3</td>
</tr>
<tr>
<td>CPS404 (Applied and Computational Mathematics)</td>
<td>3</td>
</tr>
<tr>
<td>MTH203 (Calculus III)</td>
<td>4</td>
</tr>
<tr>
<td>MTH255 (Differential Equations)</td>
<td>3</td>
</tr>
<tr>
<td>MTH324 (Linear Algebra)</td>
<td>3</td>
</tr>
<tr>
<td>MTH243 (Elementary Statistics) or MTH346 (Probability and Statistics I)</td>
<td>3</td>
</tr>
<tr>
<td>MTH346 (Probability and Statistics I)</td>
<td>3</td>
</tr>
<tr>
<td>CSC203 (Fundamentals of Computer Science I)</td>
<td>4</td>
</tr>
<tr>
<td>200-level and higher non-CPS electives from an area of application chosen under advisement</td>
<td>6</td>
</tr>
<tr>
<td>Upper division electives</td>
<td>12</td>
</tr>
<tr>
<td>200-level and higher non-CPS electives from an area of application chosen under advisement</td>
<td>6</td>
</tr>
<tr>
<td>Total Credits</td>
<td>47</td>
</tr>
</tbody>
</table>

3. Brief rationale for changes.

The core of the current computational science major consists of five courses, and it is extremely difficult to fit in all the material we wish to. The transition between sophomore and junior/senior courses is particularly steep. Now that we have an additional faculty member, we wish to add a beginning junior level scientific computing course to ease this transition, and to extend CPS404 (Applied and Computational Mathematics) to a two semester sequence. This will enable more complete coverage of a variety of computational methods and programming concepts. In addition, we wish to add Scientific Visualization, currently an elective, to the required core, since visualization is a key skill in scientific computing. Finally, we wish to change some of the math requirements. We will compensate for the required credit increase by reducing the elective credits.

The proposed curriculum will take effect in Fall 2008. Students who entered under the current
4. Description of new courses (see also attached course registration forms).
   a. CPS333 (Scientific Computing)
      A survey of programming methods and the use of UNIX for practical scientific computation: (1) Advanced features of Fortran 90/95 such as modules and operator and function overloading, and practical strategies for their use; (2) Bookkeeping; (3) Shell scripting and other methods for automating scientific computations; (4) Numerical linear algebra libraries (BLAS and LAPACK) and their use, and construction of libraries; (5) Advanced features of MATLAB and other scientific computing packages.
   b. CPS404 (Applied and Computational Mathematics I) (course revision only)
      The first in a two-semester sequence. A survey of scientific computing methods, emphasizing programming methods, interpretation of numerical results, and checks for sensibility and self-consistency. This course is divided into several modules, including (1) representation of floating point data, truncation and rounding error, and basic considerations for accurate numerical computing; (2) iterative numerical methods; (3) numerical differentiation and integration; (4) interpolation methods; (5) random number generation. Extensive programming required.
   c. CPS405 (Applied and Computational Mathematics II)
      A continuation of CPS404. Topics include: (1) iterative methods in numerical linear algebra; (2) least squares methods; (3) the Fast Fourier Transform; (4) special functions and their computation; (5) numerical solution of ordinary differential equations. Extensive programming required.

These course registrations have been approved by the Dean of Letters and Sciences and by Registration.

5. Staffing Issues
   a. With the addition of a new faculty member in the fall of 2007, the department will be able to accommodate the new courses CPS333 and CPS405.
   b. CPS533 (Scientific Visualization) is an existing course that is taught every year. Making this a swing course and allowing in undergraduates will not impact staffing in any way.

6. Academic administration commentary.
   a. Letter of recommendation from the chair—the chair of CPS drafted this document. Therefore, he approves its contents, and a separate letter is not needed.
   b. Approval from Dean Appelle, School of Letters and Sciences, follows.

7. Resources and facilities that may be needed to implement the program.
   None.

8. Other letters of support.
   Letter of endorsement from Mathematics department follows on the next page.
I support the attached proposal for a new track in Computational Science.

Stuart Appelle, Ph.D.
Dean, School of Letters and Sciences
State University of New York, College at Brockport
Brockport, New York 14420

October 25, 2007

Robert Tuzun, Chair
Department of Computational Science
SUNY Brockport

Dear Dr. Tuzun,

The faculty members of our department discussed your proposal regarding the “Revision of the Undergraduate Major in Computational Science” during our Oct. 18th 2007 department meeting.

I am pleased to inform you that you have the full support of the Department of Mathematics and we are ready to accommodate the students that will choose to register for our courses.

Good luck with your new track.

