Program Assessment Report

Bachelor of Science in Computer Science

2006-2007

Compiled by

B.A. Juliano, Ph.D.
Assessment Coordinator
Department of Computer Science
College of Engineering, Computer Science, and Construction Management
California State University, Chico
Corner First and Warner
Chico, CA 95929-0410

Assessment Committee Members, 2006-2007:

B. Britton, Ph.D.
M. Fayek, M.S. (ex-officio, Dept. Chair)
Table of Contents

I. Embedded Assessment Measures, A.Y. 2006-2007..........................................................6
   A. Executive Summary...........................................................................................................6
   B. Areas of Concern............................................................................................................6
   C. Areas on Target.............................................................................................................7

II. Detailed Assessment of Program Outcomes...................................................................8
   A. Embedded Assessment Measures...................................................................................8
      1. An ability to apply knowledge of math, science and computing...............................8
      2. An ability to design and implement programs as well as to analyze and interpret code and data..................................................................................................................9
      3. An ability to design a system, component, or process to meet desired needs...........9
      4. An ability to function on multi-disciplinary teams....................................................10
      5. An ability to identify, formulate and solve computing problems............................11
      6. An understanding of professional and ethical responsibilities...............................12
      7. An ability to communicate effectively.......................................................................13
      8. The broad education necessary to understand the impact of computing solutions in a global and societal context.........................................................................................14
      9. A recognition of the need for, and an ability to engage in, life-long learning...........14
     10. A knowledge of contemporary issues.........................................................................15
     11. An ability to use the techniques, skills, and modern computing tools necessary for computing practice..............................................................................................................15
   B. Proposed Actions from Outcomes................................................................................17
   C. Other Direct Measures..................................................................................................18
   D. Indirect Measures..........................................................................................................20
      1. College of ECC Senior Exit Survey............................................................................20
      2. Department of Computer Science Senior Exit Survey.............................................22
      3. Alumni Survey...........................................................................................................23
      4. Advisory Board Survey/Feedback.............................................................................23
      5. ACM Pacific Northwest Regional Programming Contest......................................23
   E. Proposed Revision of Measures, Metrics, or Outcomes..............................................24

III. Assessment of Program Educational Objectives.........................................................25
   A. Program Educational Objectives Summary................................................................25
   B. Alumni Survey Methodology and Results..................................................................25
   C. Employer Survey Methodology and Results...............................................................25
   D. Advisory Board Feedback.........................................................................................25
   E. Accreditation Evaluation Feedback............................................................................25
F. Program Improvement Resulting from Feedback ........................................... 25
G. Proposed Revisions to Program Objectives .................................................. 25

IV. Program Accreditation .................................................................................. 26

A. Summary of Accreditation Feedback ............................................................ 26
   1. Objectives and Assessments ..................................................................... 26
   2. Student Support ....................................................................................... 26
   3. Faculty ..................................................................................................... 26
   4. Curriculum ............................................................................................... 26
   5. Laboratories and Computing Facilities ................................................... 27
   6. Institutional Support and Financial Resources .......................................... 27
   7. Institutional Facilities ............................................................................... 27

B. Summary of Actions Taken to Address Issues ............................................. 27

V. Appendices .................................................................................................... 28

A. Program Outcomes Summary ................................................................. 28
B. Core Course Alignment / Program Outcomes Matrix .................................. 29
C. Timeline ..................................................................................................... 30
D. Embedded Assessment Measures ............................................................. 32
E. Summary of Results from the College of ECC Senior Exit Survey .............. 34
F. Summary of Results from the Department of Computer Science Senior Exit Survey ................................................................. 37
G. Measurement Standards (Rubrics) ............................................................. 40
H. Survey Instruments ...................................................................................... 48
   1. Educational Testing Services Major Field Test (MFT) in Computer Science ........................................................................... 48
   2. College of ECC Graduating Senior Survey ............................................. 59
List of Figures

Figure 1: Senior Exit Survey Trends for Program Outcome #1.................................................................7
Figure 2: Senior Exit Survey Trends for Program Outcome #2.................................................................8
Figure 3: Senior Exit Survey Trends for Program Outcome #3.................................................................9
Figure 4: Senior Exit Survey Trends for Program Outcome #4...............................................................10
Figure 5: Senior Exit Survey Trends for Program Outcome #5...............................................................11
Figure 6: Senior Exit Survey Trends for Program Outcome #6...............................................................11
Figure 7: Senior Exit Survey Trends for Program Outcome #7...............................................................12
Figure 8: Senior Exit Survey Trends for Program Outcomes #8 and #10....................................................13
Figure 9: Senior Exit Survey Trends for Program Outcome #9...............................................................14
Figure 10: Senior Exit Survey Trends for Program Outcome #11...........................................................15
Figure 11: MFT in Computer Science, Scores Statistics Trend..................................................................17
Figure 12: MFT in Computer Science, Assessment Indicators Trend......................................................18
Figure 13: Educational satisfaction – trends for questions that pertain to teaching..................................19
Figure 14: Educational satisfaction – trends for questions that pertain to courses....................................20
Figure 15: Educational satisfaction – trends for questions that pertain to laboratory access, equipment and facilities........................................................................................................20
Figure 16: Educational satisfaction – responses to a department exit survey from CSCI Majors that took CSCI 111 and CSCI 112 at Chico State.................................................................21
Figure 17: Educational satisfaction – responses to a department exit survey from CSCI Majors that did not take CSCI 111 and CSCI 112 at Chico State.........................................................................22
Figure 18: Chico State Team Placement Trends for the ACM Pacific Northwest Regional Programming Contest showing Top Team and Average Team placements.........................................................23
List of Tables
Table 1: Embedded assessment measures collected during 2006-2007................................................................. ..6
Table 2: MFT Assessment Indicators x Core Course Matrix.................................................................................... .19
Table 3: Core Course x Program Outcomes Matrix................................................................................................. 29
Table 4: Grouping of Program Outcomes into Subsets............................................................................................ 30
Table 5: Rotational Assessment Timeline.................................................................................................................. 31
Table 6: Embedded Assessment Measures and Metrics............................................................................................. 32
Table 7: Program Outcomes Trends for Computer Science........................................................................................ 34
Table 8: Educational Satisfaction Trends for Computer Science................................................................................. 35
Table 9: College of ECC Academic Year Assessment Cycle.......................................................................................... 36
Table 10: Agreement responses to the Department of Computer Science Senior Exit Survey, CSCI Majors that
took CSCI 111 and CSCI 112 from Chico State........................................................................................................ 38
Table 11: Agreement responses to the Department of Computer Science Senior Exit Survey, CSCI Majors that did
not take CSCI 111 and CSCI 112 from Chico State...................................................................................................... 39
Table 12: CSCI Program Outcomes x ABET Scoring Rubric Matrix............................................................................ 40

A. Executive Summary

The Department of Computer Science developed a staggered schedule for collecting embedded assessment measures for each of its program outcomes. The department’s eleven program outcomes were grouped into three subsets (see Tables 3 through 5 of the Appendix on pages 29-31). Table 1 lists the embedded assessment measures collected during Fall 2006 and Spring 2007.

Table 1: Embedded assessment measures collected during 2006-2007.

<table>
<thead>
<tr>
<th>Year/Sem</th>
<th>Subset</th>
<th>Program Outcome</th>
<th>Course</th>
<th>Assessment %</th>
<th>Score</th>
<th>Assessor</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006/Fa</td>
<td>1</td>
<td>1</td>
<td>CSCI 221</td>
<td>65.00</td>
<td>3.600</td>
<td>Britton</td>
</tr>
<tr>
<td>1 1</td>
<td>CSCI 320</td>
<td>74.00</td>
<td>3.960</td>
<td>Britton</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 2</td>
<td>CSCI 311</td>
<td>57.50</td>
<td>3.300</td>
<td>Stapleton</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 3</td>
<td>CSCI 340</td>
<td>69.23</td>
<td>3.769</td>
<td>Hilzer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 4</td>
<td>EECE 335</td>
<td>96.60</td>
<td>4.864</td>
<td>Hubbard</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 5</td>
<td>CSCI 340</td>
<td>92.31</td>
<td>4.692</td>
<td>Hilzer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2007/Sp</td>
<td>2 4</td>
<td>EECE 335</td>
<td>100.00</td>
<td>5.000</td>
<td>Hubbard</td>
<td></td>
</tr>
<tr>
<td>2 5</td>
<td>CSCI 311</td>
<td>82.14</td>
<td>4.286</td>
<td>Stapleton</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 6</td>
<td>CSCI 301</td>
<td>100.00</td>
<td>5.000</td>
<td>Fisk</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 7</td>
<td>CSCI 330</td>
<td>93.75</td>
<td>4.750</td>
<td>Luk</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 7</td>
<td>CSCI 490</td>
<td>100.00</td>
<td>5.000</td>
<td>Fayek</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In Table 1, scores falling at or above the department target rating of 3.80 are in green and scores falling below this target rating are bold and in red.

B. Areas of Concern.

There are three embedded assessment measures from 2006-2007 that fall below the department target rating of 3.80: Program Outcomes 1 (An ability to apply knowledge of math, science and computing), 2 (An ability to design and implement programs as well as to analyze and interpret code and data), and 3 (An ability to design a system, component, or process to meet desired needs).

For Fall 2006, embedded assessment measures for Program Outcome 1 was below the target for CSCI 221 and well above the target for CSCI 320. Referring to Table 6 (see page 32), this seems to indicate that students may be having difficulty with number systems, conversions for encoding/decoding data, and internal representations of (unsigned and signed) integers and floating point numbers.

Embedded assessment measures for Program Outcome 2 was below the target for CSCI 311 in Fall 2006. Referring to Table 6 (see page 32), this seems to indicate that emphasizing the connection between discrete structures and related algorithms needs to be improved or modified so that it is reflected in both the exam and in programming assignments.
In Fall 2006, Embedded assessment measures for Program Outcome 3 was below the target for CSCI 340. Referring to Table 6 (see page 32), this seems to indicate that students may not fully understand the association between the design and implementation of contemporary operating systems and their use/functionality.

