ABET

Self-Study Report

for the

Bachelor of Science in

Software Engineering

at

University of Virginia’s College at Wise

Wise, Virginia

REVISED

November 1, 2009
CONFIDENTIAL
The information supplied in this Self-Study Report is for the confidential use of ABET and its authorized agents, and will not be disclosed without authorization of the institution concerned, except for summary data not identifiable to a specific institution.
## Table of Contents

BACKGROUND INFORMATION .................................................................................................................................. 1

CRITERION 1. STUDENTS .................................................................................................................................. 3

CRITERION 2. PROGRAM EDUCATIONAL OBJECTIVES ...................................................................................... 8

CRITERION 3. PROGRAM OUTCOMES .................................................................................................................. 14

CRITERION 4. CONTINUOUS IMPROVEMENT .................................................................................................... 27

CRITERION 6. FACULTY ..................................................................................................................................... 45

CRITERION 7. FACILITIES ................................................................................................................................... 57

CRITERION 8. SUPPORT ..................................................................................................................................... 61

CRITERION 9. PROGRAM CRITERIA ...................................................................................................................... 64

APPENDIX A – ASSESSMENT RUBRICS AND SURVEY INSTRUMENTS ............................................................... 65

APPENDIX B – FACULTY RESUMES ................................................................................................................... 71

APPENDIX C – LABORATORY EQUIPMENT ......................................................................................................... 96

APPENDIX D – COURSE OUTLINES ................................................................................................................... 97

APPENDIX E – COURSE EVALUATION FORM .................................................................................................. 138

APPENDIX F – INSTITUTIONAL SUMMARY ....................................................................................................... 139
Self-Study Report

Bachelor of Science in Software Engineering
University of Virginia's College at Wise

BACKGROUND INFORMATION

A. Contact information
Alex Edwards
Department Chair
Abrar Qureshi
Assistant Professor of Software Engineering
Department of Mathematics and Computer Science

Mailing address:
Darden Hall
One College Ave.
Wise, Virginia, 24293

Tel. 276-376-4568
Fax. 276- 376-4589
Email: cae@uvawise.edu

B. Program History
Include year implemented and summarize major program changes with an emphasis on changes occurring since the last visit.

The Bachelor of Science in Software Engineering program started in 2006. We had our first graduating class in May 2009. This program relies heavily upon the Bachelor of Science in Computer Science program (also offered by the department and also seeking ABET accreditation) for its computer science core.

Since this is a new program, we have no changes to report at this time.

C. Options
List and describe any options, tracks, concentrations, etc. included in the program.
The Software Engineering program is a traditional day/evening lecture program. There are no options, tracks or concentrations for the Software Engineering program.

D. Organizational Structure
Use text and/or organization charts to describe the administrative structure of the program from the program to the department, college, and upper administration of your institution, as appropriate.

The administrative structure of the program starts with the Department Chair, Alex Edwards. The Department Chair reports to the Provost and Senior Vice Chancellor, Gil Blackburn. The Provost reports to the Chancellor, David Prior.

Graphically:

E. Program Delivery Modes
Describe the delivery modes used by this program, e.g., days, evenings, weekends, cooperative education, traditional lecture/laboratory, off-campus, distance education, web-based, etc.

Days, evenings, traditional lecture/laboratory

F. Deficiencies, Weaknesses or Concerns from Previous Evaluation(s) and the Actions taken to Address them
This is an initial accreditation.
CRITERION 1. STUDENTS

A. Student Admissions
Summarize the requirements and process for admission of students to the program. Complete and include the appropriate version of Table 1-1 for a baccalaureate or masters program.

All applicants to The University of Virginia’s College at Wise must be graduates of approved secondary schools or hold certificates of high school equivalency (GED). Applicants must submit a completed application, $25 nonrefundable application fee, official high school transcript, scores for the Scholastic Aptitude Test (SAT I) or American College Test (ACT), and an official transcript from all colleges attended, even for one course. While official dual enrollment transcripts are not required for admission, they will be required prior to registration for classes.

Our minimum requirements for admission are:

- High School diploma or equivalent (GED)
- ACT of 17 or higher
- SAT composite (Math and Verbal) 800 or higher

We have no specific requirements for pursuing the Bachelor of Science in Software Engineering degree.

B. Evaluating Student Performance
Summarize the process by which student performance is evaluated and student progress is monitored.

Students receive advising from the moment they arrive on campus through graduation (see next question for specifics). This advising provides mechanisms for evaluating student progress.

The college has a system using grade points to determine minimal academic progress. Basically a student has a grade point deficit if his/her GPA drops below 2.0. There is a structure through which the students move, starting with academic warning, followed by probation, followed by suspension.

The student is given ample opportunity to correct grade point deficits and in most cases can re-take a course with a grade of C- or lower. The new grade will replace the old grade in grade point calculations.

D. Advising Students
Summarize the process by which students are advised regarding curricular and career matters.

When incoming students first sign-up for classes, they are advised by faculty from the department in which they are interested. Our faculty shares this responsibility by taking turns at the four freshman-orientation sessions held by the Admissions Office as well as the transfer student orientation sessions.
All freshmen are required to take a 2-semester (Freshman Seminar) sequence in which they are assigned a temporary advisor. This advisor is a faculty member at the college but students who have a particular interest are paired with a faculty member from that target department when possible. When students declare a specific major, they are assigned a permanent advisor (a faculty member in the target department) by the Department Chair. Students can access their advisor during normal school hours, typically 8-4 M-F. Usually students drop by during faculty office hours but they can also make appointments.

In addition to their academic advisor, students have access to career service and counseling professionals through the college’s Center for Student Development.

(http://www.uvawise.edu/student_development/index.html)

Here they can receive advice and information on specific career paths.

E. Transfer Students and Transfer Courses
Summarize the requirements and process for accepting transfer students and transfer credit. Complete and include Table 1-2.

1. A transfer student should have a grade point average of at least 2.2. Other applicants with a grade point average of less than 2.2 but at least 2.0 are reviewed by the Admissions Committee. If admitted, these students may be subject to provisional status. Students with less than a 2.0 grade point average normally are not admitted to the college.

2. Transfer students must be in good standing and eligible to return to the college or university last attended at which they were enrolled full time in a degree program.

3. The college or university from which the student is transferring should be fully accredited as an institution of higher learning by a state or regional accrediting organization.

4. Only those courses which are equivalent to University of Virginia or UVa-Wise courses are eligible for transfer.

5. Transfer credit will be granted only if a grade of "C-" or better was obtained or a "CR" was obtained in a course taken with "credit-no credit" grading. Students transferring from the Virginia Community College System or Richard Bland College should see the section below for exceptions.

6. A maximum of 62 semester hours earned in a two-year college may be counted toward graduation at UVa-Wise.

7. All transfer students must meet the UVa-Wise residence requirements for graduation

F. Graduation Requirements
Summarize the process for ensuring that each graduate completes all graduation requirements for the program.
All graduates must complete the degree requirements as specified in the college catalog, page 118 of the 2009-2010 edition:

1. A minimum of 120 semester hours. (At most, two hours earned in physical education activities courses may be counted as part of the 120 hour requirement.) Completion of the general education requirements is considered to be part of the 120 hours.

2. An overall grade point average of 2.00 or higher for all college level work (including work transferred to UVa-Wise) and 2.00 or higher on all work attempted at UVa-Wise.

3. In the major area, a grade point average of 2.00 or higher as prescribed by the major.

4. Completion of requirements for one of the major program areas and the general education requirements as described in the catalog of record.

5. Enrollment at UVa-Wise for the semester in which degree requirements are completed.

6. Completion of at least 45 semester hours through UVa-Wise, with a minimum of 15 semester hours of upper-level courses completed in the major.

7. Completion of at least 58 semester hours in a regionally accredited four-year college or university.

8. The filing of an application for graduation with the Office of the Registrar at least one full semester prior to the anticipated graduation date.

9. Completion of the Cultural Activities Requirement (see page 81 on the General Education Core).

To move on from prerequisite courses, our students must attain a C or better in their classes.

G. Enrollment and Graduation Trends
Summarize the enrollment and graduation trends for the past five years. Complete and include Table 1-3.
Enrollment Trends for Past Five Academic Years

This Chart shows that our enrollment trend for the past five years has been stable.

Table 1-1. History of Admissions Standards for Freshmen Admissions for Past Five Years

<table>
<thead>
<tr>
<th>Academic Year</th>
<th>Composite ACT</th>
<th>Composite SAT</th>
<th>Percentile Rank in High School</th>
<th>Number of New Students Enrolled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MIN.</td>
<td>AVG.</td>
<td>MIN.</td>
<td>AVG.</td>
</tr>
<tr>
<td>2008/09</td>
<td>17</td>
<td>21</td>
<td>800</td>
<td>1005</td>
</tr>
<tr>
<td>2007/08</td>
<td>17</td>
<td>20</td>
<td>800</td>
<td>1005</td>
</tr>
<tr>
<td>2006/07</td>
<td>17</td>
<td>21</td>
<td>800</td>
<td>1003</td>
</tr>
<tr>
<td>2005/06</td>
<td>17</td>
<td>21</td>
<td>800</td>
<td>1003</td>
</tr>
<tr>
<td>2004/05</td>
<td>17</td>
<td>20</td>
<td>800</td>
<td>1000</td>
</tr>
</tbody>
</table>

Table 1-2. Transfer Students for Past Five Academic Years

<table>
<thead>
<tr>
<th>Academic Year</th>
<th>Number of Transfer Students Enrolled</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008/09</td>
<td>122</td>
</tr>
<tr>
<td>2007/08</td>
<td>134</td>
</tr>
<tr>
<td>2006/07</td>
<td>146</td>
</tr>
<tr>
<td>2005/06</td>
<td>163</td>
</tr>
<tr>
<td>2004/05</td>
<td>140</td>
</tr>
</tbody>
</table>
### Table 1. Enrollment Trends for Past Five Academic Years

<table>
<thead>
<tr>
<th></th>
<th>Year 2004</th>
<th>Year 2005</th>
<th>Year 2006</th>
<th>Year 2007</th>
<th>Year 2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-time Students</td>
<td>1,432</td>
<td>1,456</td>
<td>1,505</td>
<td>1,509</td>
<td>1,448</td>
</tr>
<tr>
<td>Part-time Students</td>
<td>404</td>
<td>497</td>
<td>404</td>
<td>294</td>
<td>516</td>
</tr>
<tr>
<td>Student FTE(^1)</td>
<td>1,557</td>
<td>1,599</td>
<td>1,609</td>
<td>1,610</td>
<td>1,581</td>
</tr>
<tr>
<td>Graduates</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
</tr>
</tbody>
</table>

\(^1\) FTE = Full-Time Equivalent

### Table 1. Program Graduates

(For Past Five Years or last 25 graduates, whichever is smaller)

<table>
<thead>
<tr>
<th>Numerical Identifier</th>
<th>Year Matriculated</th>
<th>Year Graduated</th>
<th>Prior Degree(s) if Master Student</th>
<th>Certification/Licensure (If Applicable)</th>
<th>Initial or Current Employment/Job Title/Other Placement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fall 2005</td>
<td>2009</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Fall 2005</td>
<td>2009</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Fall 2006</td>
<td>2009</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Note: ABET recognizes that current information may not be available for all students)
CRITERION 2. PROGRAM EDUCATIONAL OBJECTIVES

ABET Definition: *Program educational objectives* are broad statements that describe the career and professional accomplishments that the program is preparing graduates to achieve.

Assessment under this criterion is one or more processes that identify, collect, and prepare data to evaluate the achievement of program educational objectives.

Evaluation under this criterion is one or more processes for interpreting the data and evidence accumulated through assessment practices. Evaluation determines the extent to which program educational objectives are being achieved, and results in decisions and actions to improve the program.

A. Mission Statement

Provide a copy or summary of any applicable institutional, college, departmental, and program Mission Statements and document where they are published.

The University of Virginia’s College at Wise, a public liberal arts institution, provides students with learning experiences that offer opportunities to develop the insight, competence, sensitivity, and integrity necessary for living enriched lives and for enriching the lives of others. Established in 1954 as a college of the University of Virginia, it is guided by the values of citizenship and altruism. Proud of its Appalachian heritage, the College continues to honor its commitment of service to Southwest Virginia, the nation, and the world. The College is guided by a legacy of teaching and scholarly excellence and by a dedication to quality in both the arts and sciences and professional programs. Above all, The University of Virginia’s College at Wise is a diverse community of people who believe that information can be transformed into knowledge and that teaching and learning create a foundation for wisdom.

This mission statement is published in the College Catalog (page 4 of the 2009-2010 edition) and online at: [http://www.uvawise.edu/academics/documents/missionandgoals_000.pdf](http://www.uvawise.edu/academics/documents/missionandgoals_000.pdf)

B. Program Educational Objectives

List the Program Educational Objectives and state where these are published.

*Our students, after graduating with a degree in Software Engineering at the University of Virginia’s College at Wise will be able to:*

- **POE1.** Pursue careers or advanced study in their field.
- **POE2.** Effectively communicate with their peers, customers, supervisors, etc through both written and oral means.
- **POE3.** Embrace lifelong learning.
POE4. Analyze and assess problems/situations and provide/suggest appropriate solutions.

These educational objectives can be found online at the Departmental Website: 

http://www.mcs.uvawise.edu/abet/peo

C. Consistency of the Program Educational Objectives with the Mission of the Institution
Describe how the Program Educational Objectives are consistent with the Mission of the Institution.

Our educational objectives directly align with the mission of the college. By pursuing careers or advanced degrees in their chosen field (PEO 1 and PEO2), our students are leading enriched lives and potentially enriching the lives of others through lifelong learning activities. Through effective communication (PEO 3), our students will be able to contribute to society and foster the ideals learned here at the University of Virginia’s College at Wise. Also, by applying their knowledge and solving problems (PEO4), our students will contribute to society and potentially enrich the lives of others.

D. Program Constituencies
List and describe the Program Constituencies.

Our constituents need skilled problem solvers with excellent communication skills. Our stakeholders include local/regional/national/global businesses, government agencies, and graduate schools. We have a program Advisory Board whose members come directly from these constituencies. This board provides feedback and suggestions for formulating our educational objectives so that they match the expectations of said constituencies.

Some specific stakeholders include two global corporations with offices about 40 miles away from Wise: Northrup Grumman and CGI. Virginia Tech and ETSU are common graduate schools at which our students apply/attend.

Our educational objectives directly align with the needs of our constituents: they need skilled problem solvers with communication skills; our objectives directly match these needs.

E. Process for Establishing Program Educational Objectives
Describe the process that periodically documents and demonstrates that the Program Educational Objectives are based on the needs of the program's various constituencies.

We have a program Advisory Board whose members come directly from these constituencies. This board provides feedback and suggestions for formulating our educational objectives so that they match the expectations of said constituencies.

Our educational objectives directly align with the needs of our constituents: they need skilled problem solvers with communication skills; our objectives directly match these needs.
Here is a graphical depiction of our process:

**Program Educational Objectives**

We start with the College’s Mission statement, our Faculty’s knowledge, and expectations from our constituents. Our faculty then develops initial program educational objectives. These objectives are later evaluated through employer and alumni surveys. The results of these surveys are passed to the advisory board (made up of our constituents) which then determines adequacy of the objectives and makes recommendations to the faculty for changes when necessary.

**F. Achievement of Program Educational Objectives**

Describe the assessment and evaluation process that periodically documents and demonstrates the degree to which the Program Educational Objectives are attained.

POE1. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes.

Mechanisms:

Employer Survey question 5.

Alumni Survey description of current duties.


POE2. Effectively Solve problems in a team environment and effectively communicate with their peers, customers, supervisors, etc through both written and oral means.
Mechanisms:
Employer Survey question 2.

POE3. Demonstrate professionalism in their work and embrace lifelong learning.
Mechanisms:
Employer Survey question 4.

POE4. Skilled in applying engineering processes and practices to software components and systems.
Mechanisms:
Employer Survey questions 1 and 5.
Alumni Survey question 4.

POE5. Contribute to society in a meaningful manner and behave ethically and responsibly.
Mechanisms:
Employer Survey question 4.

**How Often?**
We plan to perform the Employer and Alumni surveys yearly after 2012. 2012 will be the first year we will collect data on our first graduates (for those who graduate in 2009).

**How?**
The Faculty sends email with the attached survey to appropriate employers and alumni. Because we have manageable numbers of graduates, we plan to send the Alumni survey to all graduates in the 2009 graduating class. We would then perform the first data collection in the summer of 2012, targeting our 2009 graduates (for the Alumni survey). We anticipate that as our alumni base grows, we will have to do random selection during the data collection process for subsequent years.

After we have collected the data, aggregation and analysis will be performed by the Faculty and results sent to the Advisory Board during the fall meeting. Our first year of data collection/analysis will be 2012.
Who?
The faculty members at UVA-Wise will be responsible for sending out the surveys and collecting the responses. Aggregation and analysis will be done by the faculty with reports delivered to the Advisory Board at the fall meeting (first meeting with PEO data will be 2012). Recommendations of changes and suggested improvements will occur at the Advisory Board meeting.

Please see section 4 for more information on our assessment process.

Here is our Alumni Survey:
Alumni Survey
Department of Mathematics and Computer Science
The University of Virginia's College at Wise (Uva-Wise)

Date: __________________ Your Name (optional) ___________________________

In what year did you graduate? __________________________

Who is your current employer (if pursuing a degree, which school)?

List any advanced degrees, certificates, or professional memberships

List your job title and give a description of your typical duties.

For the following statements, rank how well your degree helped you...

1. Pursue careers or advanced study in your field.
   strongly agree neutral disagree strongly disagree

2. Effectively communicate with your peers, customers, supervisors, etc. through written and oral means.
   strongly agree neutral disagree strongly disagree

   strongly agree neutral disagree strongly disagree

4. Analyze and assess problems/situations and provide/suggest appropriate solutions.
   strongly agree neutral disagree strongly disagree

Please give us any specific comments about achievements you've had in any of the above areas.

Please give any specific comments about ways your degree could do a better job preparing you for the above areas.

Return to: Jacob Somervell, 221 Darden Hall, 1 College Ave, Wise, VA 24230
email ips54@uvawis.edu
fax 276-378-4589
CRITERION 3. PROGRAM OUTCOMES

ABET definition: Program outcomes are narrower statements that describe what students are expected to know and be able to do by the time of graduation. These relate to the skills, knowledge, and behaviors that students acquire in their matriculation through the program.

Assessment under this criterion is one or more processes that identify, collect, and prepare data to evaluate the achievement of program outcomes.

Evaluation under this criterion is one or more processes for interpreting the data and evidence accumulated through assessment practices. Evaluation determines the extent to which program outcomes are being achieved, and results in decisions and actions to improve the program.

A. Process for Establishing and Revising Program Outcomes

Describe the process used for establishing and revising Program Outcomes.

The program has documented measurable outcomes that are based on the needs of the program's constituencies.

The program enables students to achieve, by the time of graduation:

(a) an ability to apply knowledge of mathematics, science, and engineering
(b) an ability to design and conduct experiments, as well as to analyze and interpret data
(c) an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
(d) an ability to function on multidisciplinary teams
(e) an ability to identify, formulate, and solve engineering problems
(f) an understanding of professional and ethical responsibility
(g) an ability to communicate effectively
(h) the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context
(i) a recognition of the need for, and an ability to engage in life-long learning
(j) a knowledge of contemporary issues
(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.
The graphic above depicts our process for creating and evaluating our program outcomes. Starting with our program educational objectives, our faculty develops program outcomes that align with those objectives and also provide coverage of the previously listed outcomes.

B. Program Outcomes
List the Program Outcomes and describe how they encompass Criterion 3 and any applicable Program Criteria. Indicate where the Program Outcomes are documented.

Upon graduation from the Bachelor of Science in Software Engineering program at the University of Virginia’s College at Wise, graduates will have:

PO1. An ability to apply an engineering approach to the development of software systems by analyzing, designing, implementing, verifying, validating and maintaining software systems.

PO2. An ability to exhibit technical competency as well as leadership and communication skills necessary to analyze, design, verify, validate, implement and maintain software systems.

PO3. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.
PO4. An ability to demonstrate the necessary communication skills to elicit, analyze and specify software requirements and to create well-written software documentation.

PO5. An ability to demonstrate the necessary organizational and business skills to work effectively in a team setting and to be able to predict the time and cost needed to create and to maintain software systems.

PO6. An ability to apply appropriate codes of ethics and professional conduct to the solution of software engineering problems.

PO7. An ability to demonstrate an awareness of their professional and social responsibility as software engineers through professionalism, continued learning and application of their knowledge for the good of society.

These program outcomes can be found on our departmental web site:

http://www.mcs.uvawise.edu/abet/po
We use senior capstone course projects extensively in assessment of our program. This project allows our students to exhibit their mastery of the course content for the degree and to illustrate their competency in both written and oral communication. In addition to the senior capstone projects, we
also use a sophomore level software engineering course that focuses heavily on team and group work. See section 4 for more details on our assessment processes.

**Program outcome (and primary assessment method).**

PO1. An ability to apply an engineering approach to the development of software systems by analyzing, designing, implementing, verifying, validating and maintaining software systems.

  Assessment mechanisms: Departmental Examination, Capstone Project

  **Departmental Examination. [Direct Measure]** The following questions from the departmental exam address Program Outcome 1. **Software Engineering exam**
  
  Part 1: 5, 7, 8, 12
  
  Part 2: 2, 4, 6

  Target: All students score 50% or higher on the exam.

  **Capstone Project. [Direct Measure]** We use section 2 (Design) and 3 (Testing) of our SE Assessment Rubric to assess how our students meet program outcome 1.

  Target: We expect our students to score “3” or higher on each section.

PO2. An ability to exhibit technical competency as well as leadership and communication skills necessary to analyze, design, verify, validate, implement and maintain software systems.

  Assessment mechanisms: Departmental Examination, Major Field Achievement Test, Capstone Project

  **Departmental Examination. [Direct Measure]** The following questions from the departmental exam address Program Outcome 2.