Sincerely,

Mihail Barbosu, Chair
Department of Mathematics
SUNY Brockport
Action concerns a new course  Discipline CPS Number 333
Action concerns an existing course: Discipline  Number

Course Submitted by: Robert E. Tuzun
Department: Computational Science

Chairperson’s Approval: ___________________________ Date:

Dean’s Approval: ___________________________ Date:

1. Action requested:
   ☑ Registration of new course
   ☐ Revision of content for existing course
   ☐ Registration of general course listing under which related titles can be taught (umbrella courses)
   ☐ Registration of topics course for specific semester (if checked, complete item 2 below)
   ☐ Change of course title – Previous Title:
   ☐ Inactivation of existing course registration (course will not be taught in near future)
   ☐ Other – Describe:

2. Complete for registration of topics courses or umbrella courses:
   a. General course registration: Discipline:  Number:
      General registration title:
      Specific course title for semester offered (topics course title):
      Topics course registration is for: Year:
   
3. a. ☑ Undergraduate listing: Discipline: CPS Number: 333
   b. ☐ Graduate listing: Discipline:  Number:

   b. Course start date: FALL SEMESTER YEAR: 2008

5. Abbreviated course title (restricted to 16 spaces) SCIENTIFIC COMP

6. a. Semester hours of credit assigned to course (invariable): 3
   b. Can this course have variable credit ☑ No ☐ Yes - Credit range to semester hours
   c. Is this course is repeatable for multiple credit? ☑ No ☐ Yes - Credit Maximum =

7. Type of Course: LIBERAL ARTS

8. General Education Information: (Complete only for General Education courses)
a. General Education Knowledge Area (choose one if applicable): NONE
b. Additional student learning outcomes: (check all codes that are currently approved)
   - Contemporary Issues (I)
   - Upper Level Writing (U)
   - Both Contemporary Issues and Upper Level Writing (J)
   - Scholarship on Women (W)
   - Diversity (D)
   - Science & Technology (E)
   - Other World Civilizations (Non-Western) (O)

9. If cross-listed in another discipline(s), give discipline(s)/number(s):
   If there are pre-requisites that are enforced, give discipline(s)/number(s): CPS202
   If there are co-requisites, give discipline(s)/number(s):

10. a. Approximate total number of seats/semester expected: 8
    b. How many sections do you expect to offer per semester: 1

11. Sections of this course are (check one): taught by one instructor ☒ taught by a team ☐

12. Planned frequency of offering: EVERY FALL

13. Grading (check any that apply):
   - ☒ Letter grade ☐ Pass/Fail (S/U) ONLY ☐ Approved for IP grade
   - ☐ Course requires a minimum grade of ☐ for General Education or the major

14. If this course requires any special scheduling arrangements with regard to time or room/space, please comment on this in the space provided:

15. If this course is required for any degrees/programs, please list them below:
    BS, CPS

16. Write a brief course description for the College Catalogs. Reflect content as accurately as possible using 65 words or less (about 500 characters. Use action verbs and omit "This course covers..." and similar phrases.

   A survey of programming methods and the use of UNIX for practical scientific computation: (1) Advanced features of Fortran 90/95 such as modules and operator and function overloading, and practical strategies for their use; (2) Bookkeeping; (3) Shell scripting and other methods for automating scientific computations; (4) Numerical linear algebra libraries (BLAS and LAPACK) and their use, and construction of libraries; (5) Advanced features of MATLAB and other scientific computing packages.

17. For all courses, please attach the following information:
   - (a) a list of major course objectives
   - (b) a topical outline of course
   - (c) a list of methods used to evaluate student performance
   - (d) a list of instructional materials used – give bibliographic citations of texts, critical readings, films, e
   - (e) a current course syllabus, if possible
   - (f) a brief statement detailing the additional work required of graduate students in a “swing course.”

   For General Education courses only, attach also:
   - (g) Supplemental General Education Course Registration Form
(h) Student Learning Outcomes Checklist (for specific codes requested).

(a) List of course objectives and (b) Topical outline of course

See question (16).

(c) List of methods used to evaluate student performance

10-12 homework assignments, equally weighted.
1 project.

(d) List of instructional materials used

Handouts and Web readings

(e) Course syllabus

Not yet available

(f) Additional work required of graduate students in a swing course

N/A
College Course Registration Form

This form is used to register all courses. It must be signed by the department chair and the school dean who will send it forward to the Registrar’s Office. Registration of General Education courses requires an additional Supplemental Course Registration Form and the appropriate Student Learning Outcomes Checklist. Approval by the Faculty Senate’s General Education Committee is necessary for all General Education courses.

Data entry fields are shaded. Some fields have limits on numbers and spaces that can be entered. Some areas have drop-down menus with options that can be selected by clicking your choice. Save the blank form before using it and then save each course form with an individual file name. Go back to the saved blank for each new form but save with a different name after filling in data.