C. Areas on Target.

Four of the seven embedded assessment measures from 2006-2007 are at or above the department target rating of 3.80: Program Outcomes 4 (An ability to function on multi-disciplinary teams), 5 (An ability to identify, formulate and solve computing problems), 6 (An understanding of professional and ethical responsibilities), and 7 (An ability to communicate effectively). Hence, there are no reasons for concern with these areas.
II. Detailed Assessment of Program Outcomes

A. Embedded Assessment Measures

The department’s schedule for collecting embedded assessment measures for its eleven program outcomes is summarized in Tables 3 through 5 of the Appendix on pages 29-31. The department’s embedded assessment measures and metrics are detailed in Table 6 of the Appendix on page 32.

On a scale of 1 (lowest) through 5 (highest), the department faculty chose a rating of 3.80 or above as the target rate for all embedded (in-class), direct (MFT), and indirect (senior survey) program outcome assessments. To transform embedded measures from the typical 0-100% scale used by faculty to the appropriate 1-5 scale, the following transformation is used:

\[
1 + 4 \times \frac{p}{n}
\]

where \( p \) is the number of students successfully achieving the measure, and \( n \) is the total number of students from which the measure was taken.

1. An ability to apply knowledge of math, science and computing.

   a) The department collected embedded assessment data for Program Outcome 1 in Spring and Fall 2006. The scores improved 11.17% from 3.40 to 3.78.

   b) Direct measurement data for Spring 2006 and Fall 2006 from MFT Assessment Indicator (AI) trends shown in Figure 12 on page 19 indicate a dramatic 42.85% improvement for AI 1 (Programming) from 45% (2.80) to 75% (4.00), and a 5.88% improvement for AI 2 (Discrete Structures and Algorithms) from 60% (3.40) to 65% (3.60). So, although there seems to be a concern for this Program Outcome relative to embedded assessment data from CSCI 221 last Fall 2006, the MFT trends coincide with the aggregated embedded assessment data trends for this program outcome.

![Figure 1: Senior Exit Survey Trends for Program Outcome #1.](image-url)

---

1 The expected rating of 3.80 is based on a 70% expected success rate, translating to \( 1 + 4 \times 0.70 = 3.80 \) in a 1-5 scale.
c) The College of ECC Senior Exit Survey Question #31 (Apply knowledge to solve problems) trends shown in Figure 1 had a 7.67% decline from a mean score of 3.91 last year to 3.61 (see Table 7 on page 34 in the Appendix) for this assessment cycle. The score for this year is also below the 3.80 department target.

2. An ability to design and implement programs as well as to analyze and interpret code and data.

a) The department collected embedded assessment data for Program Outcome 2 in Spring and Fall 2006. The scores declined 7.82% from 3.58 to 3.30.

b) Direct measurement data for Spring 2006 and Fall 2006 from MFT Assessment Indicator (AI) trends shown in Figure 12 on page 19 indicate a dramatic 42.85% improvement for AI 1 (Programming) from 45% (2.80) to 75% (4.00), and a 5.88% improvement for AI 2 (Discrete Structures and Algorithms) from 60% (3.40) to 65% (3.60).

c) Figure 2 indicates that scores for College of ECC Senior Exit Survey Question #32 (Design and conduct experiments) declined 15.60% from 3.91 to 3.30 and Question #33 (Analyze and interpret experimental data) declined 8.27% from 3.87 to 3.55 (see Table 7 on page 34 in the Appendix) for this assessment cycle. The scores for this year are below the 3.80 department target, and the decline for the score for Question #32 is the sharpest observed for all questions.

3. An ability to design a system, component, or process to meet desired needs.

a) The department collected embedded assessment data for Program Outcome 3 in Fall 2006. The score was 3.769 which is only 0.82% below the department target of 3.80.
b) Direct measurement data for Spring 2006 and Fall 2006 from MFT Assessment Indicator (AI) trends shown in Figure 12 on page 19 indicate a dramatic 42.85% improvement for AI 1 (Programming) from 45% (2.80) to 75% (4.00), a 5.88% improvement for AI 2 (Discrete Structures and Algorithms) from 60% (3.40) to 65% (3.60), and a 21.43% improvement for AI 3 (Systems: Architecture, Operating Systems, Networking, Database) from 45% (2.80) to 60% (3.40). Unfortunately, between Fall 2006 and Spring 2007, there was no change for AI 1, a 15.38% improvement for AI 2 from 65% (3.60) to 75% (4.00), and a 16.67% decline for AI 3 from 60% (3.40) to 50% (3.00).

![Figure 3: Senior Exit Survey Trends for Program Outcome #3.](image)

Figure 3: Senior Exit Survey Trends for Program Outcome #3.

4. An ability to function on multi-disciplinary teams.

a) The department collected embedded assessment data for Program Outcome 4 in Spring 2007. The scores improved 2.88% from 4.86 to 5.00.

b) None of the MFT Assessment Indicators directly assess this program outcome.
c) Figure 4 shows that College of ECC Senior Exit Survey Question #35 (Function on multidisciplinary team) received a mean score of 3.57 (see Table 7 on page 34 in the Appendix) for this assessment cycle, down 7.99% from 3.88 last year. The score for this year is also below the 3.80 department target and is the second lowest score received for this question since Spring 2002.

5. An ability to identify, formulate and solve computing problems.

a) The department collected embedded assessment data for Program Outcome 5 in Fall 2006 and Spring 2007. The scores declined 4.69% from 4.69 to 4.47.

b) Direct measurement data for Spring 2006 and Fall 2006 from MFT Assessment Indicator (AI) trends shown in Figure 12 on page 19 indicate a dramatic 42.85% improvement for AI 1 (Programming) from 45% (2.80) to 75% (4.00), a 5.88% improvement for AI 2 (Discrete Structures and Algorithms) from 60% (3.40) to 65% (3.60), and a 21.43% improvement for AI 3 (Systems: Architecture, Operating Systems, Networking, Database) from 45% (2.80) to 60% (3.40). Unfortunately, between Fall 2006 and Spring 2007, there was no change for AI 1, a 15.38% improvement for AI 2 from 65% (3.60) to 75% (4.00), and a 16.67% decline for AI 3 from 60% (3.40) to 50% (3.00).

---

2 Since both CSCI 311 (with 28 CSCI majors) and CSCI 340 (with 10 CSCI majors) were used to collect embedded assessment data for this program outcome this Spring 2007, the weighted average of these two scores was used to determine a combined score for this program outcome: \((28 \times 4.286) + (10 \times 5.00))/38 = 4.47\)
c) Figure 5 shows that College of ECC Senior Exit Survey Question #36 (Identify, formulate, solve technical problems) received a mean score of 3.83 (see Table 7 on page 34 in the Appendix) for this assessment cycle, down 2.05% from 3.91 last year. Although the score for this year is above the 3.80 department target, it is the lowest score received for this question since Spring 2002.

6. An understanding of professional and ethical responsibilities.

a) The department collected embedded assessment data for Program Outcome 6 in Spring 2007. The score of 5.00 is 31.58% above the 3.80 department target.

b) None of the MFT Assessment Indicators directly assess this program outcome.
c) Figure 6 shows that College of ECC Senior Exit Survey Question #39 (Understand professional, ethical responsibilities) received a mean score of 3.83 (see Table 7 on page 34 in the Appendix) for this assessment cycle, an improvement of 2.96% from 3.72 last year. Hopefully, this is a sign that the trend will continue to progress since there was a downward trend for this question from A.Y. 2003-2004 through last year.

7. An ability to communicate effectively.

a) Embedded assessment data was collected for Program Outcome 7 from CSCI 330 and CSCI 490 in Spring 2007. The score was 4.87\(^3\) which is 28.16% above the 3.80 department target.

b) None of the MFT Assessment Indicators directly assess this program outcome.

c) Figure 7 shows that College of ECC Senior Exit Survey Question #37 (Communicate technical matters in writing) received a mean score of 3.87 and Question #38 (Communicate technical matters orally) received a mean score of 3.83 (see Table 7 on page 34 in the Appendix) for this assessment cycle. Both scores are above the 3.80 department target, but the score for Question #7 is a slight decline of 0.27% from 3.88 last year, and the score for Question #8 is a slight 0.53% improvement from 3.81 last year. Hopefully, this is a sign that the trend for this program outcome will continue to progress.

---

\(^3\) Since both CSCI 330 (with 16) and CSCI 490 (with 14 CSCI majors) were used to collect embedded assessment data for this program outcome this Spring 2007, the weighted average of these two scores was used to determine a combined score for this program outcome: \(((16\times 4.75) + (14\times 5.00))/30 = 4.87\)
8. The broad education necessary to understand the impact of computing solutions in a global and societal context.

a) Embedded assessment data was not collected from CSCI 301 in A.Y. 2006-2007 for this program outcome.

b) None of the MFT Assessment Indicators directly assess this program outcome.

c) Figure 8 shows that College of ECC Senior Exit Survey Question #40 (Understand contemporary issues facing society) received a mean score of 3.57 (see Table 7 on page 34 in the Appendix) for this assessment cycle, down 7.03% from 3.84 last year. This is the sharpest decline observed for this score.

9. A recognition of the need for, and an ability to engage in, life-long learning.

a) Embedded assessment data was not collected from CSCI 490 in A.Y. 2006-2007 for this program outcome.

b) None of the MFT Assessment Indicators directly assess this program outcome.
Figure 9: Senior Exit Survey Trends for Program Outcome #9.

c) Figure 9 shows the score for College of ECC Senior Exit Survey Question #43 (Continue learning) as 4.04 (see Table 7 on page 34 in the Appendix) for this assessment cycle, down 1.22% from 4.09 last year. This score is above the department target of 3.80 and although Figure 9 illustrates a slight downward trend from last year, this is probably not an area of concern.

10. A knowledge of contemporary issues.

a) Embedded assessment data was not collected from CSCI 301 in A.Y. 2006-2007 for this program outcome.

b) None of the MFT Assessment Indicators directly assess this program outcome.

c) Figure 8 shows that College of ECC Senior Exit Survey Question #40 (Understand contemporary issues facing society) received a mean score of 3.57 (see Table 7 on page 34 in the Appendix) for this assessment cycle, down 7.03% from 3.84 last year. This is the sharpest decline observed for this score.