  **Software Engineering exam**
  
  Part 1: 2, 4, 6, 9, 12, 14
  
  Part 2: 2, 3, 6

  Target: All students score 50% or higher on the exam.

  **Capstone Project. [Direct Measure]** We use section 2 (Design) and 3 (Testing) of our SE Assessment Rubric to assess how our students meet program outcome 2.

  Target: We expect our students to score “3” or higher on each section.

PO3. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

  **Software Engineering exam**
  
  Part 1: 3, 4, 10
  
  Part 2: 3 (we also use CS sections of the Departmental exam)
Assessment mechanisms: Capstone Project

**Capstone Project. [Direct Measure]** We use section 2 (Design) of our SE Assessment Rubric to assess how our students meet program outcome 3.
Target: We expect our students to score “3” or higher on each section.

**PO4.** An ability to demonstrate the necessary communication skills to elicit, analyze and specify software requirements and to create well-written software documentation.

**Software Engineering exam**
Part 1: 1, 14
Part 2: 1

Assessment mechanisms: Capstone Project

**Capstone Project. [Direct Measure]** We use college wide oral and written communication rubrics to assess program outcome 4. We use section 1 (Documentation/Process) of our SE assessment Rubric.
Target: We expect our students to average “3” or higher on each rubric.

**PO5.** An ability to demonstrate the necessary organizational and business skills to work effectively in a team setting and to be able to predict the time and cost needed to create and to maintain software systems.

**Software Engineering exam**
Part 1: 11, 13
Part 2: 5, 6, 7, 8

Assessment mechanisms: CSC 2300 Team Project (self assessment)

**CSC 2300 Team Project [Direct Measure]** We use the semester-long development project from our introduction to Software Engineering to assess how well our students meet program outcome 5. The students “grade” themselves and their teammates on how they performed. This is a numeric value (0-100) that is used to weight their project grade (instructor can decide not to do this).

Target: All students will be graded with 85% or higher (including their self evaluation).

**PO6.** An ability to apply appropriate codes of ethics and professional conduct to the solution of software engineering problems.

**Software Engineering exam**
Part 1: 10
**Capstone Project. [Direct Measure]** We use section 5(Societal Awareness) of our SE assessment Rubric.

Target: We expect our students to average “3” or higher

PO7. An ability to demonstrate an awareness of their professional and social responsibility as software engineers through professionalism, continued learning and application of their knowledge for the good of society.

**Capstone Project. [Direct Measure]** By successfully completing the Capstone our students demonstrate program outcome 7.


### C. Relationship of Program Outcomes to Program Educational Objectives

Describe how the Program Outcomes lead to the achievement of the Program Educational Objectives.

<table>
<thead>
<tr>
<th>Program Outcome</th>
<th>Program Educational Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO1. An ability to apply an engineering approach to the development of software systems by analyzing, designing, implementing, verifying, validating and maintaining software systems.</td>
<td>POE1. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes. POE4. Skilled in applying engineering processes and practices to software components and systems.</td>
</tr>
<tr>
<td>PO2. An ability to exhibit technical competency as well as leadership and communication skills necessary to analyze, design, verify, validate, implement and maintain software systems.</td>
<td>POE1. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes. POE4. Skilled in applying engineering processes and practices to software components and systems.</td>
</tr>
<tr>
<td>PO3. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.</td>
<td>POE1. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes. POE4. Skilled in applying engineering processes and practices to software components and systems.</td>
</tr>
<tr>
<td>PO4. An ability to demonstrate the necessary communication skills to elicit, analyze and specify software requirements and to create well-written software documentation.</td>
<td>POE2. Effectively solve problems in a team environment and effectively communicate with their peers, customers, supervisors, etc through both written and oral means.</td>
</tr>
<tr>
<td>PO5. An ability to demonstrate the necessary organizational and business skills to work effectively in a team setting and to be able to predict the time and cost needed to create and to maintain software systems.</td>
<td>POE1. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes. POE2. Effectively solve problems in a team environment and effectively communicate with their peers, customers, supervisors, etc through both written and oral means. POE4. Skilled in applying engineering processes and practices to software components and systems.</td>
</tr>
<tr>
<td>PO6. An ability to apply appropriate codes of ethics and professional conduct to the solution of software engineering problems.</td>
<td>POE5. Contribute to society in a meaningful manner and behave ethically and responsibly.</td>
</tr>
<tr>
<td>PO7. An ability to demonstrate an awareness of their professional and social responsibility as software engineers through professionalism, continued learning and application of their knowledge for the good of society.</td>
<td>POE3. Demonstrate professionalism in their work and embrace lifelong learning. POE5. Contribute to society in a meaningful manner and behave ethically and responsibly.</td>
</tr>
</tbody>
</table>

### D. Relationship of Courses in the Curriculum to the Program Outcomes

Describe the relationship of courses in the curriculum to the Program Outcomes.

Recall that our program outcomes are:

**PO1.** An ability to apply an engineering approach to the development of software systems by analyzing, designing, implementing, verifying, validating and maintaining software systems.

**PO2.** An ability to exhibit technical competency as well as leadership and communication skills necessary to analyze, design, verify, validate, implement and maintain software systems.
PO3. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4. An ability to demonstrate the necessary communication skills to elicit, analyze and specify software requirements and to create well-written software documentation.

PO5. An ability to demonstrate the necessary organizational and business skills to work effectively in a team setting and to be able to predict the time and cost needed to create and to maintain software systems.

PO6. An ability to apply appropriate codes of ethics and professional conduct to the solution of software engineering problems.

PO7. An ability to demonstrate an awareness of their professional and social responsibility as software engineers through professionalism, continued learning and application of their knowledge for the good of society.

Our courses provide our students with the knowledge and skills necessary for covering our program outcomes. Almost all of the courses a graduate could take in completion of the degree requirements would provide coverage of our program outcomes 1, 2, and 4. Our introductory software engineering course, CSC 2300 uses extensive team based work to provide students with team experiences (PO5). The senior capstone experience (SWE 4980/4990) requires students to work on teams (PO5). Program outcome 3 (PO3) is covered in CSC 3260 – Introduction to Human Computer Interaction and in SWE 1790 – Engineering Leadership. In addition, societal/ethical issues are often included in discussions and course material in many other courses. The following table shows a breakdown of how our courses contribute to the outcomes.

The entries in the following table indicate:

- X – program outcome is addressed as a major part of the course
- S – program outcome is addressed somewhat in the course

<table>
<thead>
<tr>
<th>Course</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSC 1180 Fundamentals of Programming</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>CSC 2300 Introduction to Software Engineering</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>CSC 2180 Data Structures</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>S</td>
</tr>
<tr>
<td>CSC 3260 Introduction to Human - Computer Interaction</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>S</td>
</tr>
<tr>
<td>CSC 3400 Database Design</td>
<td>X</td>
<td></td>
<td>X</td>
<td>S</td>
<td></td>
<td></td>
<td>S</td>
</tr>
<tr>
<td>CSC 3710 Discrete Mathematics</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSC 3180 Introduction to Algorithms</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>CSC 4000 Operating Systems</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>Course Code</td>
<td>Course Title</td>
<td>X</td>
<td>S</td>
<td>X</td>
<td>S</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSC 4200</td>
<td>Programming Languages</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSC 4300</td>
<td>Computer Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTH 1180</td>
<td>Probability and Statistics</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTH 2040</td>
<td>Calculus I</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTH 2050</td>
<td>Calculus II</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTH 3060</td>
<td>Calculus III</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTH 3130</td>
<td>Intro to Linear Algebra</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHY 3010/2010</td>
<td>College Physics I/Lab</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHY 3020/2020</td>
<td>College Physics II/Lab</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COM 3450</td>
<td>Technical Writing</td>
<td>S</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 1790</td>
<td>Engineering Leadership</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 2790</td>
<td>Engineering Economics</td>
<td>X</td>
<td>S</td>
<td>X</td>
<td>S</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 2130</td>
<td>Software Construction Technology</td>
<td>X</td>
<td>X</td>
<td>S</td>
<td>S</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 2210</td>
<td>Testing, Verification and Validation</td>
<td>X</td>
<td>X</td>
<td>S</td>
<td>S</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 3210</td>
<td>Software Quality Assurance</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 3220</td>
<td>Software Requirements and Modeling</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 3230</td>
<td>Software Configuration Management</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 4240</td>
<td>Software Project Management</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 4980</td>
<td>Capstone Project</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 4990</td>
<td>Capstone Project</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**E. Documentation**

Describe by example how the evaluation team will be able to relate the display materials, i.e., course syllabi, sample student work, etc., to each Program Outcome.

Our course syllabi have a section devoted to how the course relates to the program outcomes. Please see any of the course outlines in Appendix D for an example. A better example would be to view our assessment information in Section 4 on continuous improvement.

**F. Achievement of Program Outcomes**
Explain the assessment and evaluation processes that periodically document and demonstrate the degree to which the Program Outcomes are attained. Describe the level of achievement of each Program Outcome. Discuss what evidence will be provided to the evaluation team that supports the levels of achievement of each Program Outcome.

The following table lists our program outcomes, our assessment measures, and target levels.

<table>
<thead>
<tr>
<th>Program Outcome</th>
<th>Measure</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO1. An ability to apply an engineering approach to the development of software systems by analyzing, designing, implementing, verifying, validating and maintaining software systems.</td>
<td>Departmental Examination</td>
<td>All graduates will score 50% or higher on the exam</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Design</td>
<td>3 or higher</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Testing</td>
<td>3 or higher</td>
</tr>
<tr>
<td>PO2. An ability to exhibit technical competency as well as leadership and communication skills necessary to analyze, design, verify, validate, implement and maintain software systems.</td>
<td>Departmental Examination</td>
<td>All graduates will score 50% or higher on the exam</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Documentation</td>
<td>3 or higher</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Design</td>
<td>3 or higher</td>
</tr>
<tr>
<td>PO3. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.</td>
<td>Departmental Examination</td>
<td>All graduates will score 50% or higher on the exam</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Design</td>
<td>3 or higher</td>
</tr>
<tr>
<td>PO4. An ability to demonstrate the necessary communication skills to elicit, analyze and specify software requirements and to create well-written software documentation.</td>
<td>Departmental Examination</td>
<td>All graduates will score 50% or higher on the exam</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Documentation</td>
<td>3 or higher</td>
</tr>
<tr>
<td>PO5. An ability to demonstrate the necessary organizational and business skills to work effectively in a team setting and to be able to predict the time and cost needed to create and to maintain software systems.</td>
<td>Departmental Examination</td>
<td>All graduates will score 50% or higher on the exam</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td></td>
<td>CSC 2300 – Software Engineering Team Project Score</td>
<td>All team members will be scored &gt;=85% by their peers on the team project.</td>
</tr>
<tr>
<td>PO6. An ability to apply appropriate codes of ethics and professional conduct to the solution of software engineering problems.</td>
<td>Departmental Examination</td>
<td>All graduates will score 50% or higher on the exam</td>
</tr>
<tr>
<td></td>
<td>SE Rubric section on Societal Awareness</td>
<td>3 or higher</td>
</tr>
<tr>
<td>PO7. An ability to demonstrate an awareness of their professional and social responsibility as software engineers through professionalism, continued learning and application of their knowledge for the good of society.</td>
<td>Senior Capstone Project</td>
<td>Successful completion of the capstone</td>
</tr>
</tbody>
</table>

Here is our software engineering assessment rubric:
Software Engineering Assessment Rubric for student: _________________________

Documentation/Process
5. Excellent documentation of process, clear requirements with traceability, clear schedule and plan
4. Good documentation of process, requirements clear, reasonable schedule shown
3. Process is clear from documentation, requirements are listed (some may be missing), schedule may be unreasonable
2. Some process used, poorly documented, vague requirements, weak schedule
1. No process evident, poor documentation, vague/limited requirements, no schedule or plan

Design
5. Flexible design, exceeds requirements, delightful interface
4. Some flexibility in design, exceeds requirements, usable interface
3. Meets explicit requirements, interface may be overly complex or confusing, rigid design
2. Some requirements missing, frustrating interface, rigid design
1. Missing requirements, rigid design, unusable

Testing
5. Extensive testing plans, excellent data collection and analysis, test cases clearly mapped to requirements
4. Adequate test plans, good data collection and analysis, most test cases tied to requirements
3. Test plans provided, some collection and analysis, some test cases tied to requirements
2. Weak test plan, weak collection/analysis, test cases not clearly tied to requirements
1. No testing evident

Societal Awareness
5. Student identifies most potential ethical or societal issues with his/her solution and adequately discusses ramifications.
4. Student identifies some potential ethical or societal issues with his/her solution but may not adequately discuss ramifications.
3. Student identifies at least one potential ethical or societal issue with his/her solution but does not discuss ramifications.
2. Student vaguely mentions ethical or societal issues but may not relate directly to his/her solution.
1. Student makes no attempt to identify or discuss ethical or societal issues.

Term: _______________________________
Completed By: _______________________
CRITERION 4. CONTINUOUS IMPROVEMENT

A. Information Used for Program Improvement
Describe the available information, such as results from the Criteria 2 and 3 processes, commonly used in making decisions regarding program improvements

The University of Virginia’s College at Wise Department of Mathematics and Computer Science has been doing some form of assessment for several years. As our school is accredited by the Southern Association of Colleges and Schools (SACS), we are required to perform assessment. In addition, we are required by the State Council of Higher Education for Virginia (SCHEV) to perform annual assessment of our program. In learning about ABET’s requirements, we have made some changes to our assessment processes to bring our program into compliance. Much of our assessment is taken directly from our SCHEV and SACS assessment plan but some modifications were required to align our processes with the ABET requirements.

Program Educational Objectives Assessment Results

We assess our educational objectives by performing alumni surveys, employer surveys, and through meetings with our advisory board. The results of our surveys should indicate how well our graduates are performing in either their jobs, or with pursuing advanced degrees. Our advisory board uses this information in assessing our progress towards meeting our objectives, and to provide feedback and suggestions. Here is a graphical depiction of our process.

Please see Appendix B for the employer and alumni surveys.
PEO1. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes.
Mechanisms:
Employer Survey question 5.
Alumni Survey description of current duties.

PEO2. Effectively Solve problems in a team environment and effectively communicate with their peers, customers, supervisors, etc through both written and oral means.
Mechanisms:
Employer Survey question 2.

PEO3. Demonstrate professionalism in their work and embrace lifelong learning.
Mechanisms:
Employer Survey question 4.

PEO4. Skilled in applying engineering processes and practices to software components and systems.
Mechanisms:
Employer Survey questions 1 and 5.
Alumni Survey question 4.

PEO5. Contribute to society in a meaningful manner and behave ethically and responsibly.
Mechanisms:
Employer Survey question 4.

Results
We have not administered the employer or alumni surveys yet. We recently received feedback from our Advisory Board on the survey instruments, and we plan to administer the surveys in the Spring of 2010.

Changes Resulting from Assessment
We received information from our Advisory Board about the necessity of having teamwork and professionalism as part of our Educational Objectives.

ACTION(S):
Administer the surveys (employer and alumni) in January of 2010.

Frequency and timing of assessments.
We perform our employer and alumni surveys every year, for our 3-year graduates.
Our first data will be collected in 2012 for our 2009 graduates. We meet with the advisory board once per year.
What data are collected (should include information on initial student placement and subsequent professional development).

We ask Northrup Grumman and CGI to fill out an employer survey on graduates from our degree programs in the summer. These surveys probe issues like team performance, communication, and overall ability to perform. The alumni surveys give our graduates the opportunity to provide information on what types of jobs they have, how long they have been working, career advancement, etc.

How data are collected.

We send the employer surveys to the company managers in Lebanon, VA. The survey is an editable PDF document. They return the completed survey to us and we extract the relevant data.

For the alumni surveys, we mail our former students a copy of an editable PDF and they return it to us. We extract the relevant data from those surveys.

From whom data are collected (should include students and computing professionals).

We collect data from computing professionals (employers and professors) and our former students.

How assessment results are used and by whom.

The Department of Mathematics and Computer Science uses this data to determine the extent we are meeting our educational objectives. The data is tallied and analyzed by the department and provided to the advisory board.

The advisory board helps the department determine the extent of meeting the educational objectives and whether any actions are required.

Program Outcomes Assessment Results

Frequency and timing of assessments.

We perform our Program Outcomes assessment on a yearly basis.
The Senior Capstone projects are assessed at the end of the Spring semester.
The Major Field Achievement Test (MFAT) is administered in the Spring semester.
The Departmental Examination is administered in the Spring semester.
Graduate Exit Survey is conducted after classes are finished but before graduation.

What data are collected.

Rubrics are completed by Faculty members after reviewing Project materials. Data is recorded and aggregated per student.
MFAT numeric scores are automatically generated and accessed through the web.
The Departmental Examination is scored by the Faculty. Results are recorded per student.
Graduate Exit Survey results are tabulated automatically in our Moodle server.

**How data are collected.**
Every graduating senior takes the MFAT and Departmental Examination in the Spring semester. Scores are collected and analyzed using spreadsheet programs. Capstone rubrics are collected, data added to spreadsheet where analysis is performed.
Students log in to Moodle and take the Graduate Exit Survey. Results are automatically saved in that software.

**From whom data are collected.**
We collect the data from our graduating students.

**How assessment results are used and by whom.**
The Department of Mathematics and Computer Science uses this data to determine the extent we are meeting our program outcomes. The data is tallied and analyzed by the department and provided to the advisory board.

**Results:**
We had 3 students graduate from our Software Engineering Program in 2009 (our first for the program).

Departmental Examination
Student 1: 68%
Student 2: 50.67%
Student 3: 67.33%

MFAT:
Student 1: 137
Student 2: 157
**Student 3: 129**
Mean: 149.3
St.Dev: 16.1
Lowest: 133

Capstone Experience
<table>
<thead>
<tr>
<th>Documentation</th>
<th>Design</th>
<th>Testing</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student 1</td>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Student 2</td>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Student 3</td>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>
Summary Table:

PO1.
We partially met this outcome. The reason was lack of reporting on Testing in second part of Capstone project.

ACTION(S): We need to emphasize documentation of Testing in the Capstone for Fall 2009.

PO2.
We partially met this outcome. The reason was lack of reporting on Testing in second part of Capstone project.

ACTION(S): We need to emphasize documentation of Testing in the Capstone for Fall 2009.

PO3.
We met our target for PO3. There are no actions necessary to correct any deficiencies.

ACTION(S): None.

PO4.
We met our target for PO4. There are no actions necessary to correct any deficiencies.

ACTION(S): None.

PO5. We partially met this outcome. One student failed to make the 85% rating on teamwork. We do not have sufficient data at this time. We will watch this program outcome in the next assessment cycle.

ACTION(S): Watch for data trend to monitor the progress.

PO6.
We partially met this outcome.

ACTION(S): Apply the updated Rubric for Spring 2010.
PO7.

We met our target for PO7. There are no actions necessary to correct any deficiencies.

ACTION(S): None.

B. Actions to Improve the Program

Describe actions taken to improve the program since the last general review. Indicate why, i.e., the basis for taking action, and when each action was implemented and the results of the implementation.

Any program changes within the last six years based on assessments.

2005-06

Created and Implemented Bachelor of Science in Software Engineering degree

2006-07

Courses were taught first time

2007-08

No modifications

2008-09

This is our first year with assessment data, analysis complete. We already see a need to increase focus on testing, or at least the reporting thereof. One student did not meet the minimum score for the MFAT test. We will discuss ramifications with our Board. We will present this data to our Advisory Board on October 6, 2009. The board will make recommendations on improvements at that time.

Any significant future program improvement plans based upon recent assessments, including timelines.

We are going to modify the CSC 110 offering in an attempt to attract majors to our programs. Currently, the course is a simple introduction to computing and Office skills. We would like to focus more on the Software Engineering field in general and increase interest in our offerings. The timeline for this plan is to develop the new course over the 2008-09 school year, and by Fall 2009 have a new course in place.

Any changes in program educational objectives or program outcomes within the last six years.

We adopted Computer Science assessment processes for SCHEV between 2006 and 2007 to our Software Engineering program. When we began researching the requirements for ABET accreditation in late 2007, we realized we needed to
strengthen our assessment processes. The new changes to the assessment plan presented in this Self-Study began in the 2008-09 school year. They will undergo the first year of collection in the 2008-2009 school year. The program educational objectives and program outcomes were developed by modifying our existing SCHEV assessment plans and changing language to match that used by ABET.

Note that this is a new program and we do not yet have assessment data for our Program Educational Objectives. We do have a process in place and we feel the process should be adequate for assessing our level of attainment of our educational objectives.
CRITERION 5. CURRICULUM

A. Program Curriculum

1. Describe how students are prepared for a professional career and further study in the discipline through the curriculum and indicate how the curriculum is consistent with the Program Educational Objectives and Program Outcomes.

Recall that our program outcomes are:

PO1. An ability to apply an engineering approach to the development of software systems by analyzing, designing, implementing, verifying, validating and maintaining software systems.

PO2. An ability to exhibit technical competency as well as leadership and communication skills necessary to analyze, design, verify, validate, implement and maintain software systems.

PO3. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4. An ability to demonstrate the necessary communication skills to elicit, analyze and specify software requirements and to create well-written software documentation.

PO5. An ability to demonstrate the necessary organizational and business skills to work effectively in a team setting and to be able to predict the time and cost needed to create and to maintain software systems.

PO6. An ability to apply appropriate codes of ethics and professional conduct to the solution of software engineering problems.

PO7. An ability to demonstrate an awareness of their professional and social responsibility as software engineers through professionalism, continued learning and application of their knowledge for the good of society.

Our courses provide our students with the knowledge and skills necessary for covering our program outcomes. Almost all of the courses a graduate could take in completion of the degree requirements would provide coverage of our program outcomes 1, 2, and 4. Our introductory software engineering course, CSC 2300 uses extensive team based work to provide students with team experiences. The following table shows a breakdown of how our courses contribute to the outcomes.