<table>
<thead>
<tr>
<th>Action concerns a new course</th>
<th>Discipline</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action concerns an existing course</td>
<td>Discipline CPS</td>
<td>Number 404</td>
</tr>
</tbody>
</table>

Course Submitted by: Robert E. Tuzun
Department: Computational Science

Chairperson’s Approval: __________________________ Date:
Dean’s Approval: __________________________ Date:

1. Action requested:
   - [ ] Registration of new course
   - [ ] Revision of content for existing course
   - [ ] Registration of general course listing under which related titles can be taught (umbrella courses)
   - [ ] Registration of topics course for specific semester (if checked, complete item 2 below)
   - [ ] Change of course title – Previous Title: Applied and Computational Mathematics
   - Inactivation of existing course registration (course will not be taught in near future)
   - [ ] Other – Describe:

2. Complete for registration of topics courses or umbrella courses:
   a. General course registration: Discipline: Number:
   b. General registration title:
   c. Specific course title for semester offered (topics course title):
   d. Topics course registration is for: Year:

3. a. Undergraduate listing: Discipline: CPS Number: 404
   b. Graduate listing: Discipline: CPS Number: 504

   b. Course start date: FALL  SEMESTER  YEAR: 2008

5. Abbreviated course title (restricted to 16 spaces) APPL COMP MATH I

6. a. Semester hours of credit assigned to course (invariable): 3
   b. Can this course have variable credit
   - [ ] No
   - [ ] Yes - Credit range to semester hours
   c. Is this course is repeatable for multiple credit?
   - [ ] No
   - [ ] Yes - Credit Maximum =

7. Type of Course: LIBERAL ARTS

8. General Education Information: (Complete only for General Education courses)
a. General Education Knowledge Area (choose one if applicable): NONE

b. Additional student learning outcomes: (check all codes that are currently approved)
   - Contemporary Issues (I)
   - Upper Level Writing (U)
   - Both Contemporary Issues and Upper Level Writing (J)
   - Scholarship on Women (W)
   - Diversity (D)
   - Science & Technology (E)
   - Other World Civilizations (Non-Western) (O)

9. If cross-listed in another discipline(s), give discipline(s)/number(s):
   If there are pre-requisites that are enforced, give discipline(s)/number(s): CPS304, 333; MTH203, 255; MTH243 or 346
   If there are co-requisites, give discipline(s)/number(s):

10. a. Approximate total number of seats/semester expected: 8
    b. How many sections do you expect to offer per semester: 1

11. Sections of this course are (check one): taught by one instructor ☒ taught by a team ☐

12. Planned frequency of offering: EVERY FALL

13. Grading (check any that apply):
    - Letter grade ☒
    - Pass/Fail (S/U) ONLY ☐
    - Approved for IP grade ☐
    - Course requires a minimum grade of ☐

14. If this course requires any special scheduling arrangements with regard to time or room/space, please comment on this in the space provided:

15. If this course is required for any degrees/programs, please list them below:
    BS, CPS

16. Write a brief course description for the College Catalogs. Reflect content as accurately as possible using 65 words or less (about 500 characters. Use action verbs and omit “This course covers...” and similar phrases.

   The first in a two-semester sequence. A survey of scientific computing methods, emphasizing programming methods, interpretation of numerical results, and checks for sensibility and self-consistency. This course is divided into several modules, including (1) representation of floating point data, truncation and rounding error, and basic considerations for accurate numerical computing; (2) iterative numerical methods; (3) numerical differentiation and integration; (4) interpolation methods; (5) random number generation. Extensive programming required.

17. For all courses, please attach the following information:
   (q) a list of major course objectives
   (b) a topical outline of course
   (c) a list of methods used to evaluate student performance
   (d) a list of instructional materials used – give bibliographic citations of texts, critical readings, films, e
   (e) a current course syllabus, if possible
   (f) a brief statement detailing the additional work required of graduate students in a “swing course.”
For General Education courses only, attach also:

(g) Supplemental General Education Course Registration Form
(h) Student Learning Outcomes Checklist (for specific codes requested).

(a) List of course objectives and (b) Topical outline of course

See question (16).

(c) List of methods used to evaluate student performance

10-12 homework assignments, equally weighted.
1 project.

(d) List of instructional materials used

Handouts and Web readings

(e) Course syllabus

See attached

(f) Additional work required of graduate students in a swing course

Additional homework problems and 1 additional project.
College Course Registration Form

This form is used to register all courses. It must be signed by the department chair and the school dean who will send it forward to the Registrar’s Office. Registration of General Education courses requires an additional Supplemental Course Registration Form and the appropriate Student Learning Outcomes Checklist. Approval by the Faculty Senate’s General Education Committee is necessary for all General Education courses.