11. An ability to use the techniques, skills, and modern computing tools necessary for computing practice.

a) Embedded assessment data was not collected from CSCI 330 or CSCI 490 in A.Y. 2006-2007 for this program outcome.

b) Direct measurement data for Spring 2006 and Fall 2006 from MFT Assessment Indicator (AI) trends shown in Figure 12 on page 19 indicate a dramatic 42.85% improvement for AI 1 (Programming) from 45% (2.80) to 75% (4.00), a 5.88% improvement for AI 2 (Discrete Structures and Algorithms) from 60% (3.40) to 65% (3.60), and a 21.43% improvement for AI 3 (Systems: Architecture, Operating Systems,
Networking, Database) from 45% (2.80) to 60% (3.40). Unfortunately, between Fall 2006 and Spring 2007, there was no change for AI 1, a 15.38% improvement for AI 2 from 65% (3.60) to 75% (4.00), and a 16.67% decline for AI 3 from 60% (3.40) to 50% (3.00).

Figure 10: Senior Exit Survey Trends for Program Outcome #11.

c) Figure 10 shows College of ECC Senior Exit Survey Question #41 (Use modern tools and technology) received a mean score of 3.52 for this assessment cycle, down 9.29% from 3.88 last year (see Table 7 on page 34 in the Appendix). This is the sharpest observed decline for this score.
B. Proposed Actions from Outcomes

With only four of the eleven program outcomes receiving favorable scores and trends, there is major cause for concern regarding the department’s approach in achieving its Computer Science program outcomes. Perhaps the most glaring issues are the combined embedded assessment measures that relate to the fundamental abilities of problem solving, designing and implementing programs (including analysis and interpretation of code and data), and the ability to use the techniques, skills, and modern computing tools necessary for computing practice. Additionally, the data indicates an apparent lack knowledge of the impact of computing solutions in a global and societal context, and a lack of knowledge of contemporary issues. Combine these with the 16.67% decline for AI 3 (Systems: Architecture, Operating Systems, Networking, and Databases) of the MFT from 60% (3.40) to 50% (3.00) between Fall 2006 and Spring 2007 and the issue is magnified. The fact that there was no change for AI 1 (Programming) and there was a 15.38% improvement for AI 2 (Discrete Structures and Algorithms) from 65% (3.60) to 75% (4.00) for the same period, this seems to indicate that students are understanding the fundamental problem solving and programming techniques covered in the curriculum, but they may not fully comprehend how to take these to the next level.

Hence, there seems to be a need for faculty to instill the proficiency in applying techniques, resourcefulness, and use of modern computing tools necessary for computing practice across the curriculum in general and in upper division courses in particular. This outcome will be monitored closely in future assessments. The department is also currently reviewing potential adjustments and improvements to the Computer Science curriculum that are meant to address these issues.

Scores for an understanding of professional and ethical responsibilities, an ability to communicate effectively, and a recognition of the need for, and an ability to engage in, life-long learning, were all positive and the trends do not seem to indicate any concern.
C. Other Direct Measures

Every semester since Spring 2002, the department has been using the Educational Testing Services’ *Major Field Test (MFT) in Computer Science* as a direct measure to supplement embedded assessment measures from required courses in the program. The MFT is a standardized test that provides assessment information (based on national, comparative data) in the form of score statistics and assessment indicators.

MFT results from Spring 2007 was a mean score of 156.82, a median score of 156, the lowest was 133 and the highest was 197. Figure 11 illustrates the score statistics trend from 2002 through 2007.

![MFT in Computer Science @ Chico State](image)

*Figure 11: MFT in Computer Science, Scores Statistics Trend*

The MFT in Computer Science also provides assessment information categorized into the following Assessment Indicators (or AIs; numbers in parenthesis indicate approximate percentage of MFT questions in that category):

- **AI 1:** Programming (35%)
- **AI 2:** Discrete Structures and Algorithms (40%)
- **AI 3:** Systems: Architecture, Operating Systems, Networking, Database (25%)

MFT mean percent scores from Spring 2007 was 67 (75% or 4.00) for AI 1, 44 (75% or 4.00) for AI 2, and 46 (50% or 3.00) for AI 3 (numbers in parentheses indicate percent at or below the mean percent score, based on comparative data, and the equivalent value in a 5-point scale). Figure 12 illustrates the assessment indicators trend from 2002 through 2007.

---

4 Educational Testing Services Major Field Test (MFT), www.ets.org/hep/mft/
The department also uses the MFT Assessment Indicators to determine appropriate topical coverage in and student preparation from core courses. Table 2 provides the MFT AI and CSCI core courses matrix.

Table 2: MFT Assessment Indicators x Core Course Matrix.

<table>
<thead>
<tr>
<th>Assessment Indicator (AI)</th>
<th>Core Courses</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI-1: Programming Fundamentals</td>
<td>111 112 221 301 305 311 315 320 330 340 490 550</td>
</tr>
<tr>
<td>AI-2: Computer Organization, Architecture, and Operating Systems</td>
<td></td>
</tr>
<tr>
<td>AI-3: Algorithms, Theory, and Computational Mathematics</td>
<td></td>
</tr>
</tbody>
</table>

The correlation between the MFT AI’s and the department’s program outcomes can be derived from Table 2 and Table 3. Interestingly, all three AI’s cover the same outcomes: program outcome #1 (An ability to apply knowledge of math, science, and computing), #2 (An ability to design and implement programs as well as to analyze and interpret code and data), #3 (An ability to design a system, component, or process to meet desired needs), #5 (An ability to identify, formulate, and solve computing problems), and #11 (An ability to use the techniques, skills, and modern computing tools necessary for computing practice).
D. Indirect Measures

1. College of ECC Senior Exit Survey

College of ECC Senior Exit Surveys were administered in December 2006, and again in May 2007 to graduating seniors. Of the 254 surveys administered for the College, 23 were from CSCI majors, and 13 were CIS majors.

Table 7 on page 34 in the Appendix summarizes the results from questions in the ECC Senior Exit Survey that pertain to program outcomes.

Table 8 and on page 35 in the Appendix summarizes the results from questions in the ECC Senior Exit Survey that pertain to educational satisfaction.

Although the survey results that relate to teaching are below the department target of 3.80, responses to Question #15 (Quality of teaching by faculty in department) had a slight 6.99% improvement from 2.72 to 2.91, while responses to Question #16 (Quality of teaching by other faculty) and #17 (Access to faculty in your department) are on a downward trend. Hopefully, the trend for Question #15 continues to improve since last year’s score was the lowest received for this question.
Survey results seem to indicate that although student satisfaction with the quality of courses in the department improved 5.56% from 2.88 to 3.04, this was not true for the availability of courses which dropped 9.58% from 3.13 to 2.83. Scores are also below the 3.80 department target.

Overall, Computer Science students reported negative trends regarding the quality and access to laboratories, lab facilities, and equipment. The downward trend was 12.47% for Question #20 (from 4.17 to 3.65), 15.91% for Question #21 (from 3.52 to 2.96), 9.75% for Question #22 (from 4.00 to 3.61) and 15.02% for Question #23 (from 3.53 to 3.00).
2. Department of Computer Science Senior Exit Survey

The department administered its own senior exit survey in CSCI 490 last Spring 2006. The intent was to determine a difference in educational satisfaction between CSCI majors who took their first two core programming classes (CSCI 111 and CSCI 112) from Chico State and those who took them elsewhere. The statements students had to agree or disagree to in that survey are listed in the Appendix. Twelve CSCI majors took the survey: seven took CSCI 111 and CSCI 112 at Chico State, and five did not.

Figure 16 summarizes the results from the seven CSCI majors that took CSCI 111 and CSCI 112 at Chico State. Although the sample size was small, overall, CSCI majors who took CSCI 111 and CSCI 112 at Chico State seem to be satisfied with their education. Only four of the 20 statements in the survey received agreements below 70%: #4 (The computer science labs are adequate to support the program), #11 (I have learned to work effectively on teams), #17 (My instructors were available to help me with the coursework when I needed it), and #20 (I would recommend the Computer Science Department at CSU, Chico to my friends). These results somewhat coincide with the College of ECC Senior Exit Survey results.

Figure 17 summarizes the results from the five CSCI majors that did not take CSCI 111 and CSCI 112 at Chico State. Again, although the sample size was small, overall, CSCI majors who did not take CSCI 111 and CSCI 112 at Chico State seem to be less satisfied with their education compared to their counterparts who took the same two courses at Chico State. Eleven of the 20 statements in the survey received responses below 70%: #1 (My introductory courses provided a good background/supplement to my major), #2 (My introductory courses were well taught), #3 (The computer science classrooms are adequate to support the program), #4 (The computer science labs are adequate to support the program), #6 (I now have the ability to design and implement a system or component to meet specifications using modern computing tools), #8 (I have a solid understanding of the concepts and analytical approaches used in basic science), #13 (The Computer Science Department has given me the opportunity for specialized training within computer science through coursework and research), #16 (I have had access to good advising in the computer science program), #17 (My instructors were available to help me with the academic aspects of my classes), and #19 (The computer science program has prepared me to be a successful software professional).
coursework when I needed it), #19 (I feel that my peers have positive feelings about being computer scientists), and #20 (I would recommend the Computer Science Department at CSU, Chico to my friends).

CSCI Majors that did not take CSCI 111 and CSCI 112 at Chico State

![Bar chart showing responses to department exit survey from CSCI Majors that did not take CSCI 111 and CSCI 112 at Chico State.]

Figure 17: Educational satisfaction – responses to a department exit survey from CSCI Majors that did not take CSCI 111 and CSCI 112 at Chico State.

Of particular interest from the results given above are the differences in responses to Statements #1 (My introductory courses provided a good background/supplement to my major) and #2 (My introductory courses were well taught). CSCI majors that took their CSCI 111 and CSCI 112 at Chico State agreed to both these statements 100%, whereas CSCI majors that took these same courses elsewhere only agreed 20%, with twice that strongly disagreeing at 40%. It is speculated that this may partially explain the negative educational satisfaction by CSCI majors perceived from the College of ECC Senior Exit Survey.