The entries in the following table indicate:

- X – program outcome is addressed as a major part of the course
- S – program outcome is addressed somewhat in the course
<table>
<thead>
<tr>
<th>Course</th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General Computer Science</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSC 1180 Fundamentals of Programming</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>CSC 2300 Introduction to Software Engineering</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CSC 2180 Data Structures</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>CSC 3260 Introduction to Human - Computer Interaction</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSC 3400 Database Design</td>
<td>X</td>
<td>X</td>
<td>S</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSC 3710 Discrete Mathematics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CSC 3180 Introduction to Algorithms</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>CSC 4000 Operating Systems</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>CSC 4200 Programming Languages</td>
<td>X</td>
<td>S</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
</tr>
<tr>
<td>CSC 4300 Computer Architecture</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td><strong>Mathematics</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MTH 1180 Probability and Statistics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>MTH 2040 Calculus I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>MTH 2050 Calculus II</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>MTH 3060 Calculus III</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>MTH 3130 Intro to Linear Algebra</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td><strong>Science</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHY 3010/2010 – College Physics I/Lab</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>PHY 3020/2020 – College Physics II/Lab</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td><strong>Non-technical Support</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COM 3450 Technical Writing</td>
<td></td>
<td></td>
<td>S</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 1790 Engineering Leadership</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 2790 Engineering Economics</td>
<td></td>
<td></td>
<td>X</td>
<td>S</td>
<td>X</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td><strong>Software Engineering Core Requirement</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWE 2130 Software Construction Technology</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>SWE 2210 Testing, Verification and Validation</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>SWE 3210 Software Quality Assurance</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>SWE 3220 Software Requirements and Modeling</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>SWE 3230 Software Configuration</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

35
2. **Provide evidence that the minimum credit hours and distribution, as specified in Criterion 5, are met. Complete and include Table 5-1.**

See table 5-1

3. **Describe the culminating major design experience, including how it is based on the knowledge and skills acquired in earlier course work and how appropriate engineering standards and multiple realistic constraints are incorporated in the experience.**

Students work on the capstone project and apply different Software Engineering techniques learned during the program. Students work on real world group projects and start gathering requirements by meeting and interviewing the customer and learn about the customer needs. Once the requirements have been gathered then they analyzed it to make sure they are very specific and there is no guess work involved. After completing the design they work on the prototype and meet with the customer to make sure they are on the same page. After the approval from the customer they start implementing the project design. The project also goes through several testing stages such as Unit, Integration and System test. After the successful implementation and testing of the software system, the customer is invited to perform Acceptance test(s). During the course of the project the students works with the number of tools such as Subversion, ClearQuest, MS Project etc.

4. **Demonstrate that adequate time and attention are given to each curricular component, consistent with the outcomes and objectives of the program and the institution.**

By completing the degree requirements set forth by the Bachelor of Science in Software Engineering, our students meet or exceed ABET’s requirements as specified in the Software Engineering Program Criteria. Examination of Table 5-1 shows the number of hours per category area. Our program meets or exceeds ABET’s requirements.

These courses and the experiences therein adequately cover our program outcomes and objectives.

5. **Describe the provisions for any cooperative education that is used to satisfy curricular requirements.**

NA

6. **Describe the additional materials that will be available for review during the visit to demonstrate achievement related to this criterion, per section I.E.3.c of the ABET Accreditation Policy and Procedure Manual (APPM).**

- Sufficient examples of student work
- Range of grades (low, medium and high) for assignments, including homework, quizzes, examinations, drawings, laboratory reports, projects, and samples of computer usage in technical courses.
- Examples of written work
- Course outlines and textbooks

B. Prerequisite Flow Chart
Attach a flow chart showing the prerequisite structure of the program's courses required or allowed towards the major.
SE BS Degree Requirements

MTH 1110/1120 - 3hrs
  ↓
MTH 1180 - 4hrs
  ↓
MTH 2040 - 4hrs
  ↓
MTH 2050 - 4hrs
  ↓
MTH 3060 - 4hrs
  ↓
MTH 3110 - 3hrs
  ↓
CSC 3180 - 3hrs
  ↓
CSC 3260 - 3hrs
  ↓
CSC 4200 - 3hrs
  ↓
CSC 4980/4990 - 3hrs

CSC 1180 - 4hrs
  ↓
CSC 3120 & 2120 - 4hrs + 1 hr lab
  ↓
CSC 3120 & 2120 - 4hrs + 1 hr lab
  ↓
CSC 3130 - 3hrs
  ↓
CSC 3400 - 3hrs
  ↓
CSC 4000 - 3hrs
  ↓
CSC 4240 - 3hrs
  ↓
CSC 4980/4990 - 3hrs

CSC 2300, 2180
  ↓
SWE 2130
  ↓
SWE 2210 - 3hrs
  ↓
SWE 3210 - 3hrs
  ↓
SWE 3220, 3230 - 3hrs
  ↓
SWE 4980/4990 - 3hrs
C. Course Syllabi
In Appendix D, include a syllabus for each course used to satisfy the mathematics, science, and discipline-specific requirements required by Criterion 5 or any applicable Program Criteria. The syllabi format should be consistent for each course, must not exceed two pages per course, and, at a minimum, contain the following information:

- Department, course number, and title of course
- Designation as a Required or Elective course
- Course (catalog) description
- Prerequisites
- Textbook(s) and/or other required material
- Course learning outcomes
- Topics covered
- Class/laboratory schedule, i.e., number of sessions each week and duration of each session
- Contribution of course to meeting the requirements of Criterion 5
- Relationship of course to Program Outcomes
- Person(s) who prepared this description and date of preparation
<table>
<thead>
<tr>
<th>Year; Semester or Quarter</th>
<th>Course (Department, Number, Title)</th>
<th>Math &amp; Basic Sciences</th>
<th>Category (Credit Hours)</th>
<th>Engineering Topics Check if Contains Significant Design (✓)</th>
<th>General Education</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>First Semester Freshman Year</strong></td>
<td>ENG 1010 - Composition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>MTH 2040 – Calculus I</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Foreign Language 1</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Phys Ed</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSC 1810 – Fundamentals of Programming</td>
<td></td>
<td>4 ✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SEM 1010 – Freshman Seminar</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Second Semester Freshman Year</strong></td>
<td>CSC 2810 - Data Structures</td>
<td></td>
<td>3 ✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MTH 2050 – Calculus II</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ENG 1020 – Composition 2</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Foreign Language 2</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MTH 1180 – Probability and Statistics</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SEM 1020 – Freshman Seminar</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>First Semester Sophomore Year</strong></td>
<td>CSC 3810 – Algorithms</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MTH 3060 – Calculus III</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Sophomore Year</strong></td>
<td>CSC 2300– Software Engineering</td>
<td></td>
<td>3 ✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWE 2790 – Engineering Econ.</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Visual and Performing Arts</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWE 1790 Engineering Leadership</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5-1 Curriculum

Bachelor of Science in Software Engineering
<table>
<thead>
<tr>
<th>Year; Semester or Quarter</th>
<th>Course (Department, Number, Title)</th>
<th>Math &amp; Basic Sciences</th>
<th>Category (Credit Hours)</th>
<th>Engineering Topics &lt;br&gt;Check if &lt;br&gt;Contains Significant &lt;br&gt;Design (✔)</th>
<th>General Education</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Second Semester</strong></td>
<td>CSC 3710 – Discrete Structures</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWE 2130 – Software Construction</td>
<td>3 ✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWE 2210 – Testing and Validation</td>
<td>3 ✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Sophomore Year</strong></td>
<td>COM 3450 – Technical Writing</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Social Science</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Visual and Performing Arts</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSC 3260 – HCI</td>
<td>3 ✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>First Semester</strong></td>
<td>SWE 3210 – Quality Assurance</td>
<td>3 ✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PHY 3010/2010 – Physics with Lab</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Junior Year</strong></td>
<td>Western Heritage 1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Social Science</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWE 3220 – Requirements and Modeling</td>
<td>3 ✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Second Semester</strong></td>
<td>SWE 3230 – Software Configuration Management</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PHY 3020/2020 – Physics II with Lab</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Junior Year</strong></td>
<td>Literature</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Western Heritage 2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

42
<table>
<thead>
<tr>
<th>Year; Semester or Quarter</th>
<th>Course (Department, Number, Title)</th>
<th>MTH &amp; Basic Sciences</th>
<th>Engineering Topics Check if Contains Significant Design (✓)</th>
<th>General Education</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Semester</td>
<td>CSC 3400 – Database design</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWE 4240 – Project management</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Senior Year</td>
<td>CSC 4300 – Computer Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Social Science</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>SWE 4980 Capstone Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Visual and Performing Arts</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>CSC 4000 – Operating Systems</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Second Semester</td>
<td>CSC 4200 – Programming Languages</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>MTH 3130 – Linear Algebra</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Senior Year</td>
<td>Humanities</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>SWE 4990 Capstone Project</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

Add rows as needed to show all courses in the curriculum.

<table>
<thead>
<tr>
<th>TOTALS-ABET BASIC-LEVEL REQUIREMENTS</th>
<th>34</th>
<th>55</th>
<th>39*</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVERALL TOTAL FOR DEGREE</td>
<td>131</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PERCENT OF TOTAL</td>
<td>26%</td>
<td>42%</td>
<td>30%</td>
<td>2%</td>
</tr>
<tr>
<td>Totals must satisfy one set</td>
<td>Minimum semester credit hours</td>
<td>32 hrs</td>
<td>48 hrs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Minimum percentage</td>
<td>25%</td>
<td>37.5 %</td>
<td></td>
</tr>
</tbody>
</table>

Note that instructional material and student work verifying course compliance with ABET criteria for the categories indicated above will be required during the campus visit.

*College General Education requirements include 14 hours of Math & Science, counted in column 1, for a total of 53 hours.
Table 5-2. Course and Section Size Summary

Bachelor of Science in Software Engineering

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Title</th>
<th>Responsible Faculty Member</th>
<th>No. of Sections Offered in Current Year</th>
<th>Avg. Section Enrollment</th>
<th>Lecture&lt;sup&gt;1&lt;/sup&gt;</th>
<th>Lab&lt;sup&gt;1&lt;/sup&gt;</th>
<th>Other&lt;sup&gt;1&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWE 1790</td>
<td>Engineering Leadership</td>
<td>Steve Brooks*</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 2790</td>
<td>Engineering Economics</td>
<td>Bill Skeen*</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 2130</td>
<td>Software Construction Technology</td>
<td>Abrar Qureshi/Robert Hatch</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 2210</td>
<td>Testing, Verification and Validation</td>
<td>Abrar Qureshi</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 3210</td>
<td>Software Quality Assurance</td>
<td>Abrar Qureshi/Robert Hatch</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 3220</td>
<td>Software Requirements and Modeling</td>
<td>Robert Hatch</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 3230</td>
<td>Software Configuration Management</td>
<td>Abrar Qureshi</td>
<td>1</td>
<td>4</td>
<td>80%</td>
<td>20%</td>
<td></td>
</tr>
<tr>
<td>SWE 4240</td>
<td>Software Project Management</td>
<td>Abrar Qureshi</td>
<td>1</td>
<td>4</td>
<td>70%</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td>SWE 4980</td>
<td>Capstone Project</td>
<td>Abrar Qureshi</td>
<td>1</td>
<td>4</td>
<td>10%</td>
<td>90%</td>
<td></td>
</tr>
<tr>
<td>SWE 4990</td>
<td>Capstone Project</td>
<td>Abrar Qureshi</td>
<td>1</td>
<td>4</td>
<td>10%</td>
<td>90%</td>
<td></td>
</tr>
</tbody>
</table>

<sup>1</sup> Enter the appropriate percent for each type of class for each course (e.g., 75% lecture, 25% laboratory).

* adjunct faculty
CRITERION 6. FACULTY

A. Leadership Responsibilities
Identify the person who has leadership responsibilities for the program. Describe the leadership and management responsibilities of that person.

Alex Edwards is the Department Chair. In addition to being a full time faculty member, the Chair also oversees departmental operations and manages degree program offerings.

He oversees purchasing and has authority over budget expenditures.

He supports faculty scholarship through both financial and professional means.

B. Authority and Responsibility of Faculty
Describe the role played by the program faculty with respect to course creation, modification, and evaluation. Describe the roles played by others on the campus, e.g., Dean’s Office, Provost’s Office, with respect to these areas. Describe the process used to ensure consistency and quality of the courses taught.

Our faculty members have control over the development and evaluation of our program; with oversight for program changes from the Academic Program Committee (through the Faculty Senate). Major changes to programs (e.g. adding new courses, significant content changes, adding new programs) must be recommended for approval by this committee and approved by the Faculty Senate. The evaluation of the program, determination/recognition of necessary change, and what is proposed as a change are solely determined by the Faculty.

The Provost’s office orchestrates handling of student evaluation of courses at the end of each semester. Students anonymously report on the course content, resources, and instructor. See Appendix E for an example evaluation form. Recently these evaluations have been conducted for every single course, for every faculty member. This number has changed to all courses for tenure track faculty and to 2 courses per semester for tenured faculty. The faculty member gets a copy of the evaluation and decides success of course, direction of course, and any other changes in consultation with the Department Chair.

We use information from the Departmental comprehensive exam and internal discussions of detected weaknesses in student performance to decide on content or curriculum changes (closing the loop on assessment). For example, if students seem weak in a specific area (as evidenced through assessment analysis), we might modify a course’s pre-requisite structure to increase student preparedness.
In terms of consistency, our full time faculty members teach all of our course offerings for the Software Engineering degree. In the rare instances in which an adjunct faculty member is needed, the oversight of that course offering is handled by the department Chair. If there are multiple sections of single course, one instructor will typically teach all sections of that course. This ensures the course offerings are the same for all sections and provides fewer preparations for the faculty member.

C. Faculty
Describe the composition, size, credentials, experience, and workload of the faculty that supports this program. Complete and include Tables 6-1 and 6-2.

Our teaching load is 12 semester hours (4 classes). This heavy load limits the amount and types of scholarly activity that our faculty can perform. Regardless, faculty show excellent scholarship records, with our faculty members frequently attending academic conferences and workshops. This is evident in the faculty profiles provided earlier.

In addition to the 12 semester hour teaching load, faculty members serve on committees, both for the College and for the Faculty Senate. Typically a faculty member will serve on one or two committees but can occasionally serve on more. Also we have two departmental representatives on the Faculty Senate. These faculty senate duties change every three years.

Our faculty members provide advising to our majors for the three degree programs that we offer in our department (Computer Science, Software Engineering, and Management Information Systems). Individual faculty can choose to participate in the Freshman Advising program (mentioned earlier in the Student section) and can have upwards of 30 advisees at any given time. Fortunately, our students know the degree requirements and can make sound decisions about scheduling without extensive faculty input. Individual or one-on-one advising is available but not required for the students. Students typically seek an appointment with his/her advisor for specific questions about courses and offerings and to verify schedules.

Typically, time for faculty development and scholarship is found during the summer break or through faculty leave options. Faculty leave is determined by the Faculty Development committee of the Faculty Senate. A person seeking leave time must apply for a course reduction to that Committee with approval from the Vice Provost. Of course the highly motivated and talented faculty member can do scholarship in conjunction with his/her other duties throughout the school year.

D. Faculty Competencies
Describe the competencies of the faculty and how they are adequate to cover all of the curricular areas of the program.

Degrees in the related field
We have 5 full-time faculty members devoted to the Software Engineering program. As much of the core comes from Computer Science, four of those members cover Computer Science courses. Four of the five faculty members hold PhD’s in either Computer Science or Computer Engineering. One of the faculty members has a Master’s Degree in Software Engineering. We feel these advanced degrees more than cover our curricular areas.

In addition to the full time faculty members, we periodically use 2 adjunct faculty members to help with SWE 1790 – Engineering Leadership and SWE 2790 – Engineering Economics. In these instances, the Adjuncts will have at least a Master’s Degree in an appropriate field.

Scholarship/Research in the respective field

Our faculty members pursue a wide range of research topics: software engineering, quality assurance, interface design and testing, security, and artificial intelligence. These areas provide adequate support for our curricular areas.

Professional experience in the field of Software Engineering

Some of our faculty members have extensive industry experience in Software Engineering. Dr. Qureshi has 12 years of professional experience, 10 years at General Electric (GE) where he worked in Quality Assurance.

E. Faculty Size
Discuss the adequacy of the size of the faculty and describe the extent and quality of faculty involvement in interactions with students, student advising, service activities, and professional development.

We feel we have enough faculty members to support our program. We have no problems offering the required courses for our degrees and also provide support to our students.

Small class sizes (< 20 students) allow our faculty to learn students’ names and get to know them on a personal level. All faculty hold at least 8 office hours per week (a College policy) and often have their doors open for students when they are on campus. Each course provides ample opportunity for interaction between students and faculty; both through in-class activities and outside-class assignments.

When incoming students first sign-up for classes, they are advised by faculty from the department in which they are interested. Our faculty members share this responsibility by taking turns at the four freshman-orientation sessions held by the Admissions Office as well as the transfer student orientation sessions.

All freshmen are required to take a 2-semester sequence in which they are assigned a temporary advisor. This advisor is a faculty member at the college but students who have a particular interest are paired with a faculty member from that target department. When students declare a specific
major, they are assigned a permanent advisor (a faculty member in the target department) by the Department Chair. Students can access their advisor during normal school hours, typically 8-4 M-F. Usually students drop by during faculty office hours but they can also make appointments.

Faculty members serve on various committees for the College and the Department. These duties do not start until the second year of employment.

F. Faculty
In Appendix B include an abbreviated resume for each program faculty member with the rank of instructor or above. The format should be consistent for each resume, must not exceed two pages per person, and, at a minimum, must contain the following information:

- Name and academic rank
- Degrees with fields, institution, and date
- Number of years of service on this faculty, including date of original appointment and dates of advancement in rank
- Other related experience, i.e., teaching, industrial, etc.
- Consulting, patents, etc.
- States in which professionally licensed or certified, if applicable
- Principal publications of the last five years
- Scientific and professional societies of which a member
- Honors and awards
- Institutional and professional service in the last five years
- Percentage of time available for research or scholarly activities
- Percentage of time committed to the program

F. Faculty Development
Describe the plan that is in place for faculty development and the funding available to execute this plan. Provide detailed descriptions of professional development activities for each faculty member.

“The College expects faculty to engage continually in scholarship in support of teaching” [2008-2009 Faculty Handbook, page 35]. For purposes of evaluation, faculty members are expected to spend 20-
25% of their time in scholarly pursuits. The College supports scholarship with several mechanisms, including:

- Faculty Travel: The provost’s office maintains a travel budget through which any faculty member may seek funding. The amount varies with budgetary fluctuations, but generally guarantees each faculty member enough funds to attend at least one conference per year.

- Faculty Development Committee: Faculty members may apply to this committee for the following: mini-leaves, summer research, professional travel, and general research.

- Sabbaticals: Faculty can apply for leave options through the Academic Dean.

- Departmental Budget: Departments may elect to use some portion of their budgets to augment any of the above mechanisms.

- Foundation: There is an active charitable organization associated with the college that has received sizable awards that directly benefit the Department and our Software Engineering degree.

These faculty activities are recognized by the College at an annual Faculty Awards ceremony.

In addition, our department has exceptional collegiality. Faculty members often seek advice and feedback on teaching and scholarship through an open exchange of ideas. Department members regularly socialize. Some members participate in intramurals and other student activities. Our department members hold respected offices in various committee structures.