Data entry fields are shaded. Some fields have limits on numbers and spaces that can be entered. Some areas have drop-down menus with options that can be selected by clicking your choice. Save the blank form before using it and then save each course form with an individual file name. Go back to the saved blank for each new form but save with a different name after filling in data.

---

Action concerns a new course
Discipline: CPS
Number: 405

Action concerns an existing course:
Discipline: 
Number: 

---

Course Submitted by: Robert E. Tuzun
Department: Computational Science

Chairperson’s Approval: ____________________________ Date: 

Dean’s Approval: ____________________________ Date: 

---

1. Action requested:

- [x] Registration of new course
- [ ] Revision of content for existing course
- [ ] Registration of general course listing under which related titles can be taught (umbrella courses)
- [ ] Registration of topics course for specific semester (if checked, complete item 2 below)
- [ ] Change of course title – Previous Title:
- [ ] Inactivation of existing course registration (course will not be taught in near future)
- [ ] Other – Describe:

2. Complete for registration of topics courses or umbrella courses:

a. General course registration: Discipline: __________ Number:

b. General registration title:

c. Specific course title for semester offered (topics course title):

d. Topics course registration is for: Year:

3. a. [x] Undergraduate listing: Discipline: CPS Number: 405

b. [x] Graduate listing: Discipline: CPS Number: 505

4. a. [ ] Official course title: Applied and Computational Mathematics II

b. [ ] Course start date: SPRING SEMESTER YEAR: 2008

5. Abbreviated course title (restricted to 16 spaces) APPL COMP MTH II

6. a. [ ] Semester hours of credit assigned to course (invariable): 3

   b. Can this course have variable credit [x] No [ ] Yes - Credit range to semester hours

   c. Is this course is repeatable for multiple credit? [x] No [ ] Yes - Credit Maximum =

7. Type of Course: LIBERAL ARTS
8. General Education Information: (Complete only for General Education courses)
   a. General Education Knowledge Area (choose one if applicable):  NONE

   b. Additional student learning outcomes: (check all codes that are currently approved)
      ☐ Contemporary Issues (I)          ☐ Upper Level Writing (U)
      ☐ Both Contemporary Issues and Upper Level Writing (J)
      ☐ Scholarship on Women (W)         ☐ Diversity (D)
      ☐ Science & Technology (E)         ☐ Other World Civilizations (Non-Western) (O)

9. If cross-listed in another discipline(s), give discipline(s)/number(s):
   If there are pre-requisites that are enforced, give discipline(s)/number(s): CPS404 or 504
   If there are co-requisites, give discipline(s)/number(s):

10. a. Approximate total number of seats/semester expected: 8
    b. How many sections do you expect to offer per semester: 1

11. Sections of this course are (check one): taught by one instructor ☒ taught by a team ☐

12. Planned frequency of offering: EVERY SPRING

13. Grading (check any that apply):
    ☒ Letter grade       ☐ Pass/Fail (S/U) ONLY       ☐ Approved for IP grade
    ☐ Course requires a minimum grade of    for General Education or the major

14. If this course requires any special scheduling arrangements with regard to time or room/space, please
    comment on this in the space provided:

15. If this course is required for any degrees/programs, please list them below:
    BS, CPS

16. Write a brief course description for the College Catalogs. Reflect content as accurately as possible using 65 words
    or less (about 500 characters. Use action verbs and omit “This course covers...” and similar phrases.
    A continuation of Applied and Computational Mathematics I. Topics include: (1) iterative methods in numerical linear
    algebra; (2) least squares methods; (3) the Fast Fourier Transform; (4) special functions and their computation; (5)
    numerical solution of ordinary differential equations. Extensive programming required.

17. For all courses, please attach the following information:
    (a) a list of major course objectives
    (b) a topical outline of course
    (c) a list of methods used to evaluate student performance
    (d) a list of instructional materials used – give bibliographic citations of texts, critical readings, films, e
    (e) a current course syllabus, if possible
    (f) a brief statement detailing the additional work required of graduate students in a “swing course.”
    For General Education courses only, attach also:
    (g) Supplemental General Education Course Registration Form
(h) *Student Learning Outcomes Checklist (for specific codes requested).*

(a) List of course objectives and (b) Topical outline of course

See question (16).

(c) List of methods used to evaluate student performance

10-12 homework assignments, equally weighted.
1 project.