3. Alumni Survey

Alumni surveys were not administered during this assessment cycle. The department is looking into the possibility of having Alumni Surveys available online.

4. Advisory Board Survey/Feedback

The department had an Industry Advisory Board (IAB) meeting last May 5, 2007. The department presented an overview of potential curriculum changes for the Computer Science degree program, and it was received favorably by the IAB.

5. ACM Pacific Northwest Regional Programming Contest

The department has been competing in the ACM Pacific Northwest Regional Programming Contest since the 1980s. This contest is affiliated with the ACM International Collegiate Programming Contest (ICPC), the world’s most prestigious computer programming competition. The top three winners from each regional competition advance to the ICPC World Finals for an all-out “battle of the brains.”
This year, the department sent three teams with three students per team (every team consisted of three CSCI majors, except for one team which had two CSCI majors and one CINS major). The Regionals consisted of 76 teams representing 35 schools. Chico State teams finished 23rd, 38th, and 72nd overall. These placements are converted to percentile values and the trends for the Top Team placement and the Average Team placement (average percentile over all teams) are depicted in Figure 18. The trend seems to indicate that the department’s top teams continue to progress in performance each year.

**E. Proposed Revision of Measures, Metrics, or Outcomes**

None. Major changes will be required as soon as the department approves and implements curriculum changes currently being worked out.
III. Assessment of Program Educational Objectives

A. Program Educational Objectives Summary

The objectives of the Computer Science program are that:
1. All students will be able to analyze and solve computing problems, or problems in related areas, and continually upgrade their knowledge and skills.
2. All students will be effective oral and written communicators and be able to function effectively as members of multi-disciplinary teams.
3. All students will have an appreciation for the individual, society, and human heritage and they will be aware of the impact of their work on society and the environment.
4. Those graduates who pursue careers as computing professionals will have the skills to use and design new and innovative systems that meet society’s needs.
5. Those graduates who pursue advanced degrees will have the skills to succeed in graduate programs in computing and related fields.

B. Alumni Survey Methodology and Results

Alumni surveys were not administered during this assessment cycle. The department is looking into the possibility of having Alumni Surveys available online.

C. Employer Survey Methodology and Results

The department currently does not administer employer surveys.

D. Advisory Board Feedback

The department had an Industry Advisory Board (IAB) meeting last May 5, 2007. Assessment of Program Educational Objectives was not discussed at that meeting.

E. Accreditation Evaluation Feedback


F. Program Improvement Resulting from Feedback

Not applicable.

G. Proposed Revisions to Program Objectives

None.
IV. Program Accreditation

A. Summary of Accreditation Feedback

1. Objectives and Assessments

   a) Accreditation Feedback. Concern: *(Standard I-5)* “The program demonstrated an ability to identify improvements based on assessment activities, but their formal process definition of their periodic assessment process is new. A stronger connection between the periodic assessment process and the identification of program activities needs to be established.”

   b) Corrective Actions. The department is using a faculty-approved regular assessment cycle designed to improve the connection between the periodic assessment process and the identification of program activities. This assessment cycle is outlined on pages 29 through 32 of this document. Additionally, the department is developing changes to the Computer Science curriculum based, in part, on assessment data. These changes and their rationale are being carefully documented so they can be presented at a future program assessment report.

   c) Results. Overall, the faculty seems supportive of the procedural adjustments made to the assessment activities of this department.

2. Student Support

   a) Accreditation Feedback. Concern: *(Standard II-5)* “Students feel that advising is not well coordinated and that some faculty members are not familiar enough with the curriculum requirements.”

   b) Corrective Actions. The department has assigned a Computer Science faculty member as Undergraduate Advising Coordinator for the program. The previous Undergraduate Advising Coordinator was also the program coordinator for a separate program offered by the department – that program is now offered directly by the college and is no longer offered by the department.

   c) Results. The concern remains and will continue to be monitored to the next assessment cycle.

3. Faculty

   a) Accreditation Feedback. Concern: *(Standard III-1)* “The enrollment trend should be monitored and any significant growth in student demand must be matched with adequate and appropriate faculty resources.”

---

5 The results of the program’s periodic assessments must be used to help identify opportunities for program improvement.
6 There must be established standards and procedures to ensure that graduates meet the requirements of the program.
7 There must be enough full-time faculty members with primary commitment to the program to provide continuity and stability.
b) Corrective Actions. The department is evaluating both its undergraduate and graduate Computer Science programs and is in the process of adopting some changes and improvements to both.

c) Results. The concern remains and will continue to be monitored to the next assessment cycle.

4. Curriculum

a) Accreditation Feedback. Concern: None.

b) Corrective Actions. Not applicable.

5. Laboratories and Computing Facilities

a) Accreditation Feedback. Concern: (Standard V-5)\(^8\) “Except during scheduled classes, instructional assistance in the laboratories is minimally available.”

b) Corrective Actions. Although instructional assistance in the laboratories was conceived to be minimal, majors are made aware of the availability of these tutoring services from two separate offices – one on each floor where laboratories under the jurisdiction of the department reside. Additionally, professors typically handle laboratories associated with courses they teach.

c) Results. The concern remains and will continue to be monitored to the next assessment cycle.

6. Institutional Support and Financial Resources

a) Accreditation Feedback. Concern: None.

b) Corrective Actions. Not applicable.

7. Institutional Facilities

a) Accreditation Feedback. Concern: None.

b) Corrective Actions. Not applicable.

B. Summary of Actions Taken to Address Issues

The department continues its efforts in addressing accreditation feedback/concerns and in monitoring results of any corrective actions that directly impact the stability, overall quality, or future accreditation of the program.

\(^8\) Instructional assistance must be provided for the laboratories and computing facilities.
V. Appendices

A. Program Outcomes Summary

All computer science graduates shall demonstrate
1. An ability to apply knowledge of math, science and computing.
2. An ability to design and implement programs as well as to analyze and interpret code and data.
3. An ability to design a system, component, or process to meet desired needs.
4. An ability to function on multi-disciplinary teams.
5. An ability to identify, formulate and solve computing problems.
6. An understanding of professional and ethical responsibilities.
7. An ability to communicate effectively.
8. The broad education necessary to understand the impact of computing solutions in a global and societal context.
9. A recognition of the need for, and an ability to engage in, life-long learning.
10. A knowledge of contemporary issues.
11. An ability to use the techniques, skills, and modern computing tools necessary for computing practice.
## B. Core Course Alignment / Program Outcomes Matrix

Table 3: Core Course x Program Outcomes Matrix.

<table>
<thead>
<tr>
<th>Core Courses</th>
<th>Program Outcomes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSCI 111 / EECE 135 Programming and Algorithms I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 112 Programming and Algorithms II</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 221 / EECE 221 Assembly Language Programming</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 301 Computer’s Impact on Society</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 305 / EECE 335 Project Requirements, Design, Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 311 Algorithms and Data Structures</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 315 Programming Languages</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 320 Computer Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 330 Software Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 340 Operating Systems Programming</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 490 Directed Programming Experience</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCI 550 Theory of Computing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Key:**
- Introduced
- Practiced
- Assessed

Page 29 of 60
C. Timeline

To facilitate collection of assessment data, the department identified three subsets for program outcome assessment. These subsets are listed in Table 4.

<table>
<thead>
<tr>
<th>Subset</th>
<th>Program Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>An ability to apply knowledge of math, science, and computing.</td>
</tr>
<tr>
<td>2</td>
<td>An ability to design and implement programs as well as to analyze and interpret code and data.</td>
</tr>
<tr>
<td>3</td>
<td>An ability to design a system, component, or process to meet desired needs.</td>
</tr>
<tr>
<td>4</td>
<td>An ability to function on multi-disciplinary teams.</td>
</tr>
<tr>
<td>5</td>
<td>An ability to identify, formulate, and solve computing problems.</td>
</tr>
<tr>
<td>6</td>
<td>An understanding of professional and ethical responsibilities.</td>
</tr>
<tr>
<td>7</td>
<td>An ability to communicate effectively.</td>
</tr>
<tr>
<td>8</td>
<td>The broad education necessary to understand the impact of computing solutions in a global and societal context.</td>
</tr>
<tr>
<td>9</td>
<td>A recognition of the need for, and an ability to engage in, life-long learning.</td>
</tr>
<tr>
<td>10</td>
<td>A knowledge of contemporary issues.</td>
</tr>
<tr>
<td>11</td>
<td>An ability to use the techniques, skills, and modern computing tools necessary for computing practice.</td>
</tr>
</tbody>
</table>
A rotational assessment timeline for data collection, assessment and evaluation is shown in Table 5. This timeline was developed to maximize the efficiency of the assessment data collection. The guiding principle in the department’s assessment data collection is that data does not need to be collected from every student/graduate every year on every outcome to provide valuable information for program improvement. The timeline given in Table 5 is reviewed annually to determine its appropriateness for the information that the assessment yields.

Table 5: Rotational Assessment Timeline.

<table>
<thead>
<tr>
<th></th>
<th>2006</th>
<th>2007</th>
<th>2008</th>
<th>2009</th>
<th>2010</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Spring</td>
<td>Fall</td>
<td>Spring</td>
<td>Fall</td>
<td>Spring</td>
<td>Fall</td>
</tr>
<tr>
<td>Subset 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subset 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subset 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The above rotational assessment timeline will be administered in conjunction with the College of ECC Academic Year Assessment Cycle given in Table 9 on page 36 of the Appendix.
D. Embedded Assessment Measures

The Department of Computer Science approved the list of embedded assessment measures and metrics given in Table 6 for use in core courses in the Computer Science curriculum where assessment data is regularly collected.

Table 6: Embedded Assessment Measures and Metrics.