**Attendance at professional meetings**

<table>
<thead>
<tr>
<th>Meeting</th>
<th>Attendees</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIKES (Living In a KnowLEdge Society) October 2008, Virginia Tech</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td></td>
<td>Daniel Ray</td>
</tr>
<tr>
<td></td>
<td>Robert Hatch</td>
</tr>
<tr>
<td>ABET Best Assessment Practices Symposium April 2008, Atlanta, GA</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td></td>
<td>Daniel Ray</td>
</tr>
<tr>
<td>Third WG 11.9 International Conference on Digital Forensics (Presenter), January 2007</td>
<td>Daniel Ray</td>
</tr>
<tr>
<td>Cyber Security and Information Infrastructure Research</td>
<td>Daniel Ray</td>
</tr>
<tr>
<td>Event</td>
<td>Presenter(s)</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Workshop (Presenter) – May 2008</td>
<td>Daniel Ray</td>
</tr>
<tr>
<td>The 2008 World Congress in Computer Science, Computer Engineering,</td>
<td>Daniel Ray</td>
</tr>
<tr>
<td>and Applied Computing (Presenter) – July 2008</td>
<td></td>
</tr>
<tr>
<td>IEEE Tri-Cities Section meeting (Presenter) – April 2007</td>
<td>Abrar Qureshi</td>
</tr>
<tr>
<td>Virginia IT Workforce Summit and shared ideas about what government,</td>
<td>Abrar Qureshi</td>
</tr>
<tr>
<td>education, and industry can do to increase the Shrinking IT Pipeline.</td>
<td>Daniel Ray</td>
</tr>
<tr>
<td>2007</td>
<td>Alex Edwards</td>
</tr>
<tr>
<td>2007</td>
<td></td>
</tr>
<tr>
<td>ABET faculty workshop on assessing program outcomes held in Tampa to</td>
<td>Abrar Qureshi</td>
</tr>
<tr>
<td>get our CS and SE program accredited.</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td>University of Virginia’s All-University Retreat, Science and Technology at U.Va. Discussed science, technology, and research related issues with President Casteen, Provost Tim Garson, Vice President for Research Tom Skalak and U. Va faculty in general. Fall 2008</td>
<td>Abrar Qureshi</td>
</tr>
<tr>
<td>3rd International Conference on Sociology in Athens, Greece –</td>
<td>Abrar Qureshi</td>
</tr>
<tr>
<td>Presented a paper on Social Software and Its Impact on our Society –</td>
<td></td>
</tr>
<tr>
<td>May 2009</td>
<td></td>
</tr>
<tr>
<td>2004 Conference on Frontiers in Education (FIE’04) – presented a</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td>paper on case based teaching of HCI concepts</td>
<td></td>
</tr>
<tr>
<td>Human Factors and Ergonomics Society 48th annual meeting (HFES’04) –</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td>presented a paper on heuristic creation methods</td>
<td></td>
</tr>
<tr>
<td>2006 International Conference on Frontiers in Education: Computer</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td>Science and Computer Engineering (FECS’06) – presented a paper on</td>
<td></td>
</tr>
<tr>
<td>pair programming in CS1 courses</td>
<td></td>
</tr>
<tr>
<td>2007 High Performance Computing Bootcamp (HPCB) at The University of \</td>
<td>Jacob Somervell</td>
</tr>
<tr>
<td>Virginia – attended</td>
<td></td>
</tr>
</tbody>
</table>
Publications (conference and journals)


**Development of courses**

Web Technologies – CSC 3750 created in 2005, first offered in 2005

Digital Logic – CSC 3050 created in 2008, first offered in 2009


Summer Program In Robotics and Intelligent Technology (SPIRIT) – A summer program in robotics targeting high school students. The goal is to instill curiosity and interest in careers in Technology fields with an ultimate goal of increasing enrollments in our programs through outreach. Developed over the 2008-2009 academic year, first implemented in Summer of 2009.
### Table 6-1. Faculty Workload Summary

**Bachelor of Science in Software Engineering**

<table>
<thead>
<tr>
<th>Faculty Member (name)</th>
<th>FT or PT</th>
<th>Classes Taught (Course No./Credit Hrs.)</th>
<th>Fall 2008 – Spring 2009</th>
<th>Total Activity Distribution $^2$</th>
</tr>
</thead>
</table>
| Alex Edwards (Dept Chair) | FT | Fall 08: CSC 1010, CSC 3250, CSC 4350  
SP 09: CSC 1100, CSC 4000, CSC 4110, CSC 4990/ MIS 4980 | | 50 | 15 | 35 |
| Jacob Somervell | FT | Fall 08: CSC 2220, CSC 2300, CSC 2180, CSC 3710  
SP 09: CSC 2300, CSC 2180, CSC 3260, CSC 4200 | | 60 | 20 | 20 |
| Abrar Qureshi | FT | Fall 08: CSC 110, HONR 3950, SWE 4240, SWE 4980  
SP 09: CSC 4380, CSC 110, SWE 3230, SWE 2210, SWE 4990 | | 60 | 20 | 20 |
| Danial Ray | FT | Fall 08: CSC 1180, CSC 1180, CSC 3400, CSC 3180  
SP 09: CSC 1180, CSC 2230, CSC 4150, CSC 3180 | | 60 | 20 | 20 |
| Robert Hatch | FT | Fall 08: CSC 1100, CSC 1100, CSC 4300, SWE 3210  
SP 09: CSC 1100, CSC 3750, SWE 2130, SWE 3220 | | 60 | 20 | 20 |
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Term</th>
<th>Activity %</th>
<th>Others</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steve Brooks</td>
<td>PT</td>
<td>Fall 08: SWE 1790</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bill Skeen</td>
<td>PT</td>
<td>Fall 08: SWE 2790</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

1. Indicate Term and Year for which data apply (the academic year preceding the visit).
2. Activity distribution should be in percent of effort. Members’ activities should total 100%.
3. Indicate sabbatical leave, etc., under "Other."
4. FT = Full Time Faculty    PT = Part Time Faculty
<table>
<thead>
<tr>
<th>Name</th>
<th>Rank</th>
<th>Type of Academic Appointment</th>
<th>FT or PT</th>
<th>Highest Degree and Field</th>
<th>Institution from which Highest Degree Earned &amp; Year</th>
<th>Years of Experience</th>
<th>Years of Experience</th>
<th>Level of Activity (high, med, low, none) in:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Govt./Industry Practice</td>
</tr>
<tr>
<td>Alex Edwards</td>
<td>AP</td>
<td>T</td>
<td>FT</td>
<td>MCS</td>
<td>UVA 1982</td>
<td>4</td>
<td>22</td>
<td>23</td>
</tr>
<tr>
<td>Jacob Somervell</td>
<td>AP</td>
<td>TT</td>
<td>FT</td>
<td>Ph.D. Computer Science</td>
<td>Virginia Tech 2004</td>
<td>5</td>
<td>5</td>
<td>N</td>
</tr>
<tr>
<td>Abrar Qureshi</td>
<td>AP</td>
<td>TT</td>
<td>FT</td>
<td>Ph.D. Computer Engineering</td>
<td>Florida Tech 2006</td>
<td>12</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Danial Ray</td>
<td>AP</td>
<td>TT</td>
<td>FT</td>
<td>Ph.D. Computer Science</td>
<td>The University of Alabama 2007</td>
<td>2</td>
<td>2</td>
<td>H</td>
</tr>
<tr>
<td>Robert Hatch</td>
<td>AP</td>
<td>TT</td>
<td>FT</td>
<td>Ph.D. Computer Science</td>
<td>The University of Alabama 2008</td>
<td>1</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>Steve Brooks</td>
<td>Ins</td>
<td>NTT</td>
<td>PT</td>
<td>MS Civil Engineering</td>
<td>Virginia Tech 1989</td>
<td>20</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>Name</td>
<td>Ins</td>
<td>NTT</td>
<td>PT</td>
<td>MS Engineering</td>
<td>VTech</td>
<td>Year</td>
<td>Active 1</td>
<td>Active 2</td>
</tr>
<tr>
<td>------------</td>
<td>-----</td>
<td>-----</td>
<td>----</td>
<td>----------------</td>
<td>-------</td>
<td>------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>Bill Skeen</td>
<td></td>
<td></td>
<td></td>
<td>Sanitary</td>
<td>1975</td>
<td>35</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Instructions: Complete table for each member of the program faculty. Use additional sheets if necessary. Updated information is to be provided at the time of the visit. The level of activity should reflect an average over the year prior to visit plus the two previous years.

Column 3 Code: TT = Tenure Track  T = Tenured  NTT = Non Tenure Track
CRITERION 7. FACILITIES

A. Space
Summarize the availability of program facilities and indicate how adequate they are for supporting the educational objectives and outcomes of the program. Discuss the following.

1. Offices (Administrative, Faculty, Clerical, Teaching Assistants)

Faculty and staff members have their own office space, either in a single room or within a cubicle provided in Darden Hall. Each office has at least 1 of each of the following:

- desk
- computer with Internet access
- book shelf
- chair
- printer
- filing cabinet

These office spaces are adequate for meeting the faculty needs. They provide a known location for holding office hours. The offices provide the faculty member with an area for scholarship, teaching, and advising. Anecdotal evidence suggests that faculty members are satisfied with the office spaces as there are no complaints. Furthermore, any faculty needs (in terms of equipment and supplies) are provided in a timely fashion.

2. Classrooms

Most classrooms contain one computer, one monitor, one ceiling-mount projector, one speaker system w/amplifier, one DVD/VCR combo player, and one projector screen. Classrooms also contain laptop connections for instructor laptops. Some classrooms contain a SMART Board, and all classrooms contain whiteboards and/or chalkboards.

The equipment in these classrooms is adequate for our teaching needs. At a bare minimum, the computers need Internet browsing and Microsoft Office capabilities, and all classroom computers meet those requirements. Computer lab facilities are used as classrooms when needed (for example, when special software is needed for the course).

3. Laboratories

Institutional and college computing facilities:
College facilities contain various numbers of Dell computers, ranging from Optiplex GX270 models to Optiplex 740 models. Most lab computers use 17” LCD monitors. Each computer is equipped with a mouse and keyboard, and some facilities provide headphones for patron use. College lab computers run Microsoft Windows XP Professional, with the exception of a handful of Macintosh computers running Mac OS X. All college facilities provide Microsoft Office 2007 Professional. Other labs make use of software as needed by certain departments. These software packages include Adobe Creative Suite and SPSS.

All campus computers (with the exception of a special Computer Science lab, will be detailed in next subsection) are authenticated via Microsoft Windows Active Directory Server, running on Windows Server 2003 (domain based). Students receive an account during freshman orientation. Faculty and staff receive accounts after date of hire. The campus is divided into various subnets, depending on building. Each residence hall, classroom building, and administrative building is on its own subnet. Each subnet is connected to the main campus switches via at least a 1Gb connection, with some subnet switches having up to 10Gb connections. The college’s Internet connection to its ISP is 100Mb.

Each domain account is provided with a network share. This share is mapped to the user’s My Documents folder and is also mapped as a network drive. Occasionally these shares encounter permissions problems, and the owners of the shares are denied access. This can cause problems for users that have documents saved on their share space. This problem is easily remedied, but is usually remedied on a case-by-case basis. As of this writing, the college’s IT department is working to resolve this apparently random issue.

Sometimes problems can arise for instructors who wish to take advantage of remote desktop capabilities (using their office computer for classroom demonstration) with the college’s network security policies. The instructor must contact the campus IT department to request modifications to the intranet firewall policies if such issues occur.

**Departmental computing facilities:**

The Department of Mathematics and Computer Science maintains two computer labs. One lab is open to public use, and the other lab is open only to MIS/CS/SE majors.

The public lab contains 36 patron computers and 1 teaching workstation. It is equipped with 2 monochrome laser printers, 1 color laser printer, a sound system w/amplifier, a projector screen, a ceiling-mount projector, a DVD/VCR player, and a flatbed scanner. These computers are a mix of Dell Optiplex GX280 models and Optiplex 620 models running Windows XP. Software installed includes Office 2007 Professional, Mathematica, MATLAB, Minitab, and Visual Studio 2008. The computers in this lab are joined to the college’s Active Directory domain.

The private lab consists of 14 patron computers and 1 teaching workstation. This lab does not contain a projector. Instead, the teaching workstation is connected to a 42” plasma TV. The lab is equipped with 1 monochrome laser printer, 1 color laser printer, and a flatbed scanner. The computers in this lab are Dell Optiplex 740 models. Each computer is set up to dual boot. Users can boot to either Windows XP
or Linux (Fedora 9). This lab is not joined to the campus Active Directory domain. Instead, users are authenticated via OpenLDAP and Samba running on the department’s own servers. Students must request accounts in person with the departmental systems administrator, who then approves or denies accounts based on student eligibility. Students are granted power user privileges in this lab, whereas all Active Directory joined campus computers grant students only user privileges. Software in this lab includes Office 2007, Mathematica, MATLAB, Minitab, Visual Studio 2008, Adobe Creative Suite 3, LabVIEW, and various open source programs and utilities, such as Netbeans, Eclipse, and Geany.

B. Resources and Support

1. Describe the computing resources, hardware and software used for instruction. Specify any limitations that impact the student’s ability to achieve the program’s outcomes and the faculty’s teaching and scholarly activities.

See previous question. We feel our resources are adequate to support our program.

2. Describe the laboratory equipment planning, acquisition, and maintenance processes and their adequacy.

University-wide computing resources are usually upgraded or replaced every 3-5 years. The college receives new equipment via an Equipment Trust Fund (ETF) provided by the Commonwealth of Virginia. Equipment planning and acquisition is handled by the Director of Technology Support Services. Automatic OS updates and software deployment are handled by SMS, and antivirus software is managed by a Symantec Endpoint Protection server. Imaging of laboratory computers is done on a semester-by-semester basis.

Laboratory equipment planning for departmental facilities is a joint effort between the systems administrator and the department chair, with the department chair having the final say. The chair consults the department’s faculty, and then discusses equipment planning with the systems administrator. Once decisions are made, the systems administrator is responsible for the acquisition of equipment. Departmental lab computers are imaged at the beginning of the fall, spring, and summer I semesters. These images contain the latest software updates and fixes. Lab computers are imaged during semesters as needed. Lab computers are typically replaced when their warranty expires. The college has a contract with Dell, and that contract includes an extended warranty for all computers. Computers are covered by the warranty for a total of 4 years.

These processes have proven adequate thus far.

3. Describe the type and number of support personnel available to install, maintain, and manage departmental hardware, software, and networks.

Support personnel on the college level consists of the following subgroups within the campus IT department: Technology Support Services, which consists of the Director of Technology Support Services, the Technology Support Manager, a Helpdesk Technician, a Desktop Support Technician, and
student work studies; Network Operations, which consists of a Senior Network Systems Engineer and an Educational Systems Engineer; Systems and Security, which consists of a Senior Systems and Security Administrator and a Web Systems Specialist; Media Services, which consists of the Director of Media Services and a Media Services Production Assistant. The only limitation for the level of support appears to be the number of personnel available. Otherwise, support is adequate.

Support personnel on the departmental level include a Computer Systems Administrator, a Webmaster/Systems Support Specialist, and an Office Support Specialist. These people work closely with our faculty members to meet any needs as they arise. Support is superb and we couldn’t ask for better people.

4. Describe the type and number of support personnel available to install, maintain, and manage laboratory equipment.

The personnel described in the previous question perform these duties as well.

C. Major Instructional and Laboratory Equipment

In Appendix C, include a list of major instructional and laboratory equipment.
CRITERION 8. SUPPORT

A. Program Budget Process and Sources of Financial Support
Describe the process used to establish the program budget and provide evidence of continuity of institutional support for the program.

Department budgets are determined at the Chancellor/Vice Chancellor level. The Department of Mathematics and Computer Science, through the department chair, communicates budget concerns and information about areas of need to the Provost and Senior Vice Chancellor as concerns arise. We are also presented with several formal opportunities throughout the year to express budget needs to the Provost. The Provost is then our representative in the budget process.

Despite recent state budget issues, our department has the full support of the college. Since the 1980’s there had been only 2 faculty members in Computer Science (which used to be called Computer Information Systems). A new faculty member was added in 2004. The State set aside 800,000 per year for five years to start the new Software Engineering program and the Computer Science program benefited as well. Two Software Engineers, one Computer Scientist, one Mathematician, and one Physicist were hired between 2005 and 2008. We have an ongoing search for one of the Software Engineering positions. Four staff positions were allocated, two and a half positions have been filled. ETF (equipment trust fund) money is used for updated lab equipment (3 year rotation -- approximately 1/3rd of the machines are replaced per year). Faculty computers are replaced on a 3-4 year cycle.

Northrop Grumman gave a large gift to the Department of Mathematics and Computer Science. Local governments and private donors have given significant gifts to help our programs. Clearly the college and the region have made a commitment to the success of our programs.

B. Sources of Financial Support
Describe the sources of financial support including both “hard” and “soft” monies.

Hard:
Department Budget – separate budgets for (1) department operations, (2) lab support, and (3) software engineering support
Equipment Trust Fund (ETF) – used to purchase computers and other hardware
Provost’s Office – travel, mini-leave
Faculty Development Committee – additional travel, other support

Soft:
Foundation – donations, > 1 million over last 5 years to support Computer Science and Software Engineering
We feel we have adequate resources to allow our program to achieve its educational objectives and outcomes. Some specific resources include:

New lab space: A private lab space for use by the Computer Science and Software Engineering programs will be available in the renovated Science Center. This lab will provide equipment and space for students to perform research activities. The space is divided into two separate labs: a visualization lab and a "sandbox" lab.

A dedicated server room and additional office space for a technician will be available in the renovated Science Center. The server room will house Departmental servers, Beowulf cluster, and other equipment.

Robotics Equipment: Twelve Roomba robots, with special processors and wireless capabilities, are available for use in research and course projects by students and faculty.

Beowulf Cluster: A 64-node cluster is available for research and course projects by students and faculty.

These resources are renewed through two main sources: ETF and an endowment fund. The ETF (equipment trust fund) is a state budget item designed to replace aging computers and other hardware. This money is used to periodically update computers in labs and faculty offices. This occurs on a 3-4 year cycle.

C. Adequacy of Budget
Describe the adequacy of the budget.

We believe all faculty requests and needs are being met. Faculty members have what they need for teaching, research, and other scholarly activities; hence the budget is adequate.

D. Support of Faculty Professional Development
Describe the adequacy of support for faculty professional development, how such activities are planned, and how they are supported.

Faculty members identify relevant conferences and workshops for themselves and others in the department. The faculty member and Department Chair determine appropriate funding sources from one or more of the following:

- Faculty Travel: The provost’s office maintains a travel budget through which any faculty member may seek funding. The amount varies with budgetary fluctuations, but generally guarantees each faculty member enough funds to attend at least one conference per year.
- Faculty Development Committee: Faculty members may apply to this committee for the following: mini-leaves, summer research, professional travel, and general research.
- Sabbaticals: Faculty can apply for leave options through the Academic Dean.
- Departmental Budget: Departments may elect to use some portion of their budgets to augment any of the above mechanisms.
E. Support of Facilities and Equipment
Describe the sufficiency of resources to acquire, maintain, and operate facilities and equipment appropriate for the program.

See section B. Sources of Financial Support above for descriptions of sources. We feel these resources are sufficient and appropriate for our program.

F. Adequacy of Support Personnel and Institutional Services
Describe the adequacy of support personnel and institutional services necessary to meet program needs.

See section 7.B.3 for a description of our support personnel. We feel these personnel are adequate to meet our program needs.
CRITERION 9. PROGRAM CRITERIA
Describe how the program satisfies any applicable Program Criteria. If already covered elsewhere in the Self-Study Report, provide appropriate references.

The curriculum must provide both breadth and depth across the range of engineering and computer science topics implied by the title and objectives of the program.

See CRITERION 5. CURRICULUM for details on course content.

These are specific criteria for the Software Engineering program.

The program must demonstrate that graduates have:

- the ability to analyze, design, verify, validate, implement, apply, and maintain software systems;
  This is demonstrated through the senior capstone experience (SWE 4980/4990) and a departmentally developed rubric.

- the ability to appropriately apply discrete mathematics, probability and statistics, and relevant topics in computer science and supporting disciplines to complex software systems;
  This is demonstrated through adequate performance on the Departmental Examination.

- and the ability to work in one or more significant application domains.
  This is demonstrated through the senior capstone experience (SWE 4980/4990) and a departmentally developed rubric.

Capstone Project
The goal of the Capstone is to provide students with a realistic software development experience that utilizes, as much as possible, the skills and knowledge acquired during the first three years of the software engineering program. Students work in teams since most real world projects are far too complex, and require too many different skills for any single person to manage on their own. All the projects involve real world software engineering practices. Each project has a customer and students continually interact with the customer on a weekly or bi-weekly basis when requirements are being gathered and refined and when prototypes are being evaluated. A project goes through different stages of development such as requirements analysis, design, implementation, testing and deployment. Students also have to use project management skills to estimate costs and plan schedules. The Capstone provides an opportunity for students to exercise the skills and knowledge they have acquired over the previous years. It also serves as a significant confidence building exercise. The Capstone instructor works with potential customers in scoping their concept so it can be successfully taken through the development process by a student team during the entire Capstone experience. All of these methods of showing that these specific program criteria are met can be found earlier in this document, specifically under Criterion 4 – Continuous Improvement and Criterion 5 – Curriculum.
APPENDIX A – Assessment Rubrics and Survey Instruments

Employer Survey

Employer Survey
Department of Mathematics and Computer Science
The University of Virginia’s College at Wise (UVA-WISE)

Date: ________________  Your Name: __________________________

Approximate number of Uva-Wise graduates whom you have worked with in the past 3 years: __________________________

In answering the following questions, compare our graduates with other employees who have similar degrees, if possible.

<table>
<thead>
<tr>
<th>How well do the UVA WISE graduates…</th>
<th>Exceptional</th>
<th>Above Average</th>
<th>Average</th>
<th>Below Average</th>
<th>Poor</th>
<th>Not Observed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Analyze tradeoffs and design appropriate solutions for complex problems?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Effectively communicate through written and oral means?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Seek or show the desire to pursue training or education to advance his/her self in their field?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Solve problems in a team environment while exhibiting professionalism and responsible, ethical behaviour.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Effectively analyze, design, and develop high-quality software systems using the appropriate theory, principles, tools and processes.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments?

Please return this completed form or fax to:
Jacob Somervell
221 Darden Hall
1 College Ave
Wise, VA 24293
FAX: 276-378-4589
Alumni Survey

Department of Mathematics and Computer Science
The University of Virginia’s College at Wise (Uva-Wise)

Date: ____________________ Your Name (optional) ____________________________________

In what year did you graduate? ____________________________________________________

Who is your current employer (if pursuing a degree, which school)?

List any advanced degrees, certificates, or professional memberships

List your job title and give a description of your typical duties.

For the following statements, rank how well your degree helped you...

1. Pursue careers or advanced study in your field.
   - strongly agree
   - neutral
   - disagree
   - strongly disagree

2. Effectively communicate with your peers, customers, supervisors, etc. through written and oral means.
   - strongly agree
   - neutral
   - disagree
   - strongly disagree

   - strongly agree
   - neutral
   - disagree
   - strongly disagree

4. Analyze and assess problems/situations and provide/suggest appropriate solutions.
   - strongly agree
   - neutral
   - disagree
   - strongly disagree

Please give us any specific comments about achievements you’ve had in any of the above areas.

Please give any specific comments about ways your degree could do a better job preparing you for the above areas.

Return to: Jacob Somervell, 221 Darden Hall, 1 College Ave, Wise, VA 24230
email  js54@uwwise.edu
fax  276-376-4699
Oral Communication Rubric
Department of Mathematics and Computer Science

Oral communication is defined as the ability to convey ideas/information in a fashion that is clear, ordered, and well-supported; that reflects the ability of the speaker to respond to the audience as well as to make a prepared statement; and to employ for the purpose a style that is appropriate to the occasion.