(d) List of instructional materials used

Handouts and Web readings

(e) Course syllabus

See attached

(f) Additional work required of graduate students in a swing course

Additional homework problems and 1 additional project.
Instructor:  Dr. Robert E. Tuzun  
128 Smith Hall
Office/FAX:  395-5368/5020
Office hours:  Wed 1:00 – 3:00 PM  
Thurs 1:00 – 3:00 PM
and by appointment
email:  rtuzun@brockport.edu

My Web page:  http://www.cps.brockport.edu/~tuzun
Course Web page:  http://www.cps.brockport.edu/~tuzun/courses/cps404/cps404.html

Course materials
  Supplemental handouts, including material from the Web and other sources.

Auxiliary texts:

Course description
  A survey of computational techniques commonly used in scientific and engineering calculations and an introduction to software libraries and to the vast body of literature on scientific computing. The coverage is meant to be broad but not overly deep. A major goal of this course is to enable the student to pick suitable techniques for his or her applications, to gather further information if required, and to be able to tell when numerical results make sense.

  This course is organized into 5 two- to three-week modules:

(1) Basic considerations of scientific computing and floating point computation—IEEE floating point representation, general strategies for accurate computation, types of error (truncation vs. rounding, absolute and relative, forward and backward), and sensitivity.

(2) Interpolation and approximation—interpolation vs. approximation, interpolating functions (monomial, Newton, Lagrange), piecewise interpolation (i.e., splines).

(3) Differentiation and integration—schemes for constructing numerical methods using Taylor series methods or Lagrange interpolation, Richardson extrapolation, Romberg integration, programming methods.

(4) Iterative methods—applications to root finding (fixed-point, Newton-Raphson, bisection, etc.), programming methods.

(5) Random number generation—generation of uniform distributions, statistical tests, transformation from uniform to non-uniform distributions, applications.

Homework
  Homework assignments will be given approximately once every week, including one assignment due during finals week. These will include

- Paper and pencil problems.
- Programming—modification of code provided to you, or writing some simpler codes from scratch. Concepts to be developed include generality, robustness, accuracy, and efficiency, as well as bookkeeping issues.

You may work in Fortran 90 or in C/C++, whichever is comfortable. For those of you who know both Fortran and C/C++, I would suggest using both throughout the course.
Homeworks will consist of short written presentations that you hand in, as well as any accompanying files on your computer account. Projects will require longer presentations and more elaborate calculations; specific guidelines will handed out with each project. Please follow the following guidelines.

- For any question that calls for interpretation of numerical results, please provide succinct written presentation. It is not enough to just provide numerical results; you must demonstrate that you understand the results.
- You may sometimes want to show raw data or diagnostic output in order to make some point. In your written presentations, please show snippets as short as possible, since the full output may consist of thousands of lines.
- On each graph you generate, be sure to include axis labels with units, as well as a title.
- Maintain a separate directory for this course. Organize your directory space so that each homework assignment or project has its own directory. You may sometimes find it advantageous to make subdirectories.
- In order for me to be able to provide the best feedback possible, I will want to be able to follow and to reproduce your work (including all numerical results). Please be sure to save all the source code, shell scripts, etc., required to generate your data, as well as output files. If you have a large number of files, please provide a table of input and output files (with their most relevant run parameters) and names of source codes, shell scripts, and any other relevant files.
- Please adhere to coding style guidelines shown later in this syllabus.

Computing environment
Each student will be given an account on the Computational Science department's Silicon Graphics Onyx 2 workstation. This computer has a UNIX operating environment; Fortran and C/C++ compilers; and several graphical and general purpose packages such as MATLAB.

Projects
Instead of tests, there will be one (for CPS404) or two (for CPS504) approximately three-week projects. The purpose of these projects will be to explore in greater detail than a homework assignment a problem related to the class material. Fully detailed handouts will be provided later.

The Web
The course Web page will be updated frequently. In addition to being passed out in the classroom, assignments and handouts will be posted on the course Web page. Answer keys, class announcements, and so on will also be posted. No grades will be accessible from the Web.

Grading
Projects: 20% (CPS404) or 40% (CPS504), weighted equally.
Homework: 80% (CPS404) or 60% (CPS504), weighted equally.
There will be no tests or final exam.

Late homework/projects will penalized 25% per day.

Writing/presentation skills and coding style and readability will count toward the grades. Your coding style does not have to be the same as mine; however, your code does need to be readable and to have a consistent style.