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Course</th>
<th>Embedded Assessment Measures &amp; Metrics</th>
</tr>
</thead>
</table>
| 1       | CSCI 221 | Students must demonstrate their ability to:  
  • utilize different number systems, specifically binary, hexadecimal, and decimal  
  • apply the rules for arithmetic in each number system  
  • perform conversions between number systems and encoding/decoding for representation in the memory of a computer  
  • represent, manipulate, and convert numeric quantities of different types in the memory of a computer, specifically unsigned integers, signed integers, and floating point  
To pass, a student must receive a grade of at least 80% on a subset of questions from Midterm #1, which primarily covers the above topics.  
CSCI 320\(^9\): Students must demonstrate understanding of boolean algebra, truth tables, logic gates, combinational and sequential circuits, Karnaugh maps, binary and hexadecimal notation, unsigned and signed binary number representations, fundamental ALU design, and performance metrics and evaluation. Evaluation is based on results of at least one quiz and at least one exam. |
| 2       | CSCI 311 | Students must demonstrate understanding of discrete structures and algorithms, which are foundational material in computer science. Students must also demonstrate fluency in a programming language, along with the ability to connect fundamental programming concepts, basic data structures, and algorithmic processes. Evaluation is based on results of the final exam and the final programming assignment. |
| 3       | CSCI 320 CSCI 340 | CSCI 320: Students must demonstrate how the hardware components of a computer system memory hierarchy are organized and designed; these hardware components consist of a Translation Look-aside Buffer (TLB), a cache memory, main memory, and a disk. Students must demonstrate an understanding of the following design alternative trade-offs that affect the cost, performance, and organization of a memory hierarchy: Direct Mapped versus Set Associative versus Fully Associative; Write-through versus Write-Back; and the Block Size. Evaluation is based on the results of a quiz covering these concepts.  
CSCI 340: Students must demonstrate fundamental understanding of issues that influence the design of contemporary operating systems. Students must also exhibit essential knowledge of the use of operating systems (externals) and their design and implementation (internals). Evaluation is based on a project design grade of at least 70 out of 100 averaged over all projects. |
| 4       | CSCI 305 CSCI 330 | CSCI 305\(^9\): Students must actively participate in a series of team-based writing and presentation assignments. The evaluation of these team-based assignments includes the following aspects:  
  • Research – Collaborative research and formulations of project concepts and plans  
  • Presentation – A team-based presentation that describes the project concept and plan. Students are graded both on the overall presentation, and on their individual contributions.  
  • Documents – Three collaboratively generated documents: concept, requirements, and design  
  • Participation – The instructor’s evaluation of each individual student’s successful and constructive participation in the group activities.  
Students must receive a grade of at least 75% on team participation and team contribution evaluations.  
CSCI 330: Students must exhibit, through team involvement, fundamental understanding of the central elements of team building (organization, roles, responsibilities, decision-making, etc.) and team management (role identification and assignment, risk analysis, scheduling, team problem resolution). Evaluation is based on term group projects (documents and presentation grade). |

\(^{10}\) CSCI 320 used for data gathering for Program Outcome #1 in Spring 2006 only.  
\(^{11}\) Cross-listed with EECE 335, so information and assessment results are from the Department of EECE Program Assessment Report.
<table>
<thead>
<tr>
<th>Outcome</th>
<th>Course</th>
<th>Embedded Assessment Measures &amp; Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>CSCI 311, CSCI 340</td>
<td><strong>CSCI 311</strong>: Students must demonstrate understanding of problem-solving strategies and the role of algorithms in the problem-solving process. Students must exhibit knowledge and understanding of various methods for choosing and implementing the right data structure and algorithm combination. Evaluation is based on results of the final exam and the final programming assignment. <strong>CSCI 340</strong>: Students must demonstrate fundamental understanding of issues that influence the design of contemporary operating systems. Students must also exhibit essential knowledge of the use of operating systems (externals) and their design and implementation (internals). Evaluation is based on a project design grade of at least 70 out of 100 averaged over all projects.</td>
</tr>
<tr>
<td>6</td>
<td>CSCI 301</td>
<td>Students must recognize basic cultural, social, legal, and ethical issues inherent in the discipline of computing. Hence, students must demonstrate fundamental understanding of their individual roles in the direction of the evolving field of computing, as well as appreciate the philosophical questions, technical problems, and aesthetic values that play an important part in the development of the discipline. Evaluation is based on the students’ final grades, which combines performance measures from term papers, book reports, class participation, and the exams.</td>
</tr>
<tr>
<td>7</td>
<td>CSCI 330, CSCI 490</td>
<td><strong>CSCI 330</strong>: Students must demonstrate the ability to apply theory, knowledge, and practice for effectively and efficiently building software systems that satisfy the requirements of users and customers. Students must establish a level of professionalism, quality, timeliness, and cost-effectiveness as expected in producing a quality software system. Evaluation is based on term group projects (documents and presentation grade). <strong>CSCI 490</strong>: Since this is a capstone course, students must exhibit the ability to synthesize and apply concepts learned from core courses in the development of their senior project. Evaluation is based on students’ presentation of their work and their summary report.</td>
</tr>
<tr>
<td>8</td>
<td>CSCI 301</td>
<td>Students must recognize basic cultural, social, legal, and ethical issues inherent in the discipline of computing. Hence, students must demonstrate fundamental ability to ask serious questions about the social impact of computing and to evaluate proposed answers to those questions (e.g., the skill to anticipate the impact of introducing a given product into a given environment, based on potential impact upon individuals, groups, and institutions). Evaluation is based on the students’ final grades, which combines performance measures from term papers, book reports, class participation, and the exams.</td>
</tr>
<tr>
<td>9</td>
<td>CSCI 490</td>
<td><strong>CSCI 490</strong>: Since this is a capstone course, students must exhibit the ability to synthesize and apply concepts learned from core courses in the development of their senior project. Evaluation is based on students’ attendance at scheduled peer presentations.</td>
</tr>
<tr>
<td>10</td>
<td>CSCI 301</td>
<td>Students must demonstrate fundamental comprehension of basic cultural, social, legal, and ethical issues inherent in the discipline of computing. In particular, students need to understand issues pertaining to risks and liabilities of computer-based systems, intellectual property, privacy and civil liberties, computer crime, and others. Evaluation is based on the students’ final grades, which combines performance measures from term papers, book reports, class participation, and the exams.</td>
</tr>
<tr>
<td>11</td>
<td>CSCI 330, CSCI 490</td>
<td><strong>CSCI 330</strong>: Students must demonstrate the ability to apply theory, knowledge, and practice for effectively and efficiently building software systems that satisfy the requirements of users and customers. Students must establish a level of professionalism, quality, timeliness, and cost-effectiveness as expected in producing a quality software system. Evaluation is based on term group project (overall grade). <strong>CSCI 490</strong>: Since this is a capstone course, students must exhibit the ability to synthesize and apply concepts learned from core courses in the development of their senior project. Evaluation is based on their senior project grade.</td>
</tr>
</tbody>
</table>
E. Summary of Results from the College of ECC Senior Exit Survey

The following is from Table 5.4 on page 18 of the *College of ECC Assessment Report, Senior Exit Survey Results: Academic Year 2005-2006*.

**Table 7: Program Outcomes Trends for Computer Science.**

<table>
<thead>
<tr>
<th>Outcomes</th>
<th>Program Outcomes Questions</th>
<th>Mean Score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Spr 02</td>
</tr>
<tr>
<td>1</td>
<td>Q31. Apply knowledge to solve problems.</td>
<td>3.95</td>
</tr>
<tr>
<td>2</td>
<td>Q32. Design and conduct experiments.</td>
<td>3.84</td>
</tr>
<tr>
<td>2</td>
<td>Q33. Analyze and interpret experimental data.</td>
<td>3.81</td>
</tr>
<tr>
<td>3</td>
<td>Q34. Design component or system to meet needs.</td>
<td>4.00</td>
</tr>
<tr>
<td>4</td>
<td>Q35. Function on multidisciplinary team.</td>
<td>3.84</td>
</tr>
<tr>
<td>5</td>
<td>Q36. Identify, formulate, solve technical problems.</td>
<td>4.26</td>
</tr>
<tr>
<td>7</td>
<td>Q37. Communicate technical matters in writing.</td>
<td>3.84</td>
</tr>
<tr>
<td>7</td>
<td>Q38. Communicate technical matters orally.</td>
<td>3.95</td>
</tr>
<tr>
<td>6</td>
<td>Q39. Understand professional, ethical responsibilities.</td>
<td>4.05</td>
</tr>
<tr>
<td>8, 10</td>
<td>Q40. Understand contemporary issues facing society.</td>
<td>3.53</td>
</tr>
<tr>
<td>11</td>
<td>Q41. Use modern tools and technology.</td>
<td>4.11</td>
</tr>
<tr>
<td></td>
<td>Q42. Enter the workplace.</td>
<td>4.00</td>
</tr>
<tr>
<td>9</td>
<td>Q43. Continue learning.</td>
<td>4.21</td>
</tr>
</tbody>
</table>

In Table 7 above, scores that have stayed or improved from the previous year are in **green** and scores that have fallen lower than the previous year are bold and in **red**.
The following is from Table 4.4 on page 9 of the *College of ECC Assessment Report, Senior Exit Survey Results: Academic Year 2005-2006*.

Table 8: Educational Satisfaction Trends for Computer Science.