Student_____________________________  Presentation Date_____________________
Evaluator_____________________________  Average Score_____________________

**Oral Communication Skills**

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Excellent</td>
</tr>
<tr>
<td>4</td>
<td>Good</td>
</tr>
<tr>
<td>3</td>
<td>Average</td>
</tr>
<tr>
<td>2</td>
<td>Below Average</td>
</tr>
<tr>
<td>1</td>
<td>Far Below Average</td>
</tr>
</tbody>
</table>

**A. Ability to develop the main results in a clear manner.**

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Overall intent is unmistakable; audience has compelling reason to listen; speaker’s credibility is explicitly stated or clear</td>
</tr>
<tr>
<td>4</td>
<td>Overall intent clear; audience reason to listen clear; speaker credibility good</td>
</tr>
<tr>
<td>3</td>
<td>A topic is introduced; audience reason to listen may be vague or unclear; credibility is identified</td>
</tr>
<tr>
<td>2</td>
<td>Topic/intent vague; speaker’s credibility unclear; speaker credibility unclear</td>
</tr>
<tr>
<td>1</td>
<td>Intent of presentation is not identifiable; audience has no reason to listen; speaker has no credibility</td>
</tr>
</tbody>
</table>

**PEO6. Ability to develop the main results in a concise and logical manner.**

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Points are clearly related to and support the main results; points/ideas emerge with thorough logic; cues and transitions direct</td>
</tr>
<tr>
<td>4</td>
<td>Points relate to main results; points/ideas emerge fairly clearly; cues and transitions fairly direct</td>
</tr>
<tr>
<td>3</td>
<td>Points/ideas may not all be directly related to main results; many cues/transitions direct; many ideas communicated</td>
</tr>
<tr>
<td>2</td>
<td>Points/ideas only tangentially applicable; logical progression vague; cues vague; rambles somewhat</td>
</tr>
<tr>
<td>1</td>
<td>No points are identifiable; lacks any logical progress; no clear cues or transitions at all</td>
</tr>
</tbody>
</table>

**C. Ability to present sufficient research/arguments to support main results.**

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Has excellent knowledge of &amp; effectively uses relevant literature/theory</td>
</tr>
<tr>
<td>4</td>
<td>Has good knowledge of &amp; often effectively uses relevant literature/theory</td>
</tr>
</tbody>
</table>
3 Has acceptable understanding of literature/theory; may use ineffectively in areas
2 Has less than satisfactory understanding of literature/theory; does not effectively apply to thesis
1 Appears to have no understanding of or ability to use literature/theory whatsoever

D. Use of language (grammatically and field-specifically) appropriate.
5 Language/syntax correct, even elegant; topic-applicable; free from error; direct and tactful
4 Language/syntax consistently correct; largely topic-applicable; mostly free from error; mostly direct
3 Language/syntax generally correct, with few errors of usage or application; tactful
2 Language/syntax sometimes correct; many errors of usage and application
1 Language/syntax completely inappropriate or incorrectly applied; lacking tact or direction

E. Visual aids appropriate for the context and field.
5 Visual aids are appropriate, professional, interesting, and thoroughly enhance presentation.
4 Visual aids appropriate but may be unexciting; enhance presentation
3 Visual aids are appropriate but not as professional (handmade charts versus PowerPoint); enhance presentation
2 Visual aids poorly executed; have little relevance to presentation; little reference made to them
1 Visual aids nonexistent or irrelevant; little to no reference made to them

F. Ability to respond to questions in a clear fashion.
5 Speaker responds promptly, thoroughly, respectfully to questions
4 Speaker responds fairly promptly, fairly thoroughly, & respectfully to questions
3 Speaker accepts questions; is respectful; response adequate but may require further elaboration
2 Speaker accepts questions; may be impatient or uneasy with questions; responses imprecise or inadequate
1 Speaker’s response is unclear; unable to answer question; consistently misconstrues questions

G. Delivery includes effective verbal and nonverbal techniques
5 Speaker is conversational/natural; makes consistent eye contact; audible and well-paced; gestures enhance presentation; consults notes smoothly & as appropriate or does not refer to them at
all (discipline-specific issue); good posture

4 Speaker largely natural; often makes eye contact; audible; few hesitations; gestures mostly enhance; relies on notes bit more than should be necessary but without interrupting flow; posture seldom needs correction

3 Speaks with some hesitations; makes eye contact sometimes; audible; gestures are not distracting; use of notes may interrupt presentation; posture sometimes sloppy or inappropriate

2 Speaker somewhat stilted; seldom makes eye contact; sometimes inaudible; use of notes too often interrupts flow; posture inappropriate; gestures may not fit presentation

Speaker is stilted; makes no eye contact; is not audible; no gestures or are distracting; reads only from notes giving impression that speaker is unprepared

Written Communication Rubric
Definition: Competent writing conveys ideas/information in a clear, ordered, and well-supported fashion; uses a style and sources appropriate to the purpose; and employs well-structured paragraphs, correct grammar, and appropriate language.

Please complete the following form for each student in (course) ____________________

Scale: 5-Excellent  4-Above average  3-average  2-Below average  1-Poor

A Thesis: essay has a strong unifying thesis

5 Aim is clear; introduction states thesis to be defended; conclusion revisits thesis thoughtfully

4 Aim is clear; introduction states thesis; conclusion is strong

3 Aim is fairly clear; introduction attempts to explain thesis; conclusion may be vague

2 Aim is vague; introduction states no clear thesis; there may be no conclusion

1 Essay is without apparent aim or purpose; neither introduction nor conclusion are evident

B Organization: writing is effectively structured

5 Development is logical, competent, thoughtfully addresses the complexities involved

4 Development is logical and competent; essay addresses more than one of the complexities

3 Development is logical and competent; may be a few organizational problems or weaknesses

2 Development is weak, with problems of logic and flow, though topic is still addressed

1 Development is seriously flawed or illogical

C Paragraphing: paragraphs are well-developed, have clear topics, and support thesis
5 Paragraphs well-constructed; flow logically; transitions felicitous
4 Paragraphs well-constructed; flow is logical; transitions are solid
3 Paragraphs are adequate; some transitions weak
2 Paragraphs tend to be weak and vague; transitions unclear
1 Paragraphs poor; transitions vague or nonexistent; flow is illogical

D Language: Writing is grammatical and use of language is effective and appropriate for purpose
5 Grammar consistently standard; diction excellent; stylistically mature and free from error
4 Grammar largely standard; diction appropriate; largely free from usage error
3 Grammar/diction adequate, may be weak or unimaginative; some usage errors evident
2 Grammar and diction errors frequent; many errors of application
1 Grammar and diction inappropriate for task and consistently unacceptable

E Sources and Documentation: sources are appropriate and documented according to style
5 Evidence/detail judiciously-chosen and enhance thesis; documentation is impeccable
4 Evidence/detail support thesis; documentation is solid
3 Some supporting evidence/detail provided; documentation largely adequate
2 Evidence/detail trivial or inappropriate; errors of documentation evident
1 Evidence/detail inappropriate or nonexistent or no documentation provided
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>C. Alex Edwards (CHAIR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor of Computer Science</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructor</td>
<td>1983-1986</td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>1986-1990, 1993-present</td>
</tr>
<tr>
<td>Department Chair</td>
<td>2004-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCS</td>
<td>CS</td>
<td>The University of Virginia</td>
<td>1982</td>
</tr>
<tr>
<td>BS</td>
<td>Mathematics</td>
<td>Clinch Valley College (now UVa-Wise)</td>
<td>1980</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

1990-1992 – Ph.D. program at the University of Virginia – did not finish

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Regularly attend local IEEE section meetings: 4 or 5 each year
- Appalachian Regional Mathematics Seminar (ARMS – regional group of mathematics and CS faculty): 1 or 2 each year
- ABET workshop: fall 2007
- Virginia IT Workforce Summit: fall 2007
- Workshop offered by CGI on what software development tools and methodologies they use: summer 2006
6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983-1986</td>
<td>UVa-Wise</td>
<td>Assistant Director of Computer Services</td>
</tr>
<tr>
<td>1992-1993</td>
<td>Manufacturing co. in NC</td>
<td>Software development</td>
</tr>
<tr>
<td>Summer 94</td>
<td>Insurance co. in NC</td>
<td>Software development</td>
</tr>
<tr>
<td>Summer 95</td>
<td>Manufacturing co. in NC</td>
<td>Software testing</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

8. For the academic year in which the Self Study was written (fall 2008 – spring 2009) list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department Chair: 10-15 hrs per week, extra compensation and a course reduction</td>
</tr>
<tr>
<td>Chairs and Provost’s Council: .5 hrs per week</td>
</tr>
<tr>
<td>IT Symposium Planning Committee: .5 hrs per week</td>
</tr>
<tr>
<td>Science Building Renovation Committee: .1 hr per week</td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academic Program Committee: 2004-2006</td>
</tr>
<tr>
<td>Faculty Senate: 2004-2006</td>
</tr>
<tr>
<td>Math, CS, and/or software engineering faculty search committees: 2004-2008</td>
</tr>
<tr>
<td>Science Building Renovation Committee: 2006-2008</td>
</tr>
<tr>
<td>Chairs and Provost’s Council: 2004-2008</td>
</tr>
<tr>
<td>STEM Action Committee: 2006-2007</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared (fall 2008-spring 2009); the last year was the year prior to this (fall 2007-spring 2008). If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 07</td>
<td>Csc 1010</td>
<td>Object Oriented Programming (for nonmajors)</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Fall 07</td>
<td>Csc 1180</td>
<td>Found. of Computer Programming in C++</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Fall 07</td>
<td>Csc 3300</td>
<td>Fund. of Computer Science (nonmajors)</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Fall 07</td>
<td>Csc 3950</td>
<td>ST: Advanced Web Design (independent study)</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Spr 08</td>
<td>Csc 1180</td>
<td>Found. of Computer Programming in C++</td>
<td>3</td>
<td>18</td>
</tr>
<tr>
<td>Spr 08</td>
<td>Csc 4000</td>
<td>Operating Systems</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Spr 08</td>
<td>Csc 4110</td>
<td>Advanced Database Systems</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Spr 08</td>
<td>Csc4990/MIS4980</td>
<td>Senior Seminar (both classes meet together and share a syllabus)</td>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>Sum 08</td>
<td>Csc 1100</td>
<td>Computer Literacy (nonmajors)</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Sum 08</td>
<td>Csc 2180</td>
<td>Data Structures</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Fall 08</td>
<td>Csc 1010</td>
<td>Object Oriented Programming (nonmajors)</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Fall 08</td>
<td>Csc 3250</td>
<td>UNIX Systems</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Fall 08</td>
<td>Csc 4350</td>
<td>Computer Networks</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Spr 09</td>
<td>Csc 1100</td>
<td>Computer Literacy (nonmajors)</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>Spr 09</td>
<td>Csc 4000</td>
<td>Operating Systems</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Spr 09</td>
<td>Csc 4110</td>
<td>Advanced Database Systems</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spr 09</td>
<td>Csc4990/Senior Seminar</td>
<td></td>
<td>1</td>
<td>8</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: __5-10__ %.
   Please give a brief description of your major research and scholarly activities:

| My scholarship primarily consists of spending time each week reading selections from current magazines and journals of the ACM and IEEE. Current interests are computer science education and women in computing. |
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Morris Akers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>Instructor</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>Non-tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comp. Lab Coor.</td>
<td>1994-2004</td>
</tr>
<tr>
<td>Instructor of Math</td>
<td>2004-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>Math</td>
<td>UVa’s College at Wise</td>
<td>1992</td>
</tr>
<tr>
<td>MS</td>
<td>Computational Science</td>
<td>Radford University</td>
<td>1994</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

Linear, computational, integer and non-linear programming, probability models, time series analysis, statistical inference and Bayesian analysis classes, all at the University of Tennessee.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summer 2005</td>
<td>Fort Lee, Virginia</td>
<td>Data analysis</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

8. For the academic year in which the Self Study was written (fall 2008 – spring 2009) list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per
week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Senior seminar student 3 hours per week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

Academic Computing Advisory

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

Two year leave towards Ph.D. at the University of Tennessee

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared (fall 2008-spring 2009); the last year was the year prior to this (fall 2007- spring 2008). If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2007</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>18</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC3600</td>
<td>Operations Research</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Spring ’08</td>
<td>MTH1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Spring ’08</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>Spring ’08</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Spring ’08</td>
<td>CSC4500</td>
<td>Modeling and Simulation</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>MTH 1010</td>
<td>College Algebra</td>
<td>3</td>
<td>16</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Semester</td>
<td>Course Code</td>
<td>Course Title</td>
<td>Credits</td>
<td>Notes</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>--------------------------</td>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC3600</td>
<td>Operations Research</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Spring '09</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Spring '09</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Spring '09</td>
<td>MTH 1020</td>
<td>Finite Mathematics</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Spring '09</td>
<td>CSC4500</td>
<td>Modeling and Simulation</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 15%.

Please give a brief description of your major research and scholarly activities:

Ph. D. dissertation in supply chain inventory and production control
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Robert Hatch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenure-Track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>8/25/08-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D.</td>
<td>Computer Science</td>
<td>The University of Alabama</td>
<td>8/08</td>
</tr>
<tr>
<td>Masters</td>
<td>Software Engineering</td>
<td>Auburn University</td>
<td>12/02</td>
</tr>
<tr>
<td>Bachelors</td>
<td>Computer Science</td>
<td>Auburn University</td>
<td>6/00</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/03-8/08</td>
<td>The University of Alabama</td>
<td>Graduate Teaching Assistant, Graduate Research Assistant; taught Computer Science I course, assisted with exam creation and grading; taught CS-I lab; research work involved working with the CARE Research and Development Lab in helping with installation packages, as well as testing.</td>
</tr>
</tbody>
</table>
Auburn University

Graduate Teaching Assistant, Graduate Research Assistant; taught Introduction to Engineering lab classes; graded papers for Intro course, as well as upper-division courses. Taught “Auburn Experience” course in Fall 2002. Research duties included webmaster for the Computer Science department website in Summer 2001.

7. Consulting—list agencies and dates, and briefly describe each project: none

8. For the academic year in which the Self Study was written (fall 2008 – spring 2009) list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part-time faculty, please indicate here which courses.

None – first-year faculty.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

Software development for Master’s and Ph.D. work. Master’s work involved creating review checklists for Palm Pilot technology. Ph.D. work consisted of creating reading software prototypes with both children and adults, and testing to see what software children preferred and what software the adults preferred.

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared (fall 2008-spring 2009); the last year was the year prior to this(fall 2007- spring 2008). If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2008</td>
<td>CSC 1100</td>
<td>Computer Literacy, Section 1</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC 1100</td>
<td>Computer Literacy, Section 2</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC 4300</td>
<td>Computer Architecture</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>SWE 3210</td>
<td>Software Quality Assurance</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 1100</td>
<td>Computer Literacy</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td>---------------------------</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 3750</td>
<td>Web Technology</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>SWE 2130</td>
<td>Software Construction Technology</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>SWE 3220</td>
<td>Software Requirements and Modeling</td>
<td>3</td>
<td>5</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 20%.

Please give a brief description of your major research and scholarly activities:

Major research areas include: software development for children; Human-Computer Interaction; Software Engineering, with an emphasis on process.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Abrar A. Qureshi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenure Track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>Jan. 10, 2007 to Present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.S</td>
<td>Math</td>
<td>University of the Punjab</td>
<td>1987/88</td>
</tr>
<tr>
<td>B.S</td>
<td>Electrical Engineering</td>
<td>Central Philippine University</td>
<td>03/1994</td>
</tr>
<tr>
<td>M.S</td>
<td>Computer Engineering</td>
<td>Florida Institute of Technology</td>
<td>07/1997</td>
</tr>
<tr>
<td>Ph. D.</td>
<td>Computer Engineering</td>
<td>Florida Institute of Technology</td>
<td>05/2006</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Event</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE Tri-Cities Section meeting (Presenter)</td>
<td>April 2007</td>
</tr>
<tr>
<td>Virginia IT Workforce Summit - Fall 2007</td>
<td></td>
</tr>
<tr>
<td>ABET faculty workshop on assessing program - Fall 2007</td>
<td></td>
</tr>
<tr>
<td>All-University Retreat, Science and Technology at U.Va. - Fall 2008</td>
<td></td>
</tr>
<tr>
<td>3rd International Conference on Sociology in Athens, Greece – Presented a paper on Social Software and Its Impact on our Society – May 2009</td>
<td></td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>08/1996-1997</td>
<td>Florida Institute of Technology, Melbourne, FL</td>
<td>Instructed a computer design lab, involved in building a simple computer (SBC) using an 8-bit micro controller. Instructed a digital logic design class and lab, involved in building digital circuits and testing them with a logic simulator.</td>
</tr>
<tr>
<td>02/1998-2007</td>
<td>GE Transportation, Melbourne, FL</td>
<td>Responsible for Software development, test planning, test case development, requirement analysis, reviews and test case execution.</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/2008</td>
<td>CGI, Lebanon</td>
<td>Training employees in Software Testing and Quality</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written (fall 2008 – spring 2009) list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

- Member Academic Computing Advisory Committee – ¼ hour per Week
- Member Faculty Development Committee ½ hour per week
- Member Committee on Distance Education 1-2 hours per week
- Currently advising four majors in Software Engineering – 1 hour per week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

- Academic Program Committee: 2007-2008
- Software Engineering and CS faculty search committees: 2007-2008

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

<table>
<thead>
<tr>
<th>Developed three new courses for CS and Honors Classes</th>
</tr>
</thead>
</table>

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared (fall 2008-spring 2009); the last year was the year prior to this (fall 2007- spring 2008). If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spr 2007</td>
<td>CSC 1100-0001</td>
<td>Computer Literacy</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Spr 2007</td>
<td>CSC 1100-0002</td>
<td>Computer Literacy</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>Spr 2007</td>
<td>SWE 2130</td>
<td>Software Construction Technology</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Spr 2007</td>
<td>SWE 2210</td>
<td>Testing, Verification and Validation</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC 3750</td>
<td>Web Technologies</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC 4300</td>
<td>Computer Architecture</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>SWE 2130</td>
<td>Software Construction Technology</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>SWE 2210</td>
<td>Testing, Verification and Validation</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>SWE 2130</td>
<td>Software Construction Technology</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>SWE 2210</td>
<td>Testing, Verification and Validation</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>SWE 3210</td>
<td>Software Quality Assurance</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>SWE 3230</td>
<td>Software Configuration Management</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC1100- 3301</td>
<td>Computer Literacy</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>HONR 3950</td>
<td>Software and Society</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>SWE 4240</td>
<td>Software Project Management</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>SWE 4980</td>
<td>Capstone Project</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Sum 2008</td>
<td>CSC 4380</td>
<td>Information Security</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>CSC1100- 3301</td>
<td>Computer Literacy</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Semester</td>
<td>Course Code</td>
<td>Course Title</td>
<td>Credits</td>
<td>Hours</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>------------------------------------</td>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>SWE 3230</td>
<td>Software Configuration Management</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>SWE 2210</td>
<td>Testing, Verification and Validation</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>SWE 4990</td>
<td>Capstone Project</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: __30___%.

Please give a brief description of your major research and scholarly activities:

- I have been pursuing research into Software Testing and Quality area. My research focus on the current challenges faced in the Industry today. These challenges not only include the product quality but also the contract, process, procurement, use and maintenance challenges.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Dr. Daniel A. Ray</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenure Track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>August 2007 - Present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>CS</td>
<td>The University of Alabama</td>
<td>5/2003</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>The University of Alabama</td>
<td>5/2005</td>
</tr>
<tr>
<td>PhD</td>
<td>CS</td>
<td>The University of Alabama</td>
<td>8/2007</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Event</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Third WG 11.9 International Conference on Digital Forensics (Presenter), January 2007</td>
<td></td>
</tr>
<tr>
<td>ABET Best Assessment Procedures Symposium – April 2008</td>
<td></td>
</tr>
<tr>
<td>Cyber Security and Information Infrastructure Research Workshop (Presenter) – May 2008</td>
<td></td>
</tr>
<tr>
<td>The 2008 World Congress in Computer Science, Computer Engineering, and Applied Computing (Presenter) – July 2008</td>
<td></td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
2005-2007 | N/A | Freelance Web Design Work

7. Consulting—list agencies and dates, and briefly describe each project:

8. For the academic year in which the Self Study was written (fall 2008 – spring 2009) list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Member Academic Computing Advisory Committee – Negligible Time per Week

Faculty Advisor for the Student Chapter of the ACM – 1-2 hours per week

Currently advising three majors in MIS and CS – Approx. 1 hour per week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

For the past three years I have been working intermittently with fellow colleagues on development of a software system for enterprise networks that supports proactive digital forensics.

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared (fall 2008-spring 2009); the last year was the year prior to this (fall 2007-spring 2008). If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2007</td>
<td>CSC1180a</td>
<td>Fundamentals of Programming</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC1180b</td>
<td>Fundamentals of Programming</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC3400</td>
<td>Database Design</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC3180</td>
<td>Algorithms</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>CSC1100</td>
<td>Computer Literacy</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>CSC1180</td>
<td>Fundamentals of Programming</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>CSC3180</td>
<td>Algorithms</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Spr 2008</td>
<td>CSC4150</td>
<td>Intro. to Robotics and A.I.</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC1180a</td>
<td>Fundamentals of Programming</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC1180b</td>
<td>Fundamentals of Programming</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC3400</td>
<td>Database Design</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC3180</td>
<td>Algorithms</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>CSC1180b</td>
<td>Fundamentals of Programming</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>CSC2230</td>
<td>Programming in C#</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Spr 2009</td>
<td>CSC3180</td>
<td>Algorithms</td>
<td>3</td>
<td>15</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: 40%.

Please give a brief description of your major research and scholarly activities:

I have been pursuing research into digital and proactive forensics in both theory and application. This means trying to understand how trust issues can be handled with digital data and what technologies and techniques (DSLs, Proactive Forensics Programs, etc) can be used to further the fields of digital forensics and security.