I am a great believer in partial credit. I can’t give credit for a one-line wrong answer, but if you show your work, you will receive partial credit for what you did correctly. (You will also receive higher quality, more targeted feedback).
Final grades will be assigned according to the following cutoffs. The cutoffs may be lowered under some circumstances, but they will never be raised.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Average, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>93.0 &amp; above</td>
</tr>
<tr>
<td>A-</td>
<td>90.0-92.9</td>
</tr>
<tr>
<td>B+</td>
<td>87.0-89.9</td>
</tr>
<tr>
<td>B</td>
<td>83.0-86.9</td>
</tr>
<tr>
<td>B-</td>
<td>80.0-82.9</td>
</tr>
<tr>
<td>C+</td>
<td>77.0-79.9</td>
</tr>
<tr>
<td>C</td>
<td>73.0-76.9</td>
</tr>
<tr>
<td>C-</td>
<td>70.0-72.9</td>
</tr>
<tr>
<td>D+*</td>
<td>67.0-69.9</td>
</tr>
<tr>
<td>D*</td>
<td>63.0-66.9</td>
</tr>
<tr>
<td>D-</td>
<td>60.0-62.9</td>
</tr>
<tr>
<td>E</td>
<td>Below 60.0</td>
</tr>
</tbody>
</table>

*These grades are not allowed for graduate level courses.

Coding style guidelines

In order to make it as easy as possible to understand and grade your work, I am gradually introducing a set of coding guidelines to be used in all of my upper level courses. At a minimum, please adhere to the following guidelines:

1. Begin your program with a comment block that includes your name, the homework problem or project number, and any other relevant identifying information, as well as any other comments you wish to include.
2. Separate your code into paragraphs interspersed with comment blocks. Paragraphs and/or comment blocks should be separated from one another by one empty line.
3. Indent in a way that enhances readability:
   a. Indent each for, do, while, or do-while statement block by some number of spaces (at least two).
   b. Indent each if block by some number of spaces (at least two).
   c. Do not use TAB characters to indent.
   d. In C/C++ programs, it does not matter to me whether braces off-setting statement blocks are on their own separate lines or not. However you choose, be consistent throughout the program.
4. In Fortran programs, use implicit none in all main programs, functions, and subroutines. Never rely on the Fortran naming convention.
5. Name variables, and other identifiers such as function names, in a manner that makes it as easy as possible to determine their scope and purpose. Here is a sampling of common practices, meant to inspire you to develop your own style:
   a. Local loop variables are named i, j, k, ...
   b. Variables appearing within only a small section of code begin with lower case letters (iStart, iEnd).
   c. Variables that are larger in scope, or other identifiers such as subroutine names, consist of short capitalized words strung together (e.g., SumX for sum of X, SumXX for sum of squares of X, NumIterMax for maximum number of iterations). An alternative is to use lower case and to separate words with underscores (e.g., sum_x, sum_xx).
   d. Parameters (in Fortran) or defined constants (in C/C++) have all upper case letters, with words separated by underscores (e.g., MAX_SIZE).
   e. Variable names are no longer than 15 characters.

An excellent book on all aspects of software construction is Mike McConnell's Code Complete (Microsoft Press). Although the book is aimed at business programmers, it is full of advice useful to all programmers.
General policies

Course policies are in accordance with “SUNY Brockport 2007-2008: Your Right to Know & Academic Policies Handbook”.

Students are expected to attend all classes. Absences will be excused for (a) documented illness, (b) official representation of the college, (c) death of a close relative, (d) religious holidays, and (e) other circumstances beyond the control of the student. The student is responsible for obtaining class notes and handouts for classes missed. Missing classes does not absolve the student of responsibility for any course work. In accordance with suggested College policy, any student whose unexcused absences exceed 15% of classes may receive a lowered grade or failure at the instructor's discretion.

No incompletes or withdrawals will be given for this course except in cases of illness, personal tragedy, or extraordinary circumstances beyond the student's control, and then only if documented to the instructor's satisfaction.

While you are encouraged to work together, any assignments handed in must be your own work. Submitting material as your own work that is not entirely your authorship or knowingly providing an answer to another person constitutes cheating. If I am convinced beyond a reasonable doubt that cheating has occurred, the person(s) submitting the illicitly obtained answer, and any person(s) knowingly providing the same, will receive a grade of zero for that assignment. Further disciplinary procedures may also be considered.

SUNY Brockport is committed to maintaining a workplace and learning environment free of sexual harassment and intimidation. Sexual harassment is unacceptable behavior, unlawful, and intolerable.

I do not have the authority to cancel classes, no matter how severe the weather. Only the President of the College makes that decision. On days of severe weather, please listen to WBSU (89.9 FM) or WHAM (1180 AM) for class cancellations. You may also call 395-COLD. If I have any information, I will do my best to send email or post a message on my Web page.

Disability statement

Students with documented disabilities may be entitled to specific accommodations. SUNY Brockport's Office for Students with Disabilities makes this determination. Please contact the Office for Students with Disabilities at 395-5409 or osdoffic@brockport.edu to inquire about obtaining an official letter to the course instructor detailing approved accommodations. The student is responsible for providing the course instructor with the official letter. Faculty and staff work as a team with the Office for Students with Disabilities to meet the needs of students with disabilities.