<table>
<thead>
<tr>
<th>Educational Satisfaction Questions</th>
<th>Mean Score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Spr 02 N=19</td>
</tr>
<tr>
<td></td>
<td>AY 03-04 N=25</td>
</tr>
<tr>
<td></td>
<td>AY 04-05 N=46</td>
</tr>
<tr>
<td></td>
<td>AY 05-06 N=31</td>
</tr>
<tr>
<td></td>
<td>AY 06-07 N=23</td>
</tr>
<tr>
<td>Q15. Quality of teaching of faculty in department</td>
<td>3.32</td>
</tr>
<tr>
<td></td>
<td>2.95</td>
</tr>
<tr>
<td></td>
<td>2.87</td>
</tr>
<tr>
<td></td>
<td>2.72</td>
</tr>
<tr>
<td></td>
<td>2.91</td>
</tr>
<tr>
<td>Q16. Quality of teaching by other faculty.</td>
<td>3.89</td>
</tr>
<tr>
<td></td>
<td>3.42</td>
</tr>
<tr>
<td></td>
<td>3.74</td>
</tr>
<tr>
<td></td>
<td>3.84</td>
</tr>
<tr>
<td></td>
<td>3.57</td>
</tr>
<tr>
<td>Q17. Access to faculty in your department.</td>
<td>3.79</td>
</tr>
<tr>
<td></td>
<td>4.04</td>
</tr>
<tr>
<td></td>
<td>3.68</td>
</tr>
<tr>
<td></td>
<td>3.72</td>
</tr>
<tr>
<td></td>
<td>3.91</td>
</tr>
<tr>
<td>Q18. Availability of courses in your department.</td>
<td>2.53</td>
</tr>
<tr>
<td></td>
<td>3.42</td>
</tr>
<tr>
<td></td>
<td>2.79</td>
</tr>
<tr>
<td></td>
<td>3.07</td>
</tr>
<tr>
<td></td>
<td>3.13</td>
</tr>
<tr>
<td>Q19. Quality of courses in your department.</td>
<td>3.42</td>
</tr>
<tr>
<td></td>
<td>3.52</td>
</tr>
<tr>
<td></td>
<td>2.79</td>
</tr>
<tr>
<td></td>
<td>3.07</td>
</tr>
<tr>
<td></td>
<td>2.88</td>
</tr>
<tr>
<td>Q20. Access to lab facilities and equipment.</td>
<td>3.11</td>
</tr>
<tr>
<td></td>
<td>3.92</td>
</tr>
<tr>
<td></td>
<td>3.22</td>
</tr>
<tr>
<td></td>
<td>3.78</td>
</tr>
<tr>
<td></td>
<td>4.17</td>
</tr>
<tr>
<td>Q21. Quality of laboratories and equipment.</td>
<td>3.21</td>
</tr>
<tr>
<td></td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>2.61</td>
</tr>
<tr>
<td></td>
<td>3.28</td>
</tr>
<tr>
<td></td>
<td>3.52</td>
</tr>
<tr>
<td>Q22. Access to computer facilities.</td>
<td>3.21</td>
</tr>
<tr>
<td></td>
<td>3.96</td>
</tr>
<tr>
<td></td>
<td>3.32</td>
</tr>
<tr>
<td></td>
<td>3.89</td>
</tr>
<tr>
<td></td>
<td>4.00</td>
</tr>
<tr>
<td>Q23. Quality of computer facilities.</td>
<td>3.42</td>
</tr>
<tr>
<td></td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>2.74</td>
</tr>
<tr>
<td></td>
<td>3.39</td>
</tr>
<tr>
<td></td>
<td>3.53</td>
</tr>
<tr>
<td>Q24. Academic advising from your major adviser.</td>
<td>4.05</td>
</tr>
<tr>
<td></td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>3.63</td>
</tr>
<tr>
<td></td>
<td>3.59</td>
</tr>
<tr>
<td></td>
<td>4.06</td>
</tr>
<tr>
<td>Q25. Academic advising from the Advising Office.</td>
<td>3.37</td>
</tr>
<tr>
<td></td>
<td>3.36</td>
</tr>
<tr>
<td></td>
<td>3.26</td>
</tr>
<tr>
<td></td>
<td>3.35</td>
</tr>
<tr>
<td></td>
<td>3.35</td>
</tr>
<tr>
<td>Q26. Career advice from faculty in your department.</td>
<td>3.32</td>
</tr>
<tr>
<td></td>
<td>3.38</td>
</tr>
<tr>
<td></td>
<td>3.37</td>
</tr>
<tr>
<td></td>
<td>3.20</td>
</tr>
<tr>
<td></td>
<td>3.29</td>
</tr>
<tr>
<td>Q27. Availability of GE courses.</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>3.79</td>
</tr>
<tr>
<td></td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>4.03</td>
</tr>
<tr>
<td>Q28. Quality of GE courses.</td>
<td>3.63</td>
</tr>
<tr>
<td></td>
<td>3.68</td>
</tr>
<tr>
<td></td>
<td>3.58</td>
</tr>
<tr>
<td></td>
<td>3.51</td>
</tr>
<tr>
<td></td>
<td>3.65</td>
</tr>
<tr>
<td>Q29. Overall quality of your education.</td>
<td>3.58</td>
</tr>
<tr>
<td></td>
<td>3.61</td>
</tr>
<tr>
<td></td>
<td>3.16</td>
</tr>
<tr>
<td></td>
<td>3.52</td>
</tr>
<tr>
<td></td>
<td>3.63</td>
</tr>
<tr>
<td>Q30. Overall experience at Chico State.</td>
<td>3.89</td>
</tr>
<tr>
<td></td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>3.32</td>
</tr>
<tr>
<td></td>
<td>3.67</td>
</tr>
<tr>
<td></td>
<td>3.75</td>
</tr>
<tr>
<td></td>
<td>3.13</td>
</tr>
</tbody>
</table>

In Table 8 above, scores that have stayed or improved from the previous year are in *green* and scores that have fallen lower than the previous year are bold and in *red*. 
Table 9: College of ECC Academic Year Assessment Cycle.

**Academic Year Assessment Cycle**

The following are grouped logically by semesters and breaks but have overlapping timelines.

<table>
<thead>
<tr>
<th>Fall Semester</th>
<th>Date</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Weeks Before Classes Begin</td>
<td>Faculty submit course syllabi to Dept. Chairs for assessment review</td>
<td></td>
</tr>
<tr>
<td>1 Week Before Classes Begin</td>
<td>Department Chairs approve syllabi, return to faculty with assessment deliverables identified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Faculty administer Senior Exit Surveys prior to end of semester</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Winter Break</th>
<th>Date</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall Semester Grading Due Date</td>
<td>Faculty submit assessment materials to Dept. Chair and/or department Assessment Committee.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dept. Chair and/or Assessment Committee reviews pertinent outcomes assessment results</td>
<td></td>
</tr>
<tr>
<td>1 Week Before Classes Begin</td>
<td>Dept. Chair and/or Assessment Committee present assessment results to faculty, set actions if needed.*</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Spring Semester</th>
<th>Date</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Weeks Before Classes</td>
<td>Faculty submit course syllabi to Dept. Chairs for assessment review</td>
<td></td>
</tr>
<tr>
<td>1 Week Before Class</td>
<td>Department Chairs approve syllabi, return to faculty with assessment deliverables identified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Faculty administer Senior Exit Surveys prior to end of semester</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Summer</th>
<th>Date</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring Semester Grading Due Date</td>
<td>Faculty submit assessment materials to Dept. Chair and/or department Assessment Committee.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dept. Chair and/or Assessment Committee reviews pertinent outcomes assessment results</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dept. Chair and/or Assessment Committee conducts Alumni and Employer surveys</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Senior Exit Survey Results Analyzed - Results to Chairs</td>
<td></td>
</tr>
<tr>
<td>1 Week Before Classes Begin</td>
<td>Dept. Chair and/or Assessment Committee presents assessment results to faculty, set actions as needed.*</td>
<td></td>
</tr>
<tr>
<td>October</td>
<td>Annual Assessment Report from Dept. Chairs to Dean’s Office summarizing assessment results, actions.*</td>
<td></td>
</tr>
</tbody>
</table>

* Department assessment results are to be shared, analyzed, and reflected with appropriate constituent groups including advisory boards or student groups at regularly scheduled meetings.
F. Summary of Results from the Department of Computer Science Senior Exit Survey

The department administered its own senior exit survey in CSCI 490 last Spring 2006. The intent was to determine a difference in educational satisfaction between CSCI majors who took their first two core programming classes (CSCI 111 and CSCI 112) from Chico State and those who took them elsewhere. The statements students had to agree/disagree to in that survey were:

1. My introductory courses provided a good background/supplement to my major.
2. My introductory courses were well taught.
3. The computer science classrooms are adequate to support the program.
4. The computer science labs are adequate to support the program.
5. The computer science curriculum has given me a solid understanding of basic computer science principles and practices.
6. I now have the ability to design and implement a system or component to meet specifications using modern computing tools.
7. I am capable of evaluating and comparing the efficiencies of alternate problem solutions.
8. I have a solid understanding of the concepts and analytical approaches used in basic science.
9. I have effective oral communication skills, with technical and non-technical audiences.
10. I have effective written communication skills, both within and outside the CS discipline.
11. I have learned to work effectively on teams.
12. I have confidence in my ability to work independently, and recognize the need for lifelong learning.
13. The Computer Science Department has given me the opportunity for specialized training within computer science through coursework and research.
14. I have gained a breadth of knowledge in non-technical areas and understand the impact of computer science on society.
15. I am aware of the ethical issues and societal concerns relating to information technology in society.
16. I have had access to good advising in the computer science program.
17. My instructors were available to help me with the coursework when I needed it.
18. I have positive feelings right now about being a computer scientist.
19. I feel that my peers have positive feelings about being computer scientists.
20. I would recommend the Computer Science Department at CSU, Chico to my friends.
The following table presents the results from CSCI majors that took CSCI 111 and CSCI 112 at Chico State:

Table 10: Agreement responses to the Department of Computer Science Senior Exit Survey, CSCI Majors that took CSCI 111 and CSCI 112 from Chico State.