In addition, I have recently been working on methods for integrating robotics hardware into the CS curriculum and plan on instituting a summer program in robotics for HS juniors and seniors this coming summer.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Jacob Somervell</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenure-Track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>8/04-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D.</td>
<td>Computer Science</td>
<td>Virginia Polytechnic Institute and State University (Va Tech)</td>
<td>7/04</td>
</tr>
<tr>
<td>Masters</td>
<td>Computer Science</td>
<td>Virginia Polytechnic Institute and State University (Va Tech)</td>
<td>5/01</td>
</tr>
<tr>
<td>Bachelors</td>
<td>Computer Information Systems</td>
<td>The University of Virginia's College at Wise (formerly Clinch Valley College of the University of Virginia)</td>
<td>5/99</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Event</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>2004 Conference on Frontiers in Education (FIE’04)</td>
<td>presented a paper on case based teaching of HCI concepts</td>
</tr>
<tr>
<td>Human Factors and Ergonomics Society 48th annual meeting (HFES’04)</td>
<td>presented a paper on heuristic creation methods</td>
</tr>
<tr>
<td>2006 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS’06)</td>
<td>presented a paper on pair programming in CS1 courses</td>
</tr>
<tr>
<td>2007 High Performance Computing Bootcamp (HPCB) at The University of Virginia</td>
<td>attended</td>
</tr>
<tr>
<td>Event</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>2008 Best Assessment Practices Symposium (BAPS '08) – attended</td>
<td></td>
</tr>
<tr>
<td>Living in a KnowlEdge Society (LIKES) conference, Va Tech, Oct '08</td>
<td></td>
</tr>
<tr>
<td>ARMS (Appalachian Regional Math Society) meetings: Jan ’06, Jan’07, Apr ’07, Sep ’07, Apr ’08, Sep ’08, Jan ’09</td>
<td></td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

7. Consulting—list agencies and dates, and briefly describe each project: none

8. For the academic year in which the Self Study was written (fall 2008 – spring 2009) list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

   **Advising**: 28 students, most from our school’s Freshman Advising program

   Average maybe 2 hours per week over entire school year

   **Committees**: 6 permanent

   - Faculty Senate, Executive Officer (1 hour per month for meeting plus additional duties as executive officer)
   - General Education Committee (1 hour per month when needed)
   - Undergraduate Research Council (1 hour per week, weekly meetings)
   - Teacher Education Program Admission Committee (1 hour per month)
   - Academic Computing Advisory Committee (1 hour per month when needed)
   - Faculty Relations Committee (1 hour per week when needed)

   **ABET self-study preparation (5 hours per week) since spring 2008**

   **Assessment** [data collection, analysis, and reporting] for Computer Science, MIS, and Software Engineering programs (maybe 1 hour per week over a semester)

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

   **2004-05**: none, first year at institution
2005-06:

Honors Committee (1 year)
GIS Committee (1 year)
Librarian Search Committee (ad-hoc, temporary)

2006-07:

Honors Committee
Director of Information Technology/CIO search
Welcome Week Committee
Library Search
Admissions Counselor Search
Teacher Education Program Admission
Vice Chancellor for Information Technology Search
General Education Committee
Academic Counselor Search
Software Engineer Search

2007-08:

Teacher Education Program Admission
General Education Committee
Resident Director Selection Committee
Assistant Director of Residence Life Search
Assistant Director of Recreation and Intramurals Search
Departmental Representative to Faculty Senate, Executive Officer
Undergraduate Research Council
Computer Science Search
10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

I worked on a test program studying Queen’s Domination for a former colleague, Paul Burchette in 2006. [software development]

I worked with a former colleague, David Lane, on putting our mathematics placement exam online. (2005-2007) [software development]

I participated in our departmental faculty research meetings from 2005-2008. These are informal meetings designed to keep our faculty involved in research as much as possible. Local, individual projects are presented and discussed.
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared (fall 2008-spring 2009); the last year was the year prior to this (fall 2007- spring 2008). If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2007</td>
<td>SEM 1010</td>
<td>Freshman Seminar</td>
<td>1</td>
<td>17</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC 2220</td>
<td>Programming in Java</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC 2180</td>
<td>Data Structures</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC 3250</td>
<td>Unix/Linux Systems</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CSC 4350</td>
<td>Computer Networks</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>SEM 1020</td>
<td>Freshman Seminar</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CSC2180a</td>
<td>Data Structures</td>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CSC2180b</td>
<td>Data Structures</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CSC2180c</td>
<td>Data Structures</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CSC3260</td>
<td>Introduction to Human Computer Interaction</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CSC 4200</td>
<td>Concepts of Programming Languages</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC 2220</td>
<td>Programming in Java</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC 2300</td>
<td>Software Engineering</td>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC 2180</td>
<td>Data Structures</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Fall 2008</td>
<td>CSC 3710</td>
<td>Discrete Mathematical Structures</td>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 2300</td>
<td>Software Engineering</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 2300</td>
<td>Software Engineering</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 2180</td>
<td>Data Structures</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 3260</td>
<td>Introduction to Human Computer Interaction</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Spring 2009</td>
<td>CSC 4200</td>
<td>Concepts of Programming Languages</td>
<td>3</td>
<td>13</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: 25%.

Please give a brief description of your major research and scholarly activities:

During the first two years of my time here, I was able to finish some projects that carried over from my dissertation work. My focus then shifted to computer science education and I was able to do a small study on pedagogical techniques in CS1 courses.
APPENDIX C – LABORATORY EQUIPMENT

The Department of Mathematics and Computer Science maintains two computer labs. One lab is open to public use, and the other lab is open only to MIS/CS/SE majors.

The public lab contains 36 patron computers and 1 teaching workstation. It is equipped with 2 monochrome laser printers, 1 color laser printer, a sound system w/amplifier, a projector screen, a ceiling-mount projector, a DVD/VCR player, and a flatbed scanner. These computers are a mix of Dell Optiplex GX280 models and Optiplex 620 models running Windows XP. Software installed includes Office 2007 Professional, Mathematica, MATLAB, Minitab, and Visual Studio 2008. The computers in this lab are joined to the college’s Active Directory domain.

The private lab consists of 14 patron computers and 1 teaching workstation. This lab does not contain a projector. Instead, the teaching workstation is connected to a 42” plasma TV. The lab is equipped with 1 monochrome laser printer, 1 color laser printer, and a flatbed scanner. The computers in this lab are Dell Optiplex 740 models. Each computer is set up to dual boot. Users can boot to either Windows XP or Linux (Fedora 9). This lab is not joined to the campus Active Directory domain. Instead, users are authenticated via OpenLDAP and Samba running on the department’s own servers. Students must request accounts in person with the departmental systems administrator, who then approves or denies accounts based on student eligibility. Students are granted power user privileges in this lab, whereas all Active Directory joined campus computers grant students only user privileges. Software in this lab includes Office 2007, Mathematica, MATLAB, Minitab, Visual Studio 2008, Adobe Creative Suite 3, LabVIEW, and various open source programs and utilities, such as Netbeans, Eclipse, and Geany.
COURSE DESCRIPTION

Course Number
CSC 1180

Course Title:
Foundations of Computer Programming in C/C++

Hours:
4

Required/Elective?:
Required

Current Catalog Description:
Beginning computer programming. C/C++ language syntax, data types, sequence, branch, loop, and select constructs, pointer variables, record structures, input/output, and programming techniques. Student cannot receive credit for both CSC 181 and CSC 320.

Textbook:
C++ How to Program (6th Edition), Deitel & Deitel

Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
MWF, 1hr per class

References:

Course Outcomes:
Students will be able to take a general set of programming requirements and generate programs using correct C++ syntax. Students will also be able to derive the functionality of simple programs given only the C++ source code. Students will be able to demonstrate an understanding of all basic C++ control structures, good programming practice, use of an integrated development environment, skill in both object-oriented and procedural programming, input and output techniques as well as a basic understanding of data types and objects.

Relationship between Course Outcomes and Program Outcomes:
Understanding programs and programming solutions, as well as the proper use of programming techniques directly relates to PO1, an ability to identify and analyze constraints and trade-offs for a specific problem. These, in addition with the use of an IDE, skill in both object-oriented and procedural programming and the use of C++ control structures, relate directly to PO2, an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills and tools.
Prerequisites by Topic: None

Major Topics Covered in the Course: C++ programming language syntax, basic programming practices, data types, input/output techniques, basic program constructs, functions and procedures, structured and object-based techniques

Assessment Plan for the Course: Assessment material for the course includes the examination of weekly programming assignments of incrementally increasing difficulty, three to four comprehensive examinations throughout the semester, and a comprehensive final examination.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0.5</td>
<td></td>
<td>Software design</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Data Structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming languages</td>
<td>1.5</td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Daniel A. Ray
Date: May 26, 2009
COURSE DESCRIPTION

Course Number
CSC 2180

Course Title
Data Structures

Hours
3

Required/Elective?
Required

Current Catalog Description
Data and abstract data types; programming principles, lists, dynamic memory allocation, stacks, queues, trees, graphs, and recursion.

Textbook
Data Structures Using C++. D.S. Malik

Class/laboratory schedule, ie, number of sessions per week, duration of sessions
MWF (50min) or TH (75 min)

References
C++ How to Program. Deitel & Deitel
C++ Plus Data Structures. Nell Dale

Course Outcomes
Students will be able to:
- Know and understand basic abstract data types
- Write, compile, and execute programs that illustrate usage of these data types
- Understand the situations and circumstances for using these data structures
- Determine the appropriate data structure(s) for a given problem

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:
PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.
PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.
PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

Prerequisites by Topic
CSC 1180 – Fundamentals of Computer Programming

**Major Topics Covered in the Course**
Linked lists, pointers, dynamic arrays, recursion versus iteration, stacks, queues, trees, graphs

**Assessment Plan for the Course**
Pop up Quizzes, programming projects, exams

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

*For a computer science program*

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.25</td>
<td></td>
<td>Software design</td>
<td>.25</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>2.75</td>
<td></td>
<td>Concepts of programming languages</td>
<td>.25</td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Jacob Somervell
Date: 6/8/09
COURSE DESCRIPTION

Course Number
CSC 2300

Course Title
Software Engineering

Hours
3

Required/Elective?
Required

Current Catalog Description
An introduction to software engineering. Development of a total system concept is emphasized. Project planning, requirements analysis, system design, program design, program implementation, program testing, system testing, system delivery, maintenance, documentation, and examining the development process.

Textbook

Class/laboratory schedule, ie, number of sessions per week, duration of sessions
MWF (50min) or TH (75 min)

References

Course Outcomes

- Students should know and understand basic software development concepts and practices
- Students should be able to gather requirements, design a system, prototype the system, test the system, and deploy the system
- Students should understand basic concepts of project management and scheduling
- Students should apply these skills in the development of an actual software project

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:

PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.

PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

PO5: It will provide the students with opportunities for team based work and experience.
Prerequisites by Topic
CSC 2180 – Data Structures

Major Topics Covered in the Course
Software development process, project management, project scheduling, requirements gathering, system design and modeling, development, testing

Assessment Plan for the Course
Pop up Quizzes, project and presentations, Exams

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Jacob Somervell
Date: 6/8/09
**COURSE DESCRIPTION**

**Course Number**
CSC 3260

**Course Title**
Introduction to Human Computer Interaction

**Hours**
3

**Required/Elective?**
Elective

**Current Catalog Description**
User-centered graphical user interface (GUI) development concepts and application, iterative development process, activity, information, and interaction design, prototyping tools, user testing, event-driven programming, analytic and empiric evaluation

**Textbook**
About Face 3.0. Alan Cooper

**Class/laboratory schedule, ie, number of sessions per week, duration of sessions**
MWF (50min) or TH (75 min)

**References**
Usability Engineering. Rosson & Carroll
Design of Everyday Things. Don Norman

**Course Outcomes**
Students will be able to:
- Be familiar with fundamental concepts of human-computer interaction
- Design, prototype, and test interfaces for various applications
- Design and apply appropriate evaluation techniques to various designs

**Relationship between Course Outcomes and Program Outcomes**
The above course outcomes are related to the program outcomes in the following manners:

**PO1:** It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.

**PO2:** It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

**PO4:** It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

**PO5:** It will also provide students with opportunities to work on team projects and gain valuable team experience.
Prerequisites by Topic
CSC 1180 – Fundamentals of Computer Programming

Major Topics Covered in the Course
Interface design, interaction design, user models, design models, analytic evaluation methods, and empiric evaluation methods, prototyping, ethics

Assessment Plan for the Course
In-class activities, quizzes, exams, project, project presentations

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

*For a computer science program*
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Jacob Somervell
Date: 6/8/09
**COURSE DESCRIPTION**

**Course Number:**
CSC 3400

**Course Title:**
Database Design

**Hours:**
3

**Required/Elective?:**
Required

**Current Catalog Description:**
Evolution of data models, relational, network, hierarchical, and ER model concepts. DDL and DML concepts, relational algebra, relational calculus, SQL, Network and Hierarchical DMLs. Design, implementation, and manipulation of a relational database using current database technology.

**Textbook:** Fundamentals of Database Systems, Elmasri & Navathe

**Class/laboratory schedule, ie, number of sessions per week, duration of sessions:** TR, 1.5 hrs per class

**References:**

**Course Outcomes:**
- Students will exhibit an understanding of how to analyze relational data and system requirements to create proper database structures. They will show a basic to intermediate understanding of the design and maintenance of database management systems.
- Also, students will be able to use modern database management systems to implement and make use of their own database designs.
- Additionally, students will have mastered the basic methods for capturing requirements and functionality in models (Entity-relationship, Enhance Entity-Relationship, Relational, etc) and incrementally mapping from these captured models to a suitable database design.
- They will also gain a core understanding of relational algebra, querying databases via SQL, and functional dependencies and normalization for relational databases.

**Relationship between Course Outcomes and Program Outcomes:**
There is a strong relationship between the above stated outcomes with both PO1 and PO2 (An ability to identify and analyze constraints and trade-offs for a specific problem, and an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills and tools, respectively) in that requirements analysis and the application of appropriate design techniques is key in this course. Additionally, teamwork assignments and in-class presentations in the course directly relate to PO4 and PO5 (an ability to effectively communicate their solutions and impacts to the appropriate audience, and an ability to do PO1 – PO4 as an individual or while working with groups or teams, respectively).
Prerequisites by Topic: Data and abstract data types; programming principles, lists, abstract data types.

Major Topics Covered in the Course: Major topics covered include the fundamental principles of database systems, relational data models, and the design, organization, querying, and management of database systems.

Assessment Plan for the Course: Assessment material includes three or four homework and database design assignments, including several team-based assignments, two comprehensive examinations, and one comprehensive final examination.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.5</td>
<td></td>
<td>Software Design</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>Data Structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Daniel A. Ray
Date: May 26, 2009
COURSE DESCRIPTION

Course Number:
CSC 3180

Course Title:
Introduction to Algorithms

Hours:
3

Required/Elective?:
Required

Current Catalog Description:
Sorting and searching; recursion; analysis of complexity; algorithm paradigms; NP complete problems; complexity metrics


Class/laboratory schedule, i.e., number of sessions per week, duration of sessions: TR, 75 minutes per class

References:

Course Outcomes:
Students will exhibit a basic understanding both of the ability to use analytical tools to find the mathematically founded relative complexity of algorithmic solutions as well as the ability to use basic algorithmic design techniques (divide-and-conquer, dynamic programming, the greedy technique, etc.) in conjunction with appropriate data structures to create efficient algorithms.

Relationship between Course Outcomes and Program Outcomes:
There is a strong relationship between the above stated outcomes with both PO1 and PO2 (An ability to identify and analyze constraints and trade-offs for a specific problem, and an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills and tools, respectively).

Prerequisites by Topic:
Data and abstract data types; programming principles, lists, dynamic memory allocation, stacks, queues, trees, graphs, and recursion.

Major Topics Covered in the Course:
Analysis of complexity, algorithm design paradigms, foundational algorithms, sorting and searching, recursion, complexity theory
Assessment Plan for the Course:
Assessment material includes five or six homework assignments consisting of hand-selected problems from the course text appropriate for the most recently covered material, three comprehensive examinations, and one comprehensive final examination.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a computer science program

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>2.5</td>
<td></td>
<td>Software Design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Structures</td>
<td>.5</td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Daniel A. Ray
Date: May 26, 2009
COURSE DESCRIPTION

Course Number
CSC 3710

Course Title
Discrete Structures

Hours
3

Required/Elective?
Required

Current Catalog Description
Topics include: algebra of sets, equivalence relations, counting techniques, induction, algebraic structures, flowcharts, algorithms, syntax and semantics, graphs, monoids, and machines and logic

Textbook
Discrete Mathematical Structures. Kolman, Busby, and Ross

Class/laboratory schedule, ie, number of sessions per week, duration of sessions
MWF (50min) or TH (75 min)

References

Course Outcomes
Students will be able to:
- Know and understand basic mathematical structures: sets, relations, flowcharts and graphs
- Understand the concepts necessary for advanced computer science mastery, including logic, machines, and counting techniques
- Have introductory knowledge of mathematical proofs and techniques, including direct proof, proof by contradiction, and mathematical induction

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:

PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.

PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

Prerequisites by Topic
MTH 2040 – Calculus I
**Major Topics Covered in the Course**
Set theory, symbolic logic, methods of proof, counting techniques, graphs, trees, relations, groups and semi-groups

**Assessment Plan for the Course**
Homework and tests

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

*For a computer science program*

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Jacob Somervell
Date: 6/8/09
COURSE DESCRIPTION

Course Number: CSC 4000

Course Title: Operating Systems: Theory and Practice

Hours: 3

Required/Elective? Required

Current Catalog Description: The concepts behind the design and working of the operating system on a computer are introduced. A system programming course emphasizing topics such as resource management, processes, process management, file design and allocation, system calls, and shell design.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions: course may be taught on MWF schedule, meeting three times a week for one hour, or on a T-Th schedule, meeting two times a week for 1.5 hours.

References:

Course Outcomes: Students will demonstrate an understanding of processes and threads. Students will demonstrate an understanding of interprocess communication. Students will be able to describe virtual memory concepts. Students will demonstrate an understanding of File and I/O Systems. Students will demonstrate an understanding of process scheduling. Students will demonstrate an understanding of security and protection mechanisms.

Relationship between Course Outcomes and Program Outcomes: There is a strong connection with P01 (An ability to identify and analyze constraints and trade-offs for a specific problem) and there is a link to P02 (An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools) when discussing the various operating system structures. Discussion on security and threats provides a limited connection with P03 (An ability to assess solutions in light of societal and ethical issues and understand the impacts of those solutions on the world). And there is a link to P04 (An ability to effectively communicate their solutions and impacts to the appropriate audience) and P05 (An ability to do 1-4 as an individual or while working with groups or teams) through work students do on assignments and projects.

Prerequisites by Topic: programming concepts, abstract data types
Major Topics Covered in the Course: hardware review; types of operating systems; operating system structures; processes and threads; interprocess communication; memory management; virtual memory; I/O systems; file systems; synchronization and deadlock; security and protection;

Assessment Plan for the Course: Assessment material for the course includes several homework assignments, programming projects, two (or three) tests, and a final exam.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>1</td>
<td>Software design</td>
<td></td>
<td>.5</td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td>.5</td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Alex Edwards

Date: 6/15/09
COURSE DESCRIPTION

Course Number
CSC 4200

Course Title
Programming Languages

Hours
3

Required/Elective?
Required

Current Catalog Description
Survey of contemporary languages, compiler construction techniques, parsing, formal grammars, virtual machines, subprograms, functional and logic programming.

Textbook
Concepts of Programming Languages. Robert Sebesta

Class/laboratory schedule, ie, number of sessions per week, duration of sessions
MWF (50min) or TH (75 min)

References

Course Outcomes
Students will be able to:

- Know and understand fundamental differences among language types, including procedural, functional, logic, and object oriented languages
- Appreciate the reasons for studying various programming languages and concepts
- Recognize and be able to identify multiple programming languages
- Determine among various options which language would be best for specific problem

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:

PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.

PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

Prerequisites by Topic
CSC 2180 – Data Structures
Major Topics Covered in the Course
Reason for studying programming languages, subprograms, grammars, lexical analysis, data types and implementation, programming structures and implementations, functional programming (Scheme), logic programming (Prolog)

Assessment Plan for the Course
Homework, programming projects, and tests

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>.25</td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td>.25</td>
<td>Concepts of programming languages</td>
<td>2.5</td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Jacob Somervell
Date: 6/8/09
COURSE DESCRIPTION

Course Number:
CSC 4300

Course Title:
Computer Architecture

Hours:
3

Required/Elective?
Required

Current Catalog Description: A detailed study of the design and functional organization of a modern digital computer. Instruction sets, I/O handling, interrupts, addressing schemes, microprogramming and memory management are investigated.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions: Two sessions per week, 75 minutes each.

References: On-line references suggested by textbook author.

Course Outcomes:
- Students will be able to understand different types of memory components, such as cache, internal memory, and external memory.
- Students will be able to know the different kinds of storage and the capacities. With respect to caches, the students will be able to understand the different strategies involved in the storage and retrieval of data in a cache.
- Also, students will be able to understand the different addressing strategies.
- They should also understand computer arithmetic, instruction generation and pipelining.
- Last, the students should understand the progression of technologies used for the processor, and realize why current configurations are used.

Relationship between Course Outcomes and Program Outcomes: The course outcomes best relate to the following program outcomes: 1) an ability to identify and analyze constraints and trade-offs for a specific problem, and 2) an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

An example of meeting the first program outcome occurs when studying the various caching strategies, and realizing that a simple solution may not be the most time-effective solution.

An example of the second program outcome being met involves completing homework assignments, which call for the application of calculations and formulas discussed in class.

Prerequisites by Topic
Algorithms – realizing what solutions yield the fastest results
Stacks – this data structure is used mapping/reading instructions from memory

Logic

**Major Topics Covered in the Course:**
Number Systems
Digital Logic
Why Computer Organization and Architecture are Studied
Computer Evolution and Performance
Top-Level View of Computer Components, Functionality, and the Interconnections in a Computer
Cache Memory
Internal Memory
External Memory
Computer Arithmetic
Instruction Sets
Processor Structure
Reduced Instruction Set Computer (RISC) Architecture
Control Unit Operation

**Assessment Plan for the Course:** Assignments and Exams

**How Data in the Course is Used to Assess Program Outcomes:** N/A

*For a computer science program*

**Estimate Curriculum Category Content (Semester hours)**

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>.5</td>
<td>Software design</td>
<td></td>
<td>.25</td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td>1</td>
<td>Concepts of programming languages</td>
<td></td>
<td>.25</td>
</tr>
</tbody>
</table>

Prepared by: Robert Hatch
Date: May 26, 2009
COURSE DESCRIPTION

Course Number: SWE 1790

Course Title: Engineering Leadership

Hours: 3

Required/Elective? Required

Current Catalog Description:
Professionalism, group dynamics, professional ethics, psychology, social responsibility of engineers, legal issues for engineers, negotiation skills, documentation practice.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
Evening course, 1 day per week

References: On-line references suggested by textbook author.