Disclaimer

Announcements given in class or on the Web are considered official addenda to course policies. All policies described are subject to change as the situation warrants. Should changes be necessary, all students will be notified in a timely fashion.
Course description

A continuation of CPS404/504. Special emphasis is placed on testing/verification of software. This course is organized into 5 two- to three-week modules:

6. **Iterative methods in numerical linear algebra**—Methods for approximate solutions of systems of linear equations, and for computation of eigenvalues/eigenvectors. Issues related to convergence are extensively discussed. Methods applicable to special matrix types (symmetric, banded, and so on) are considered.

7. **Least squares problems**—Linear least squares methods, and some non-linear least squares methods. Fundamentals, constrained least squares problems, and sensitivity are thoroughly discussed.

8. **Fast Fourier transform (FFT)**—Time and frequency domains, applications, different types of Fourier transforms (for continuous, periodic, and discrete functions), the fast Fourier transform, limitations and caveats. Only one-dimensional problems are considered here.

9. **Special functions and their computation**—General theory of orthogonal functions, and properties of and methods for computation of several special function types (Bessel, Legendre, Chebyshev).

10. **Numerical solution of ordinary differential equations**—Trial solution for different types of differential equations, applications, specification of boundary value and initial value problems, forward Euler and other simple methods for numerical solution of initial value problems.

Homework

Homework assignments will be given approximately once every week, including one assignment due during finals week. These will include:

- Paper and pencil problems.
- Programming—modification of code provided to you, or writing some simpler codes from scratch. Concepts to be developed include generality, robustness, accuracy, and efficiency, as well as bookkeeping issues.

You may work in Fortran 90 or in C/C++, whichever is comfortable. For those of you who know both Fortran and C/C++, I would suggest using both throughout the course.
Homeworks will consist of short written presentations that you hand in, as well as any accompanying files on your computer account. Projects will require longer presentations and more elaborate calculations; specific guidelines will handed out with each project. Please follow the following guidelines.

- For any question that calls for interpretation of numerical results, please provide succinct written presentation. It is not enough to just provide numerical results; you must demonstrate that you understand the results.
- You may sometimes want to show raw data or diagnostic output in order to make some point. In your written presentations, please show snippets as short as possible, since the full output may consist of thousands of lines.
- On each graph you generate, be sure to include axis labels with units, as well as a title.
- Maintain a separate directory for this course. Organize your directory space so that each homework assignment or project has its own directory. You may sometimes find it advantageous to make subdirectories.
- In order for me to be able to provide the best feedback possible, I will want to be able to follow and to reproduce your work (including all numerical results). Please be sure to save all the source code, shell scripts, etc., required to generate your data, as well as output files. If you have a large number of files, please provide a table of input and output files (with their most relevant run parameters) and names of source codes, shell scripts, and any other relevant files.
- Please adhere to coding style guidelines shown later in this syllabus.

Computing environment
Each student will be given an account on the Computational Science department's Silicon Graphics Onyx 2 workstation. This computer has a UNIX operating environment; Fortran and C/C++ compilers; and several graphical and general purpose packages such as MATLAB.

Projects
Instead of tests, there will be one (for CPS405) or two (for CPS505) approximately three-week projects. The purpose of these projects will be to explore in greater detail than a homework assignment a problem related to the class material. Fully detailed handouts will be provided later.

The Web
The course Web page will be updated frequently. In addition to being passed out in the classroom, assignments and handouts will be posted on the course Web page. Answer keys, class announcements, and so on will also be posted. No grades will be accessible from the Web.

Grading
Projects: 20% (CPS405) or 40% (CPS505), weighted equally.
Homework: 80% (CPS405) or 60% (CPS505), weighted equally.
There will be no tests or final exam.

Late homework/projects will penalized 25% per day.

Writing/presentation skills and coding style and readability will count toward the grades. Your coding style does not have to be the same as mine; however, your code does need to be readable and to have a consistent style.

I am a great believer in partial credit. I can’t give credit for a one-line wrong answer, but if you show your work, you will receive partial credit for what you did correctly. (You will also receive higher quality, more targeted feedback).
Final grades will be assigned according to the following cutoffs. The cutoffs may be lowered under some circumstances, but they will never be raised.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Average, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>93.0 &amp; above</td>
</tr>
<tr>
<td>A-</td>
<td>90.0-92.9</td>
</tr>
<tr>
<td>B+</td>
<td>87.0-89.9</td>
</tr>
<tr>
<td>B</td>
<td>83.0-86.9</td>
</tr>
<tr>
<td>B-</td>
<td>80.0-82.9</td>
</tr>
<tr>
<td>C+</td>
<td>77.0-79.9</td>
</tr>
<tr>
<td>C</td>
<td>73.0-76.9</td>
</tr>
<tr>
<td>C-*</td>
<td>70.0-72.9</td>
</tr>
<tr>
<td>D+*</td>
<td>67.0-69.9</td>
</tr>
<tr>
<td>D*</td>
<td>63.0-66.9</td>
</tr>
<tr>
<td>D</td>
<td>60.0-62.9</td>
</tr>
<tr>
<td>E</td>
<td>Below 60.0</td>
</tr>
</tbody>
</table>

*These grades are not allowed for graduate level courses.