<table>
<thead>
<tr>
<th>Agreement</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA A N D SD</td>
<td></td>
</tr>
<tr>
<td>2 5 0 0 0 0</td>
<td>1. My introductory courses provided a good background/supplement to my major.</td>
</tr>
<tr>
<td>3 4 0 0 0 0</td>
<td>2. My introductory courses were well taught.</td>
</tr>
<tr>
<td>3 3 1 0 0 0</td>
<td>3. The computer science classrooms are adequate to support the program.</td>
</tr>
<tr>
<td>3 1 2 1 0 0</td>
<td>4. The computer science labs are adequate to support the program.</td>
</tr>
<tr>
<td>1 6 0 0 0 0</td>
<td>5. The computer science curriculum has given me a solid understanding of basic computer science principles and practices.</td>
</tr>
<tr>
<td>1 6 0 0 0 0</td>
<td>6. I now have the ability to design and implement a system or component to meet specifications using modern computing tools.</td>
</tr>
<tr>
<td>1 4 2 0 0 0</td>
<td>7. I am capable of evaluating and comparing the efficiencies of alternate problem solutions.</td>
</tr>
<tr>
<td>1 6 0 0 0 0</td>
<td>8. I have a solid understanding of the concepts and analytical approaches used in basic science.</td>
</tr>
<tr>
<td>1 4 2 0 0 0</td>
<td>9. I have effective oral communication skills, with technical and non-technical audiences.</td>
</tr>
<tr>
<td>2 4 1 0 0 0</td>
<td>10. I have effective written communication skills, both within and outside the CS discipline.</td>
</tr>
<tr>
<td>2 2 2 1 0 0</td>
<td>11. I have learned to work effectively on teams.</td>
</tr>
<tr>
<td>4 3 0 0 0 0</td>
<td>12. I have confidence in my ability to work independently, and recognize the need for lifelong learning.</td>
</tr>
<tr>
<td>0 5 1 1 0 0</td>
<td>13. The Computer Science Department has given me the opportunity for specialized training within computer science through coursework and research.</td>
</tr>
<tr>
<td>1 4 2 0 0 0</td>
<td>14. I have gained a breadth of knowledge in non-technical areas and understand the impact of computer science on society.</td>
</tr>
<tr>
<td>1 5 0 0 1 0</td>
<td>15. I am aware of the ethical issues and societal concerns relating to information technology in society.</td>
</tr>
<tr>
<td>1 4 1 1 0 0</td>
<td>16. I have had access to good advising in the computer science program.</td>
</tr>
<tr>
<td>1 3 3 0 0 0</td>
<td>17. My instructors were available to help me with the coursework when I needed it.</td>
</tr>
<tr>
<td>1 6 0 0 0 0</td>
<td>18. I have positive feelings right now about being a computer scientist.</td>
</tr>
<tr>
<td>1 6 0 0 0 0</td>
<td>19. I feel that my peers have positive feelings about being computer scientists.</td>
</tr>
<tr>
<td>0 3 2 2 0 0</td>
<td>20. I would recommend the Computer Science Department at CSU, Chico to my friends.</td>
</tr>
</tbody>
</table>
The following table presents the results from CSCI majors that did not take CSCI 111 and CSCI 112 at Chico State:

**Table 11: Agreement responses to the Department of Computer Science Senior Exit Survey, CSCI Majors that did not take CSCI 111 and CSCI 112 from Chico State.**

<table>
<thead>
<tr>
<th>Agreement</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA A N D SD</td>
<td></td>
</tr>
<tr>
<td>0 1 2 0 2</td>
<td>1. My introductory courses provided a good background/supplement to my major.</td>
</tr>
<tr>
<td>0 1 2 0 2</td>
<td>2. My introductory courses were well taught.</td>
</tr>
<tr>
<td>0 2 3 0 0</td>
<td>3. The computer science classrooms are adequate to support the program.</td>
</tr>
<tr>
<td>1 1 3 0 0</td>
<td>4. The computer science labs are adequate to support the program.</td>
</tr>
<tr>
<td>2 3 0 0 0</td>
<td>5. The computer science curriculum has given me a solid understanding of basic computer science principles and practices.</td>
</tr>
<tr>
<td>1 2 2 0 0</td>
<td>6. I now have the ability to design and implement a system or component to meet specifications using modern computing tools.</td>
</tr>
<tr>
<td>1 3 1 0 0</td>
<td>7. I am capable of evaluating and comparing the efficiencies of alternate problem solutions.</td>
</tr>
<tr>
<td>2 1 2 0 0</td>
<td>8. I have a solid understanding of the concepts and analytical approaches used in basic science.</td>
</tr>
<tr>
<td>2 3 0 0 0</td>
<td>9. I have effective oral communication skills, with technical and non-technical audiences.</td>
</tr>
<tr>
<td>0 4 1 0 0</td>
<td>10. I have effective written communication skills, both within and outside the CS discipline.</td>
</tr>
<tr>
<td>0 5 0 0 0</td>
<td>11. I have learned to work effectively on teams.</td>
</tr>
<tr>
<td>3 2 0 0 0</td>
<td>12. I have confidence in my ability to work independently, and recognize the need for lifelong learning.</td>
</tr>
<tr>
<td>1 0 3 1 0</td>
<td>13. The Computer Science Department has given me the opportunity for specialized training within computer science through coursework and research.</td>
</tr>
<tr>
<td>1 3 1 0 0</td>
<td>14. I have gained a breadth of knowledge in non-technical areas and understand the impact of computer science on society.</td>
</tr>
<tr>
<td>1 3 1 0 0</td>
<td>15. I am aware of the ethical issues and societal concerns relating to information technology in society.</td>
</tr>
<tr>
<td>1 2 2 0 0</td>
<td>16. I have had access to good advising in the computer science program.</td>
</tr>
<tr>
<td>0 3 2 0 0</td>
<td>17. My instructors were available to help me with the coursework when I needed it.</td>
</tr>
<tr>
<td>1 3 1 0 0</td>
<td>18. I have positive feelings right now about being a computer scientist.</td>
</tr>
<tr>
<td>0 3 2 0 0</td>
<td>19. I feel that my peers have positive feelings about being computer scientists.</td>
</tr>
<tr>
<td>0 1 2 1 0</td>
<td>20. I would recommend the Computer Science Department at CSU, Chico to my friends.</td>
</tr>
</tbody>
</table>
### G. Measurement Standards (Rubrics)

**Table 12: CSCI Program Outcomes x ABET Scoring Rubric Matrix**

<table>
<thead>
<tr>
<th>CSCI Program Outcome X</th>
<th>Level 5 Performance characterized by ...</th>
<th>Level 3 Performance characterized by ...</th>
<th>Level 1 Performance characterized by ...</th>
</tr>
</thead>
</table>
| **CSCI Program Outcome 1:** An ability to apply knowledge of math, science, and computing | • Combines mathematical and/or scientific principles to formulate algorithms  
• Applies the appropriate mathematical concepts to solve computing problems  
• Shows appropriate interpretation of mathematical and scientific terms  
• Translates academic theory into computing applications and accepts limitations of mathematical models of physical reality  
• Performs/executes calculations, conversions, transformations correctly  
  – by hand  
  – in software implementations  
• Correctly analyzes data sets using statistical concepts | • Chooses a mathematical model or scientific principle that applies to a problem, but has trouble in algorithm development  
• Shows nearly complete understanding of applications of mathematical concepts in problem-solving  
• Most mathematical and scientific terms are interpreted correctly  
• Some gaps in understanding the application of theory to the problem and expects theory to predict reality  
• Minor errors in calculations, conversions, transformations  
  – by hand  
  – in software implementations  
• Minor errors in statistical analysis of data | • Does not understand the connection between mathematical models and/or scientific principles and algorithm development  
• Does not understand the application of mathematical concepts in solving computing problems  
• Mathematical and scientific terms are interpreted incorrectly or not at all  
• Does not appear to grasp the connection between theory and the problem  
• Calculations, conversions, transformations not performed or performed incorrectly  
  – by hand  
  – in software implementations  
• No application of statistics to analysis of data |
| **CSCI Program Outcome 2:** An ability to design and implement programs as well as to analyze and interpret code and data | • Demonstrates creative synthesis of solution and creates new alternatives by combining knowledge and information  
• Can relate theoretical concepts to practical problem solving  
• Can predict and defend problem outcomes  
• Uses appropriate resources to locate information needed to solve problems  
• Takes new information and effectively integrates it with previous knowledge  
• Demonstrates understanding of how various pieces of the problem relate to each other and the whole | • Demonstrates solution with integration of diverse concepts or derivation of useful relationships involving ideas covered in course concepts; however, no alternative solutions are generated  
• Connects theoretical concepts to practical problem-solving when prompted  
• Occasionally predicts and defends problem outcomes  
• Uses limited resources to solve problems  
• Must be assisted in integrating previous knowledge and new information  
• Is missing some of the pieces of the problem | • Demonstrates solutions implementing simple applications of one formula or equation with close analogies to class/lecture problems  
• Does not see the connection between theory and practical problem solving  
• Is unable to predict or defend problem outcomes  
• Uses no resources to solve problems  
• Has no concept of how previous knowledge and new information relate  
• Does not realize when major components of the problem are missing |
<table>
<thead>
<tr>
<th>CSCI Program Outcome X</th>
<th>Level 5 Performance characterized by ...</th>
<th>Level 3 Performance characterized by ...</th>
<th>Level 1 Performance characterized by ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABET Scoring Rubric</td>
<td>• Formulates strategies for solving problems</td>
<td>• Has some strategies for problem-solving, but does not apply them consistently</td>
<td>• Has no coherent strategies for problem solving</td>
</tr>
<tr>
<td></td>
<td>• Answers/Solutions are correct and properly labeled/commented</td>
<td>• Answers/Solutions are nearly correct, but properly labeled/commented (within reasonable and logical range of the correct answer-it’s in the &quot;ballpark&quot;)</td>
<td>• Answers/Solutions are incorrect and not checked for its reasonableness</td>
</tr>
<tr>
<td></td>
<td>• Answers/Solutions are correct and checked in other ways when they can be; interpretations are appropriate and make sense</td>
<td>• Answers/Solutions are correct, but not checked in other ways</td>
<td>• No attempt at checking obviously incorrect answers/solutions -- no commentary provided</td>
</tr>
<tr>
<td></td>
<td>• Can select appropriate programming language/environment and corresponding implementation tools to develop and maintain software</td>
<td>• Needs some guidance in selecting appropriate programming language/environment and corresponding implementation tools to develop and maintain software</td>
<td>• Cannot select the appropriate programming language/environment and corresponding implementation tools to develop and maintain software</td>
</tr>
<tr>
<td></td>
<td>• Is able to master use of the latest programming tools and integrated development environments</td>
<td>• Is tentative in the mastery of the latest programming tools and integrated development environments</td>
<td>• Lacks the desire/motivation to master use of the latest programming tools and integrated development environments, and does so incorrectly or requires frequent supervision</td>
</tr>
<tr>
<td></td>
<td>• Seeks information for projects/studies from multiple sources</td>
<td>• Seeks information for projects/studies from a few sources - mainly from the textbook or the instructor</td>
<td>• Seeks no extra information for projects/studies other than what is provided by instructor</td>
</tr>
<tr>
<td></td>
<td>• Develops sound design and implementation strategies, including a plan of attack, decomposition of solution into subtasks, and development of a timetable</td>
<td>• Can develop and compare multiple solutions to a problem, but does not usually arrive at the best result; conducts optimization but neglects one or two key aspects</td>
<td>• No design strategy; haphazard approach</td>
</tr>
<tr>
<td></td>
<td>• Capable of suggesting new approaches and improves on what has been done before</td>
<td>• Can use prior knowledge to design code segments competently when guided to do so</td>
<td>• Cannot design algorithms or individual code segments without significant amounts of help</td>
</tr>
<tr>
<td></td>
<td>• Develops several potential solutions and finds optimum</td>
<td>• Does not think holistically: does not see the integration of the pieces clearly</td>
<td>• Only focuses on one solution to a problem; no optimization attempted</td>
</tr>
<tr>
<td></td>
<td>• Understands how areas interrelate and demonstrates ability to integrate prior knowledge into a new problem</td>
<td></td>
<td>• Unable to relate prior knowledge to the design problem</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Has no concept of the process as a sum of its parts</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• No use of software tools and computing resources</td>
</tr>
</tbody>
</table>
|                        | | | • Design is done incompletely without the
<table>
<thead>
<tr>
<th>CSCI Program Outcome X ABET Scoring Rubric</th>
<th>Level 5 Performance characterized by ...</th>
<th>Level 3 Performance characterized by ...</th>
<th>Level 1 Performance characterized by ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Thinks holistically: sees the whole as well as the parts</td>
<td>• Minimal or incorrect use of software tools and computing resources</td>
<td>proper algorithms and without documentation or references</td>
<td>• Is absent from group/team meetings or work sessions &gt;50% of the time</td>
</tr>
<tr>
<td>• Uses software tools and computing resources effectively</td>
<td>• Completes design, but algorithms and implementations are not documented or referenced</td>
<td>• No consideration execution time, resource requirement, and other realistic constraints</td>
<td>• Does not contribute to group work at all or submits own work as the group’s</td>
</tr>
<tr>
<td>• Supports design procedure with documentation and references</td>
<td>• Includes only minor or cursory consideration of execution time, resource requirement, and other realistic constraints</td>
<td>• No application of computing and/or scientific principles</td>
<td>• Routinely fails to prepare for meetings</td>
</tr>
<tr>
<td>• Develops a solution that includes execution time, resource requirement, and other realistic constraints</td>
<td>• Applies computing and/or scientific principles incompletely or incorrectly to design a practical algorithm</td>
<td>• Software design is incomplete, no answer is given</td>
<td>• Does work on his/her own; does not value team work</td>
</tr>
<tr>
<td>• Applies computing and/or scientific principles correctly to design practical algorithms</td>
<td>• Gives an answer or produces output, but does not check its practicality</td>
<td>• Claims work of group as own or frequently blames others</td>
<td>• Claims work of group as own or frequently blames others</td>
</tr>
<tr>
<td>• Recognizes practical significance of problem solutions and/or program outputs</td>
<td>• Demonstrates the ability to assume a designated role in the group</td>
<td>• Hides in the background; only participates if strongly encouraged</td>
<td>• Is openly critical of the performance of others</td>
</tr>
</tbody>
</table>