Course Outcomes:
Students will be introduced to the concepts of engineering leadership
Students will understand basis of a career in engineering through exposure to professionalism, teamwork, communication, ethics, and problem solving

Relationship between Course Outcomes and Program Outcomes:

PO1. An ability to identify and analyze constraints and trade offs for a specific problem.

A heavy focus on problem solving will provide students with opportunities to meet outcome 1

PO2. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

A heavy focus on problem solving will provide students with opportunities to meet outcome 2

PO3. An ability to assess solutions in light of societal and ethical issues and understand the impacts of those solutions on the world.

Heavy focus on societal and ethical issues surrounding engineering practices and solutions will provide students with opportunities to meet outcome 3

PO4. An ability to effectively communicate their solutions and impacts to the appropriate audience.

Requiring students to write reports and give oral reports in class will provide students with opportunities to meet outcome 4

PO5. An ability to work effectively in a team setting.
Team projects provide students with opportunities to meet outcome 5

Prerequisites by Topic
None

Major Topics Covered in the Course:
Engineering as a career
Professionalism
Team work
Ethics
Communication skills
Problem solving
Social responsibility
Project planning
Engineering’s importance to software development

Assessment Plan for the Course:
Assignments and Exams

How Data in the Course is Used to Assess Program Outcomes:
N/A

Prepared by: Jacob Somervell
Date: June 23, 2009
COURSE DESCRIPTION

Course Number: SWE 2130

Course Title: Software Construction Technology

Hours: 3

Required/Elective? Required

Current Catalog Description: Graphical languages, software metrics, automatic code generation, development tool integration, component library engineering, architecture styles and patterns, off-the-shelf software integration, development tools, evolutionary development strategies, formal design languages, state based and table based design, design patterns, object oriented design evaluation, efficiency, reliability, maintainability, design modification, reverse engineering.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions: Three sessions per week, 50 minutes each.

References: On-line references suggested by textbook author.

Course Outcomes: Students will have knowledge of the different types of tools available to assist them when writing programs and developing projects. Students will also have knowledge on good programming practices, such as naming conventions and how to introduce changes to software, for example. Finally, students will be able to identify and critique their own programming styles after completing a series programming assignments, in the hopes of achieving the better programming practices studied in the course.

Relationship between Course Outcomes and Program Outcomes: The course outcomes best relate to the following program outcomes: 1) an ability to identify and analyze constraints and trade-offs for a specific problem; 2) an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools; 3) an ability to effectively communicate their solutions and impacts to the appropriate audience; and 4) be able to complete 1)-3) in a group setting. An example of meeting the first program outcome occurs when studying the insertion of a repetitive task, and realizing that a loop may not yield the quickest result, in terms of execution time. An example of the second program outcome being met involves completing programming projects; the students are left to determine how to complete the assignment, and may think that their initial ideas may not be best after beginning the project.

Prerequisite by Topic
Algorithms – realizing what solutions yield the fastest results
Data Structures – helpful to have knowledge of basic data structures, to help complete the programming assignments

Major Topics Covered in the Course:
Metaphors
Prerequisites
Programming Language and Technology Choices
Design Issues
ADT Guidelines
Containment, Inheritance
Variable Naming Conventions
Testing
Debugging
Code Tuning
Integration
Refactoring
Using Software Tools
Self-Documenting Code
Personal Character
Loops – programming practices
Fundamental Data Types – programming practices

Assessment Plan for the Course: Assignments and Exams

How Data in the Course is Used to Assess Program Outcomes: N/A

Estimate Curriculum Category Content (Semester hours)

Prepared by: Robert Hatch
Date: June 23, 2009
COURSE DESCRIPTION

Course Number:
SWE 2210

Course Title:
Testing, Verification and Validation

Hours:
3

Required/Elective?
Required

Current Catalog Description
Requirements oriented testing, test plan design, effective testing techniques, test coverage evaluation, statistical techniques for testing, reviews and inspections.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
MWF (50mins)

References

The Art of Software Testing, Glenford Myers, Wiley.


Course Outcomes: Upon successful completion of this course students will be able to:

- Understand the basic concepts of software verification and validation
- Investigate and understand the prominent software testing techniques and tools.
- Develop a software verification and validation plan
- Perform effective & efficient structural testing of software under test
- Perform effective & efficient functional testing of software under test
- Integrate & test the various units & components of software under test
- Plan, track & control the software testing effort

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:

PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.

PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.
PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.
PO5: It will provide the students with opportunities for team based work and experience.

Prerequisites by Topic: None

Major Topics Covered in the Course
Software Test Overview covers principles and concepts
Development of Testing Strategies, Test Plans, Test case design, and Schedules Test Planning
Test Results and Reporting
Software Verification Requirement and Validation needed
Unit, Integration, System Testing and Acceptance Testing
Equivalence Partitioning and Boundary Analysis
Understanding What Needs to be Tested
(Requirement oriented testing)
Black box testing techniques
White box testing techniques
Gray box testing techniques
Integration testing
Regression testing

Assessment Plan for the Course
Pop up Quizzes, programming projects, exams

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a Software Engineering program

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0.5</td>
<td>0.5</td>
<td>Software Design</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Data Structures</td>
<td>0.5</td>
<td>0.5</td>
<td>Concepts of programming languages</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Prepared by: Dr. Abrar A. Qureshi
Date: June 23, 2009
COURSE DESCRIPTION

Course Number:  SWE 2790

Course Title:  Engineering Economics

Hours:  3

Required/Elective?  Required

Current Catalog Description:
Supply and Demand analysis, break-even analysis, cost benefit analyses, investments, marketing, time value of money, risk potentials.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
Evening course, 2 days per week

References:  On-line references suggested by textbook author.

Course Outcomes:
Students will be introduced to the concepts of engineering economics.
Students will apply knowledge to practical problems.
Students will be prepared to take the Fundamentals of Engineering (FE) exam.

Relationship between Course Outcomes and Program Outcomes:

PO1. An ability to identify and analyze constraints and trade offs for a specific problem.
   A heavy focus on problem solving (with focus on monetary issues) will provide students with opportunities to meet outcome 1

PO2. An ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.
   A heavy focus on practical problem solving will provide students with opportunities to meet outcome 2

PO3. An ability to assess solutions in light of societal and ethical issues and understand the impacts of those solutions on the world.
   Some focus on societal and ethical issues surrounding engineering practices and solutions (with focus on monetary issues) will provide students with opportunities to meet outcome 3

PO4. An ability to effectively communicate their solutions and impacts to the appropriate audience.
   Requiring students to write examination answers and give oral opinions in class will provide students with opportunities to meet outcome 4
Prerequisites by Topic
MTH 2040 - Calculus I

Major Topics Covered in the Course:
Understanding money and its management (time value of money)
Evaluating business and engineering assets
   Present worth analysis
   Rate-of-return analysis
Development of project cash flows
   Depreciation
   Taxes
   Discount Rates
   Project Risk
Replacement basics
Benefit-cost analysis
Financial statements
Financial ratios analysis

Assessment Plan for the Course:
   Assignments, tests, in-class participation, and exams

How Data in the Course is Used to Assess Program Outcomes:
   N/A

Prepared by: Jacob Somervell
Date: June 23, 2009
COURSE DESCRIPTION

Course Number: SWE 3210

Course Title: Software Quality Assurance

Hours: 3

Required/Elective? Required

Current Catalog Description: Inspection techniques, comprehensive test strategies, test coverage analysis, software process evaluation, process documentation, defect management, training documentation, inspection management tools


Class/laboratory schedule, ie, number of sessions per week, duration of sessions: Two sessions per week, 75 minutes each.

References: On-line references suggested by textbook author.

Course Outcomes: Students will know what software quality is, and what the practice of software quality assurance involves. This includes the knowledge of different software development strategies, as well as review types, testing, and maintenance. Students will know what CASE tools are, as well as their purpose. Students will also have knowledge of software configuration management and realizing its importance. Last, a student will be able to apply concepts toward a semester-long project.

Relationship between Course Outcomes and Program Outcomes: The course outcomes best relate to the following program outcomes: 1) an ability to identify and analyze constraints and trade-offs for a specific problem; 2) an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools; 3) an ability to effectively communicate their solutions and impacts to the appropriate audience; and 4) be able to apply 1)-3) in a group setting. An example of meeting the first program outcome occurs when proposing a solution to a project, as declared by a customer; the students will need to be able to adapt and respond to the customer’s needs. The remaining project outcomes are also met through their project, as students will need to be able to develop correct solutions to address the customer’s problem(s), and be able to communicate their solutions to the customer. Finally, the project is assigned as a group project.

Prerequisites by Topic
Construction – programming and development at this point should be viewed as one portion of the software process
Testing/Verification – another portion of the software process; having some ideas of testing is necessary to help show why software quality assurance is important

Major Topics Covered in the Course:
Software Quality and Quality Factors
SQA Components, Etc.
Software Development Processes
Reviews
Software Testing Strategies
High Quality
External Participants
CASE Tools
Procedures and Work Instructions
Templates and Checklists
Staff Certification
Configuration Management
Risk Management
Metrics
Actors and Management
Challenges to SQA

Assessment Plan for the Course: Assignments and Exams

How Data in the Course is Used to Assess Program Outcomes: N/A

Prepared by: Robert Hatch
Date: June 23, 2009
COURSE DESCRIPTION

Course Number:  SWE 3220

Course Title:  Software Requirements and Modeling

Hours:  3

Required/Elective?  Required

Current Catalog Description:  Elicitation of requirements, prioritization, goal analysis, validation criteria, feasibility analysis, ambiguity and clarity, functional and nonfunctional requirements, representation techniques, quality attributes, negotiations, feature interactions.


Class/laboratory schedule, ie, number of sessions per week, duration of sessions:  Two sessions per week, 75 minutes each.

References:  On-line references suggested by textbook author, Association of Computing Machinery’s (ACM) web site for paper presentations.

Course Outcomes:  Students will have a base knowledge of various modeling techniques, such as use case modeling, activity diagrams, and mapping class collaborations, for example.  Upon completion of the course, students will have a better understanding of the evolution of requirements gathering and analysis, from the elicitation of requirements from the customers to a refinement and elaboration of requirements.  Students will be able to apply the concepts through the development of a semester-long project.

Relationship between Course Outcomes and Program Outcomes:  The course outcomes best relate to the following program outcomes:  1) an ability to identify and analyze constraints and trade-offs for a specific problem; 2) an ability to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools; 3) an ability to effectively communicate their solutions and impacts to the appropriate audience; and 4) the ability to 1)-3) in a team setting.  An example of meeting the first program outcome occurs when examining collaborations between classes.  An example of the remaining program outcomes occur within the context of a student’s semester-long project.

Prerequisites by Topic
Software Construction Technologies – having a basis for good programming skills and tools used when developing software.
Software Engineering – having a familiarity with the concept of requirements and the purpose that requirements serve in software development.
Data Structures – basic knowledge is necessary for what is contained within a class, working with class hierarchies, and interactions with other classes.

Major Topics Covered:
Software Process
Requirements Analysis
Software Requirements Elicitation
Requirements Specification
Advanced Analysis of Requirements
System Design
User Interface Design
Requirements and Databases
Program and Transaction Design
Testing and Change Management

Assessment Plan for the Course: Exams and Project

How Data in the Course is Used to Assess Program Outcomes: N/A

Prepared by: Robert Hatch
Date: June 23, 2009
COURSE DESCRIPTION

Course Number:
SWE 3230

Course Title:
Software Configuration Management

Hours:
3

Required/Elective?
Required

Current Catalog Description
Configuration management, configuration tools, maintenance standards, verification and validation documentation, contract specifications, software library maintenance, project version interaction

Textbook:
Jessica Keyes, Software Configuration Management

Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
MWF (50mins)

References
Configuration Management Principles and Practice (Agile Software Development Series) by Anne Mette Jonassen Hass

Course Outcomes:
Proper application of software configuration management is a key component in the development of quality software. The objective of this course is to provide a solid understanding of Configuration management functions such as Configuration Identification, Configuration Control, Configuration Status Accounting, configuration auditing, release management and delivery. Student will also learn configuration tools, maintenance standards, verification and validation documentation, contract specifications, software library maintenance, and project version interaction.

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:
PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.
PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.
PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.
PO5: It will provide the students with opportunities for team based work and experience.
Prerequisites by Topic: None

Major Topics Covered in the Course

- Introduction to Software Configuration Management
- Project Management in a CM Environment
- Configuration Identification
- Configuration Control
- Configuration Status Accounting
- Configuration Management and Data Management
- Configuration Change Management
- Metrics and Configuration Management Reference
- Configuration tools
- Maintenance standards
- Verification and validation documentation
- Contract specifications
- Software library maintenance
- Project version interaction.jsp

Assessment Plan for the Course

Pop up Quizzes, configuration management projects, Tests, use of CM tools, exams

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a Software Engineering program

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0.5</td>
<td>0.5</td>
<td>Software Design</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Data Structures</td>
<td>0.5</td>
<td>0.5</td>
<td>Concepts of programming languages</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Prepared by: Dr. Abrar A. Qureshi
Date: June 23, 2009
COURSE DESCRIPTION

Course Number:
SWE 4240

Course Title:
Software Project Management

Hours:
3

Required/Elective?
Required

Current Catalog Description
Project planning and documentation, management tools, cost estimation, productivity, metrics, options and risks, expectations management, contracts, intellectual property, process standards, long term maintenance, progress measurement, earned value analysis, legal document management, project management standards.

Textbook:
Information Systems Project Management: A Process and Team Approach, by Fuller/Valacich/George

Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
MWF (50mins)

References
Effective Software Project Management by Robert K. Wysocki, Ph.D.

Course Outcomes:
Proper application of software configuration management is a key component in the development of quality software. The objective of this course is to provide a solid understanding of Configuration management functions such as Configuration Identification, Configuration Control, Configuration Status Accounting, configuration auditing, release management and delivery. Student will also learn configuration tools, maintenance standards, verification and validation documentation, contract specifications, software library maintenance, and project version interaction.

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:
PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.
PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.
PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

PO5: It will provide the students with opportunities for team based work and experience.

**Prerequisites by Topic:** None

**Major Topics Covered in the Course**
- Introduction to Project Management
- The Project Management Life Cycle
- Managing Project Teams
- Managing Project Communication
- Managing Project Scope
- Managing Project Scheduling
- Managing Project Resources
- Managing Project Quality
- Managing Project Risk
- Managing Project Procurement
- Managing Project Execution
- Managing Project Control and Closure

**Assessment Plan for the Course**
Pop up Quizzes, Project Management projects, Tests, use of MS Project , Exams

**How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)** We rely on our departmental examination, which covers the same course content as our required courses.

For a Software Engineering program

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0.5</td>
<td>0.5</td>
<td>Software Design</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Data Structures</td>
<td>0.5</td>
<td>0.5</td>
<td>Concepts of programming languages</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Prepared by: Dr. Abrar A. Qureshi
Date: June 23, 2009
COURSE DESCRIPTION

Course Number: SWE 4980

Course Title: Capstone Project I

Hours: 3

Required/Elective? Required

Current Catalog Description
Group project utilizing and demonstrating all software engineering skills of the major requirements. Student conceived and marketed product is developed with complete documentation, quality control, and configuration management.

Textbook:
Information Systems Project Management: A Process and Team Approach, by Fuller/Valacich/George

Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
MWF (50mins)

References
Effective Software Project Management by Robert K. Wysocki, Ph.D.
The Art of Software Testing, Glenford Myers, Wiley

Course Outcomes: Upon successful completion of this project, students should be able to:

- Work on a real life software development projects in Groups.
- Describe the main technical activities associated with software engineering: requirements elicitation, modeling, analysis and specification, architectural and detailed design specification, implementation, testing and maintenance.
- Provide sufficient knowledge for a student to be able to choose development techniques, tools and life-cycle models for a given project.
- Give students an understanding of the importance of quality assurance, human factors, professional issues and project management in software development Give students an understanding of the importance of user involvement throughout the development process.
- Illustrate the role of CASE tools in software engineering.
- Demonstrate the need for and practice of effective communication skills, both oral and written.

Relationship between Course Outcomes and Program Outcomes
The above course outcomes are related to the program outcomes in the following manners:
PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.
PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

PO5: It will provide the students with opportunities for team based work and experience.

Prerequisites by Topic
None

Major Topics Covered in the Course
- Implementation of Project Management skills
- Execution of Software Development Life Cycle
- Project Planning, Requirement Analysis and Design
- Development, Test and Integration, Acceptance and Installation and Maintenance
- The Project Management Life Cycle
- Managing Project Teams, Communication, Scope, Scheduling, Resources, Quality, Risk, Procurement, Execution, Control and Closure

Assessment Plan for the Course
Pop up Quizzes, Project Management projects, Tests, use of MS Project, Exams

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4) We rely on our departmental examination, which covers the same course content as our required courses.

For a Software Engineering program

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0.5</td>
<td></td>
<td>Software Design</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>Data Structures</td>
<td>0.5</td>
<td></td>
<td>Concepts of programming languages</td>
<td>0.5</td>
<td></td>
</tr>
</tbody>
</table>

Prepared by: Dr. Abrar A. Qureshi
Date: June 23, 2009
COURSE DESCRIPTION

Course Number:
SWE 4990

Course Title:
Capstone Project II

Hours:
3

Required/Elective?
Required

Current Catalog Description
Group project utilizing and demonstrating all software engineering skills of the major requirements. Student conceived and marketed product is developed with complete documentation, quality control, and configuration management.

Textbook:
Information Systems Project Management: A Process and Team Approach, by Fuller/Valacich/George

Class/laboratory schedule, ie, number of sessions per week, duration of sessions:
MWF (50mins)

References
Effective Software Project Management by Robert K. Wysocki, Ph.D.

The Art of Software Testing, Glenford Myers, Wiley

Course Outcomes: Upon successful completion of this project, students should be able to:

- Work on a real life software development projects in Groups.
- Describe the main technical activities associated with software engineering: requirements elicitation, modeling, analysis and specification, architectural and detailed design specification, implementation, testing and maintenance.
- Provide sufficient knowledge for a student to be able to choose development techniques, tools and life-cycle models for a given project.
- Give students an understanding of the importance of quality assurance, human factors, professional issues and project management in software development
- Illustrate the role of CASE tools in software engineering.
- Demonstrate the need for and practice of effective communication skills, both oral and written.
Relationship between Course Outcomes and Program Outcomes

The above course outcomes are related to the program outcomes in the following manners:

PO1: It will increase their ability to identify and analyze constraints and trade-offs for a specific problem.

PO2: It will enable them to apply appropriate algorithms, design and development principles, and mathematical theory to correctly model, design, and implement solutions to problems using current techniques, skills, and tools.

PO4: It will also increase their ability to effectively communicate their solutions and impacts to the appropriate audience.

PO5: It will provide the students with opportunities for team based work and experience.

Prerequisites by Topic

None

Major Topics Covered in the Course

- Implementation of Project Management skills
- Execution of Software Development Life Cycle
- Project Planning, Requirement Analysis and Design
- Development, Test and Integration, Acceptance and Installation and Maintenance
- The Project Management Life Cycle
- Managing Project Teams, Communication, Scope, Scheduling, Resources, Quality, Risk, Procurement, Execution, Control and Closure

Assessment Plan for the Course

Pop up Quizzes, Project Management projects, Tests, use of MS Project, Exams

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

We rely on our departmental examination, which covers the same course content as our required courses.

For a Software Engineering program

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0.5</td>
<td></td>
<td>Software Design</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Data Structures</td>
<td>0.5</td>
<td>0.5</td>
<td>Concepts of programming languages</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Prepared by: Dr. Abrar A. Qureshi
Date: June 23, 2009
# APPENDIX E - COURSE EVALUATION FORM

**THE UNIVERSITY OF VIRGINIA'S COLLEGE AT WISE**

**Course Evaluation Form**

- **Semester:** SPRING 2009
- **Faculty Name:**
- **Course:**
- **Section:**

### 1. Gender
- ○ Female
- ○ Male

### 2. Classification
- ○ Freshman (0-25 hours)
- ○ Sophomore (20-49 hours)
- ○ Junior (60-89 hours)
- ○ Senior (90+ hours)

### 3. Cumulative GPA
- ○ Under 2.00
- ○ 2.00-2.50
- ○ 2.51-3.00
- ○ Over 3.00

### 4. I am taking this class
- ○ As a core class
- ○ For my major
- ○ For my minor
- ○ As an elective
- ○ Under 10
- ○ 11-20
- ○ 21-30
- ○ 31-40
- ○ Over 40

### Statement

<table>
<thead>
<tr>
<th>Statement</th>
<th>Strongly Agree</th>
<th>Agree</th>
<th>Neutral</th>
<th>Disagree</th>
<th>Strongly Disagree</th>
<th>Not Applicable</th>
</tr>
</thead>
<tbody>
<tr>
<td>The professor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Appears to know the subject matter</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>7. Was accessible outside the class</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>8. Made course requirements clear in the syllabus</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>9. Taught with enthusiasm</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>10. Used effective strategies</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>11. Gave clear presentations</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>12. Treated students fairly</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>13. Appeared well prepared</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>14. Increased my understanding of the subject</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>15.Used time effectively</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>16. Returned tests, papers in a timely fashion</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
</tbody>
</table>
APPENDIX F – INSTITUTIONAL SUMMARY

The institution may employ any means it chooses to represent itself to ABET and the visiting team. Consequently, the references to specific tables in the following are for guidance only. The information may be presented in any manner the institution chooses.

A. The Institution

1. Name and Address of the Institution
   The University of Virginia’s College at Wise
   1 College Ave
   Wise, VA 24293

2. Name and Title of the Chief Executive Officer of the Institution
   David J. Prior, Chancellor

B. Type of Control
Description of the type of managerial control of the institution, e.g., private-non-profit, private-other, denominational, state, federal, public-other, etc.

The University of Virginia’s College at Wise is a 4-year, state, liberal arts college.

C. History of Institution
Provide a brief history of the Institution, its origin, and its development.

Our Heritage
The University of Virginia’s College at Wise, formerly Clinch Valley College of the University of Virginia, is the only four-year, state-supported college in far Southwest Virginia and the only branch of the University of Virginia.