**Coding style guidelines**

In order to make it as easy as possible to understand and grade your work, I am gradually introducing a set of coding guidelines to be used in all of my upper level courses. At a minimum, please adhere to the following guidelines:

6. Begin your program with a comment block that includes your name, the homework problem or project number, and any other relevant identifying information, as well as any other comments you wish to include.

7. Separate your code into paragraphs interspersed with comment blocks. Paragraphs and/or comment blocks should be separated from one another by one empty line.

8. Indent in a way that enhances readability:
   a. Indent each for, do, while, or do-while statement block by some number of spaces (at least two).
   b. Indent each if block by some number of spaces (at least two).
   c. Do not use TAB characters to indent.
   d. In C/C++ programs, it does not matter to me whether braces off-setting statement blocks are on their own separate lines or not. However you choose, be consistent throughout the program.

9. In Fortran programs, use implicit none in all main programs, functions, and subroutines. Never rely on the Fortran naming convention.

10. Name variables, and other identifiers such as function names, in a manner that makes it as easy as possible to determine their scope and purpose. Here is a sampling of common practices, meant to inspire you to develop your own style:
   a. Local loop variables are named i, j, k, ...
   b. Variables appearing within only a small section of code begin with lower case letters (iStart, iEnd).
   c. Variables that are larger in scope, or other identifiers such as subroutine names, consist of short capitalized words strung together (e.g., SumX for sum of X, SumXX for sum of squares of X, NumIterMax for maximum number of iterations). An alternative is to use lower case and to separate words with underscores (e.g., sum_x, sum_xx).
   d. Parameters (in Fortran) or defined constants (in C/C++) have all upper case letters, with words separated by underscores (e.g., MAX_SIZE).
   e. Variable names are no longer than 15 characters.

An excellent book on all aspects of software construction is Mike McConnell's *Code Complete* (Microsoft Press). Although the book is aimed at business programmers, it is full of advice useful to all programmers.
Course policies are in accordance with “SUNY Brockport 2007-2008: Your Right to Know & Academic Policies Handbook”.

Students are expected to attend all classes. Absences will be excused for (a) documented illness, (b) official representation of the college, (c) death of a close relative, (d) religious holidays, and (e) other circumstances beyond the control of the student. The student is responsible for obtaining class notes and handouts for classes missed. Missing classes does not absolve the student of responsibility for any course work. In accordance with suggested College policy, any student whose unexcused absences exceed 15% of classes may receive a lowered grade or failure at the instructor’s discretion.

No incompletes or withdrawals will be given for this course except in cases of illness, personal tragedy, or extraordinary circumstances beyond the student’s control, and then only if documented to the instructor’s satisfaction.

While you are encouraged to work together, any assignments handed in must be your own work. Submitting material as your own work that is not entirely your authorship or knowingly providing an answer to another person constitutes cheating. If I am convinced beyond a reasonable doubt that cheating has occurred, the person(s) submitting the illicitly obtained answer, and any person(s) knowingly providing the same, will receive a grade of zero for that assignment. Further disciplinary procedures may also be considered.

SUNY Brockport is committed to maintaining a workplace and learning environment free of sexual harassment and intimidation. Sexual harassment is unacceptable behavior, unlawful, and intolerable.

I do not have the authority to cancel classes, no matter how severe the weather. Only the President of the College makes that decision. On days of severe weather, please listen to WBSU (89.9 FM) or WHAM (1180 AM) for class cancellations. You may also call 395-COLD. If I have any information, I will do my best to send email or post a message on my Web page.

Disability statement
Students with documented disabilities may be entitled to specific accommodations. SUNY Brockport’s Office for Students with Disabilities makes this determination. Please contact the Office for Students with Disabilities at 395-5409 or osdoffic@brockport.edu to inquire about obtaining an official letter to the course instructor detailing approved accommodations. The student is responsible for providing the course instructor with the official letter. Faculty and staff work as a team with the Office for Students with Disabilities to meet the needs of students with disabilities.

Disclaimer
Announcements given in class or on the Web are considered official addenda to course policies. All policies described are subject to change as the situation warrants. Should changes be necessary, all students will be notified in a timely fashion.