Situated on 367 acres in Wise County, Virginia, UVa-Wise is an incredible example of the determination and perseverance of the Southwest Virginia region. Before the College was created, there were no public colleges in Virginia west of Radford. Higher education was simply out of reach for most residents of Virginia's mountains.

When local residents made their case to the University of Virginia for establishing a college in Wise, the Commonwealth of Virginia supported the cause by offering $5,000 to open, staff, and operate the two-year school for one year. Another $5,000 was appropriated for the second year, if there was to be a second year.

Local citizens donated over $6,000 to furnish and equip the classrooms. Wise County donated
property and two sandstone buildings, remnants of the county poor farm, to house the first classes. All of this happened in the winter of 1954. In September of that same year, Clinch Valley College opened its doors to 100 freshmen.

The first classes of graduates went on to become some of the region's most successful professionals. As more graduates of the two-year college expressed their preference to stay closer to home to complete their baccalaureate degrees, the College began the process to become a senior institution. The College first granted Bachelor of Arts degrees in June 1970. Bachelor of Science degrees were first awarded in 1973. Bachelor of Science in Nursing degrees were first awarded in 1996. Approximately 6,000 students have received baccalaureates in one or more of the College's academic disciplines.

The small college once nicknamed "Poor Farm University" is now a vibrant senior institution ranked by U.S. News & World Report as one of the South's top public liberal arts colleges. Perhaps the most significant event in the College's recent history is its adoption of a new name. After extensive research and consultation with the College's many constituencies, the College's governing board decided a name change was needed to more accurately reflect the institution's association with The University of Virginia. On July 1, 1999, following passage of legislation by the Virginia General Assembly, Clinch Valley College became The University of Virginia's College at Wise.

In recent years, UVa-Wise has broadened its outreach. In 1998, the College forged a sister institution agreement with Istanbul University, Turkey's oldest and largest institution of higher learning. An agreement was reached with Dumlupinar University in Kutahya, Turkey in 1999. Cooperative programs beneficial to the students and faculty of the institutions will be developed, including mutual exchanges. A third sister institution agreement with the University of Seville in Spain was signed in December 2000.

<taken from http://www.uvawise.edu/about/her_mission.html>

D. Student Body
Briefly describe the student body and where the students come from.

Student Body

<table>
<thead>
<tr>
<th>PRIMARY SERVICE REGION</th>
<th>2007 Median Household Income</th>
</tr>
</thead>
<tbody>
<tr>
<td>U.S.</td>
<td>$ 50,740</td>
</tr>
<tr>
<td>Virginia</td>
<td>$ 59,575</td>
</tr>
<tr>
<td>Fairfax Co.</td>
<td>$104,984</td>
</tr>
</tbody>
</table>

Buchanan, Dickenson, Lee, Norton, Russell, Scott, Tazewell and Wise

VIRGINIA'S COALFIELDS
Coalfields $ 31,233

2007 Poverty Rate

<table>
<thead>
<tr>
<th>Region</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>U.S.</td>
<td>13.0%</td>
</tr>
<tr>
<td>Virginia</td>
<td>9.9%</td>
</tr>
<tr>
<td>Fairfax Co.</td>
<td>4.9%</td>
</tr>
<tr>
<td>Coalfields</td>
<td>20.0%</td>
</tr>
</tbody>
</table>

Composite Index, 2008-2010

<table>
<thead>
<tr>
<th>Region</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fairfax Co.</td>
<td>.7650</td>
</tr>
<tr>
<td>Richmond</td>
<td>.4272</td>
</tr>
<tr>
<td>Coalfields Range</td>
<td>.1552 - .3095</td>
</tr>
</tbody>
</table>

Lowest in VA:
- Lee .1552
- Wise .1798
- Scott .1849

% of Adults, College Grad or Higher

<table>
<thead>
<tr>
<th>Region</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>U.S.</td>
<td>27.0%</td>
</tr>
<tr>
<td>Virginia</td>
<td>32.9%</td>
</tr>
<tr>
<td>Fairfax Co.</td>
<td>58.4%</td>
</tr>
</tbody>
</table>

Coalfields Average 10.7% (range 6.7%-14.0%)

STUDENTS

- 1964 total headcount, Fall 2008
- 95% in-state, Fall 2008
- 717 first generation, Fall 2008 (22.1% increase from Fall 2007)
- 53.5% from the Coalfields, Fall 2008
- 12% minority, Fall 2008
- 82.4% of students who applied for financial aid consideration in 2007-08 were needy. (81.9% in 2006/07)
- 57.0% of students who applied for financial aid consideration in 2007-2008 demonstrated eligibility for a Federal Pell Grant.
- 28% of students applying for aid in 2007-2008 demonstrated a $0 Expected Family Contribution: the family could pay nothing toward the student’s educational expenses. A majority of these students demonstrated incomes that put them at or below the poverty level. (23% for 2006/07)

E. Regional or Institutional Accreditation

Name the organizations by which the institution is currently accredited and the dates of initial and most recent accreditation evaluations.

Southern Association of Colleges and Schools (SACS)

- Initial accreditation: 1970
- Most recent evaluation: 2006
- Next affirmation: 2016
F. Personnel and Policies

Summarize the following elements

The promotion and tenure system

Faculty rank and promotion in rank are based on criteria proposed by the Faculty Senate and adopted by the College Board. Rank is granted on the basis of academic qualifications, years of experience, and demonstrated achievement. The criteria stated below will serve as guides in assigning academic rank. Exceptions to these criteria may be made in situations where the candidate shows outstanding merit based upon (a) academic qualifications; (b) experience; or (c) demonstrated achievement in teaching effectiveness, research/creative activity, and college/public service related to the faculty member's discipline.

The Department Chair and departmental faculty within each academic department must draft specific, written criteria for determining what constitutes "demonstrated achievement" for each rank. These written criteria must conform to minimum College standards listed below and must be distributed to each faculty member. Departmental guidelines for all ranks must be recommended by the Department Chair to the Academic Dean, and approved by the Provost.

Assistant Professor

1. Academic Qualifications: An earned doctorate or terminal degree appropriate for the discipline.
2. Effective achievement in classroom instruction and student advising. The faculty member should have a demonstrable desire to improve teaching ability and a program for development of instructional skills.
3. Demonstrable evidence in support of the faculty member's intentions to engage in research/creative/professional activity as well as intentions to design and complete such research/creative/professional activity, including course development.
4. Evidence of ability to serve the institution through College committees, participation in College and departmental programs and activities, and community service.

Associate Professor

1. Academic Qualifications: An earned doctorate or terminal degree appropriate for the discipline.
2. Experience: A minimum of six years service at the rank of assistant professor.
3. Superior achievement in classroom instruction and student advising. The faculty member should be willing to provide assistance to junior faculty members.
4. Demonstrable evidence in support of research/creative/professional activity including, but not limited to, articles published, books written, papers
presented, courses developed, dramatic productions staged, shows given, concerts and performances given.
5. Evidence of appropriate service to the institution through service on College committees, participation in College and departmental programs and activities, and service to the community which includes an appreciation of the unique relationship between the College and the local community. The faculty member should be progressing toward becoming a respected member of the College community.

Professor

1. Academic Qualifications: An earned doctorate or terminal degree appropriate for the discipline.
2. Experience: A minimum of six years service at the rank of associate professor.
3. Excellence in classroom instruction and student advising. The faculty member should set a clear example of teaching excellence.
4. Demonstrable evidence in support of research/creative/professional activity including, but not limited to, articles published, books written, papers presented, courses developed, dramatic productions staged, shows given, concerts and performances given.
5. Evidence of leadership and appropriate service to the institution through service on College committees, participation in College and departmental programs and activities, and service to the community which includes an appreciation of the unique relationship between the College and the local community. The faculty member should be a respected member of the College community.

The process used to determine faculty salaries
Salaries for instructors in software engineering and computer science are determined outside the official College salary scale. Since these programs are supported by legislative grants and other funding sources that are not available to other College programs, we are able to pay instructors salaries that are competitive with similar institutions on the national level. This means that instructors in software engineering and computer science at UVa-Wise earn considerably more than their colleagues in disciplines that are driven by less favorable market forces. In each instance in which a faculty search was carried out and an offer made, a careful study was done to determine an appropriate beginning salary. This included consultation with colleagues in institutions with similar programs and studies of average salaries for persons of similar academic training and experience. Appropriate experience in a corporate setting in the field of information technology or engineering was also considered.

Faculty benefits
UVa-Wise complies with the policy of the Commonwealth of Virginia, which provides that benefits should be available to certain employees. The policy specifies that full-time, salaried faculty with appointments of six months or more are eligible for retirement, life insurance, disability insurance and health care benefit programs.

Part-time salaried faculty with a term of six months or more and currently working at least 50 percent effort (20 hours per week minimum) are eligible for retirement, life insurance and disability insurance.

Part-time and wage faculty are not eligible for benefits.

Visiting faculty members who are employed for six months or more and are full-time are eligible for benefits.

G. Educational Unit
Describe the educational unit (see General Instructions). Describe the administrative chain of responsibility from the individual responsible for the program to the chief executive officer of the institution. Include names and titles. An organization chart may be included.

The Department of Mathematics and Computer Science offers courses in mathematics, computer science, and software engineering, leading to majors in mathematics, computer science, software engineering, and management information systems. Minors may be earned in mathematics and computer science. General education courses and courses designed for students majoring in other disciplines also are provided.

The administrative chain of responsibility for the Department of Mathematics and Computer Science starts with the Department Chair. The Department Chair reports to the Provost and Senior Vice Chancellor. The Provost and Senior Vice Chancellor reports to the Chancellor.

Graphically, we have:
H. Credit Unit
It is assumed that one semester or quarter credit normally represents one class hour or three laboratory hours per week. One academic year normally represents at least 28 weeks of classes, exclusive of final examinations. If other standards are used for this program, the differences should be indicated.

Further, in cases where the Criteria specify curricular content in terms of years, one year is equivalent to either 32 semester hours (48 quarter hours) or the quotient of the number of credits required for graduation divided by the nominal length of the program in years, whichever is less. Thus, for programs with 128 semester hours (192 quarter credits) or greater, one year is 32 semester hours (48 quarter hours). For programs with less than 128 semester hours (192 quarter credits), one year is the number of credits required for graduation divided by the nominal length of the program in years.

    One semester credit represents one class hour or three laboratory hours per week. One academic year represents 30 weeks of classes, exclusive of final examinations.

I. Instructional Modes
If modes other than traditional on-campus instruction are employed in any programs, the additional modes of instruction should be listed and described in relation to the applicable programs. The institutional and/or unit policies under which the alternate modes are offered should be summarized.

    NA

J. Grade-Point Average
Indicate the grade-point average required for graduation. If there are differences in requirements among the regular and alternative instructional modes, please explain.
2.0 on a 4.0 scale is required for graduation

K. Academic Supporting Units
Provide information about units that teach courses required by the programs being evaluated, e.g., mathematics, physics, etc. Include names and titles of the individuals responsible for these units.

Mathematics – Department of Mathematics and Computer Science, Alex Edwards, Chair. cae@uvawise.edu, 276-376-4568, fax: 276-376-4589

Physics and Chemistry – Department of Natural Sciences, Margie Tucker, Chair. mat5w@uvawise.edu, 276-328-0224

L. Non-Academic Supporting Units
Provide information about units that provide non-academic support to the programs being evaluated, e.g., library, computing facilities, placement, tutoring, etc. Include names and titles of the individuals responsible for these units.

Library – Robin Benke, Director, rpb@uvawise.edu, 276-328-0151

Office of Information Technology (IT) – Keith Fowlkes, Vice Chancellor of Information Technology, jkf7e@uvawise.edu, 276-376-4578

Student support services – Marcia Mitchell, Director of Student Support Services, myb3u@uvawise.edu, 276-328-0177

Enrollment management – Rusty Necessary, Vice Chancellor of Enrollment Management, rdn2f@uvawise.edu, 276-328-0322

Center for Student Development – Lelia Bradshaw, Assistant Director, lab8q@uvawise.edu, 276-328-0131

M. Faculty Workload
Describe the faculty workload policy. Define what constitutes a full-time load.

The normal full-time teaching load for faculty members is 12 semester hours. Faculty members are also expected to participate in the work of their departments outside of the classroom, to provide academic advising to students, to serve in governance of the College, to develop new courses as needed, and to engage in scholarly activities. Teaching loads may be reduced for faculty engaged in special activities which require an unusual amount of time. Every administrator with appropriate qualifications is considered a part of the teaching faculty and may be expected to teach at least one three-hour course each semester.

N. Tables
The tables that follow are simply a guide and are not required in the Self-Study Report. All are optional. The institution is encouraged to employ any means it chooses to represent itself to ABET and the visiting evaluation team.
Table D-1. Programs Offered by the Educational Unit

<table>
<thead>
<tr>
<th>Program Title</th>
<th>Day</th>
<th>Cooperative Education</th>
<th>Off Campus</th>
<th>Alternate Mode</th>
<th>Nominal Years to Complete</th>
<th>Administrative Head</th>
<th>Administrative Unit or Units Exercising Budgetary Control</th>
<th>Submitted for Evaluation</th>
<th>Offered, Not Submitted for Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bachelor of Science in Computer Science</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>Alex Edwards</td>
<td>Gil Blackburn Provost</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Bachelor of Science in Software Engineering</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>Alex Edwards</td>
<td>Gil Blackburn Provost</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Bachelor of Science in Management Information Systems</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td>4</td>
<td>Alex Edwards</td>
<td>Gil Blackburn Provost</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Bachelor of Science in Mathematics</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>Alex Edwards</td>
<td>Gil Blackburn Provost</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Bachelor of Arts in Mathematics</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>Alex Edwards</td>
<td>Gil Blackburn Provost</td>
<td>Y</td>
<td></td>
</tr>
</tbody>
</table>
List the titles of all degrees offered by the educational unit responsible for the programs being evaluated, undergraduate and graduate, granted by the institution. If there are differences in the degrees awarded for completion of cooperative education programs, these should be clearly indicated.

1 Give program title as shown on a graduate’s transcript
2 Indicate all modes in which the program is offered. If separate accreditation is requested for an alternative mode, list on a separate line. Describe “Other” by footnote.
3 Only those programs being submitted at this time for reaccreditation (now accredited) or initial accreditation (not now accredited) should be checked in this column.
4 Programs not submitted for evaluation at this time should be checked in this column.
## Table D-2. Degrees Awarded and Transcript Designations by Educational Unit

<table>
<thead>
<tr>
<th>Program Title</th>
<th>Modes Offered</th>
<th>Name of Degree Awarded</th>
<th>Designation on Transcript</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bachelor of Science in Computer Science</td>
<td>Y</td>
<td>Bachelor of Science in Computer Science</td>
<td>Bachelor of Science in Computer Science</td>
</tr>
<tr>
<td>Bachelor of Science in Software Engineering</td>
<td>Y</td>
<td>Bachelor of Science in Software Engineering</td>
<td>Bachelor of Science in Software Engineering</td>
</tr>
<tr>
<td>Bachelor of Science in Management Information Systems</td>
<td>Y</td>
<td>Bachelor of Science in Management Information Systems</td>
<td>Bachelor of Science in Management Information Systems</td>
</tr>
<tr>
<td>Bachelor of Science in Mathematics</td>
<td>Y</td>
<td>Bachelor of Science in Mathematics</td>
<td>Bachelor of Science in Mathematics</td>
</tr>
<tr>
<td>Bachelor of Arts in Mathematics</td>
<td>Y</td>
<td>Bachelor of Arts in Mathematics</td>
<td>Bachelor of Arts in Mathematics</td>
</tr>
</tbody>
</table>

Complete the table for all programs, as follows:

1. Give the program title as officially published in catalog.
2. Indicate all modes in which the program is offered. If separate accreditation is requested for an alternative mode, list on a separate line. Describe “Other” by footnote.
3. List degree awarded for each mode offered. If different degrees are awarded, list on separate lines.
4. Indicate how the program is listed on transcript for each mode offered. If different designations are used, list on separate lines.
### Table D-3. Support Expenditures

Department of Mathematics and Computer Science

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Dept. allocation - Other than Personal Services (OTPS)²</td>
<td>76000</td>
<td>76000</td>
<td>76000</td>
</tr>
<tr>
<td>OTPS Spent</td>
<td>65000</td>
<td>46000</td>
<td>*</td>
</tr>
<tr>
<td>Operations (not including staff)</td>
<td>39000</td>
<td>23000</td>
<td>23000</td>
</tr>
<tr>
<td>Travel</td>
<td>13000</td>
<td>9900</td>
<td>10000</td>
</tr>
<tr>
<td>Equipment and Supplies</td>
<td>47000</td>
<td>55000</td>
<td>117000</td>
</tr>
<tr>
<td>(a.1) Institutional Funds - dept</td>
<td>17000</td>
<td>17000</td>
<td>77000³</td>
</tr>
<tr>
<td>(a.2) Equipment Trust Fund</td>
<td>30000</td>
<td>30000</td>
<td>30000</td>
</tr>
<tr>
<td>(b) Grants and Gifts</td>
<td>0</td>
<td>8000</td>
<td>10000</td>
</tr>
<tr>
<td>Part-time Assistance (other than teaching)</td>
<td>8000</td>
<td>8000</td>
<td>8000</td>
</tr>
<tr>
<td>Faculty Salaries</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Department</td>
<td>740000</td>
<td>761000</td>
<td>765000</td>
</tr>
<tr>
<td>Computer Science and Software Engineering</td>
<td>374000</td>
<td>385000</td>
<td>385000</td>
</tr>
</tbody>
</table>

Report Department Level and Program Level data for each program being evaluated. Updated tables are to be provided at the time of the visit.

² OTPS does not include any personnel dollars

³ allocated but not able to spend because of state budget crisis

Travel can come from Provost Travel budget, Department budget, Faculty Development Committee, and Foundation

³ includes a one-time $60,000 (State budget item) to furnish two new computer labs

Equipment Trust Fund (ETF): money provided by the State each year for equipment

Undergraduate student assistants primarily paid with Federal Work-Study dollars
Table D-4. Personnel and Students

Bachelor of Science in Software Engineering

Year\(^1\): __2008____

<table>
<thead>
<tr>
<th>HEAD COUNT</th>
<th>FTE2</th>
<th>RATIO TO FACULTY(^3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT</td>
<td>PT</td>
<td></td>
</tr>
<tr>
<td>Administrative(^4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Faculty (tenure-track)</td>
<td>7 (5 CS/SE)</td>
<td>7</td>
</tr>
<tr>
<td>Other Faculty (excluding student Assistants)</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Student Teaching Assistants</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Student Research Assistants</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Technicians/Specialists</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Office/Clerical Employees</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Others(^5)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Undergraduate Student enrollment(^6)</td>
<td>800 (w/ F&amp;S)</td>
<td>800</td>
</tr>
</tbody>
</table>

Report data for the program unit(s) and for each program being evaluated.

1 Data on this table should be for the fall term immediately preceding the visit. Updated tables for the fall term when the ABET team is visiting are to be prepared and presented to the team when they arrive.
2 For student teaching assistants, 1 FTE equals 20 hours per week of work (or service). For undergraduate and graduate students, 1 FTE equals 15 semester credit-hours (or 24 quarter credit-hours) per term of institutional course work, meaning all courses — science, humanities and social sciences, etc. For faculty members, 1 FTE equals what your institution defines as a full-time load.
3 Divide FTE in each category by total FTE Faculty. Do not include administrative FTE.
4 Persons holding joint administrative/faculty positions or other combined assignments should be allocated to each category according to the fraction of the appointment assigned to that category.
5 Specify any other category considered appropriate, or leave blank.
6 Specify whether this includes freshman and/or sophomores.
Table 5. Program Enrollment and Degree Data
Department of Mathematics and Computer Science

<table>
<thead>
<tr>
<th>Academic Year</th>
<th>Enrollment Year</th>
<th>Total Undergrad</th>
<th>Total Grad</th>
<th>Degrees Conferred</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st</td>
<td>2nd</td>
<td>3rd</td>
<td>4th</td>
</tr>
<tr>
<td>CURRENT (2008-2009)</td>
<td>FT</td>
<td>26</td>
<td>24</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>5</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>1 (2007-2008)</td>
<td>FT</td>
<td>30</td>
<td>25</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>11</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>2 (2006-2007)</td>
<td>FT</td>
<td>29</td>
<td>20</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>21</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>3 (2005-2006)</td>
<td>FT</td>
<td>30</td>
<td>14</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>16</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>17</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>5 (2003-2004)</td>
<td>FT</td>
<td>23</td>
<td>16</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>12</td>
<td>10</td>
<td>4</td>
</tr>
</tbody>
</table>

Give official fall term enrollment figures (head count) for the current and preceding five academic years and undergraduate and graduate degrees conferred during each of those years. The "current" year means the academic year preceding the fall visit.
## Table 5. Program Enrollment and Degree Data

### Software Engineering

<table>
<thead>
<tr>
<th>Academic Year</th>
<th>Enrollment Year</th>
<th>Total Undergrad</th>
<th>Total Grad</th>
<th>Degrees Conferred</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st 2nd 3rd 4th 5th</td>
<td>Bachelor</td>
<td>Master</td>
<td>Doctor</td>
</tr>
<tr>
<td>CURRENT (2008-2009)</td>
<td>FT 4 3 3 - 1</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT - - - - -</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1 (2007-2008)</td>
<td>FT 4 4 - 1 -</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT 1 - - - -</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2 (2006-2007)</td>
<td>FT 5 1 - - -</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT 1 - - - -</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>3 (2005-2006)</td>
<td>FT 1 - - - -</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT - - - - -</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>4 (2004-2005)</td>
<td>FT - - - - -</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT - 1 - - -</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>5 (2003-2004)</td>
<td>FT - - - - -</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT 1 - - - -</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Give official fall term enrollment figures (head count) for the current and preceding five academic years and undergraduate and graduate degrees conferred during each of those years. The "current" year means the academic year preceding the fall visit.

FT—full time PT—part-time *one student is completing requirements in Summer of 2009
Faculty Salary Data removed from public version.