CSCI Program Outcome 4: An ability to function on multi-disciplinary teams

| • Routinely present at group/team meetings or work sessions | • Absent occasionally, but does not inconvenience group/team | • Is absent from group/team meetings or work sessions >50% of the time |
| • Contributes a fair share to the project workload | • Sometimes depends on others to complete the work; contributes less than fair share | • Does not contribute to group work at all or submits own work as the group’s |
| • Is prepared for group/team meetings with clearly formulated ideas | • Prepares somewhat for group meetings, but ideas are not clearly formulated | • Routinely fails to prepare for meetings |
| • Cooperates with others (outside of the discipline) | • Occasionally works as a loner or interacts to a minor extent with extra-disciplinary team members | • Does work on his/her own; does not value team work |
| • Shares credit for success with others and accountability for team results | • Makes subtle references to other’s poor performance or sometimes does not identify contributions of other team members | • Claims work of group as own or frequently blames others |
| • Shares information with others and provides assistance to others | • Sometimes keeps information to himself/herself; not very willing to share | • Hides in the background; only participates if strongly encouraged |
| • Demonstrates the ability to assume a designated role in the group | • Takes charge when not in the position to lead | • Does not willingly assume team roles |
| • Values alternative perspectives and encourages participation among all team members | • Persuades others to adopt only his/her ideas or grudgingly accepts the ideas of others | • Does not consider the ideas of others |
| • Remains non-judgmental when disagreeing with others; seeks conflict resolution; does not "point fingers" or | | • Is openly critical of the performance of others |
| • Persuades others to adopt only his/her ideas or grudgingly accepts the ideas of others | | • Is discourteous to other group members |
| | | • Has no knowledge of disciplines outside of computer science |
### CSCI Program Outcome X:
#### ABET Scoring Rubric

<table>
<thead>
<tr>
<th>Level 5</th>
<th>Level 3</th>
<th>Level 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance characterized by ...</td>
<td>Performance characterized by ...</td>
<td>Performance characterized by ...</td>
</tr>
</tbody>
</table>

- blame others when things go wrong
- Is a courteous and collegial group member
- Has knowledge of technical skills, issues and approaches germane to disciplines outside of computer science

- Sometimes criticizes ideas of other team members or blames others for errors
- Is not always considerate or courteous towards team members
- Has some knowledge of other disciplines, but gets lost in discussions with extra-disciplinary team members

- Sometimes criticizes ideas of other team members or blames others for errors
- Is not always considerate or courteous towards team members
- Has some knowledge of other disciplines, but gets lost in discussions with extra-disciplinary team members

- Sometimes criticizes ideas of other team members or blames others for errors
- Is not always considerate or courteous towards team members
- Has some knowledge of other disciplines, but gets lost in discussions with extra-disciplinary team members

### CSCI Program Outcome 5:
An ability to identify, formulate and solve computing problems

- Fully capable of conceiving or contriving detailed algorithmic specifications into operational software products that can be justified in terms of (time and space) efficiency, flexibility, use and other factors.
- Fully capable of integrating computing, science, and mathematical principles to resolve all the constraints involved in the design process.
- Complete ability to produce a reasonable number of software design and/or algorithmic alternatives knowing the pros and cons and advantages and disadvantages of each alternative. Completely confident about defending the various alternative designs.

- Has ideas about transforming algorithmic specifications into operational software products including the justification of each alternative for efficiency, flexibility, use and other factors.
- Aware of how computing, scientific, and mathematical principles should be used in developing alternative designs.
- Aware of the advantages and disadvantages of each alternative design and/or algorithm and may be able to defend these various alternatives.

- Unable to use existing algorithmic specifications to come up with a viable software design.
- Unaware of how computing, scientific, and mathematical techniques are used for developing alternative designs.
- Does not understand the importance of having alternative design schemes for a single project.
- Is unable to defend the software design team’s algorithmic alternatives.

### CSCI Program Outcome 6:
An understanding of professional and ethical responsibilities

- Student understands and abides by the ACM Code of Ethics and the CSUC Students’ Code of Conduct
- Participates in class discussions and exercises on ethics and professionalism
- Demonstrates ethical behavior among peers and faculty
- Takes personal responsibility for his/her actions
- Is punctual, professional, and collegial;

- Student is aware of the existence of the ACM Code of Ethics and other bases for ethical behavior
- Does not take the discussion of ethics seriously but is willing to accept its existence
- Does not model ethical behavior among peers and faculty
- Doesn’t recognize the need to take personal responsibility for his/her actions

- Student is not aware of any codes for ethical behavior
- Does not participate in or contribute to discussions of ethics; does not accept the need for professional ethics
- Student has been caught cheating or plagiarizing the work of others
- Blames others for own issues and problems
- Is frequently absent from class and is
## ABET Scoring Rubric

### CSCI Program Outcome X

**ABET Scoring Rubric**

<table>
<thead>
<tr>
<th><strong>Level 5</strong></th>
<th><strong>Level 3</strong></th>
<th><strong>Level 1</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance characterized by...</td>
<td>Performance characterized by...</td>
<td>Performance characterized by...</td>
</tr>
</tbody>
</table>
| attends classes regularly  
• Evaluates and judges a situation in practice or as a case study, using facts and a professional code of ethics  
• Uses personal value system to support actions, but understands the role of professional ethical standards for corporate decisions | • Sometimes exhibits unprofessional behavior; is sometimes absent from class without reason  
• Evaluates and judges a situation in practice or as a case study using personal understanding of the situation, possibly applying a personal value system  
• Uses personal value system to support actions, but confuses personal ethics with professional ethics | generally not collegial to fellow students, staff, and faculty  
• Evaluates and judges a situation in practice or as a case study using a biased perspective without objectivity  
• Uses personal value system to support actions to the exclusion of all other ethical standards |

### CSCI Program Outcome 7a:

**An ability to communicate effectively (written)**

<table>
<thead>
<tr>
<th>Level 5</th>
<th>Level 3</th>
<th>Level 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance characterized by...</td>
<td>Performance characterized by...</td>
<td>Performance characterized by...</td>
</tr>
</tbody>
</table>
| Articulates ideas clearly and concisely  
• Organizes written materials in a logical sequence to enhance the reader’s comprehension (paragraphs, subheading, etc.)  
• Uses graphs, tables, and diagrams to support points and to explain, interpret, and assess information  
• Written work is presented neatly and professionally  
• Grammar and spelling are correct  
• Figures and/or tables are all in proper format  
• Uses good professional writing style  
• Conforms to the prescribed style guide or format (if any) | Articulates ideas, but writing is somewhat disjointed, superfluous or difficult to follow  
• Material are generally organized well, but paragraphs combine multiple thoughts or sections and sub-sections are not identified clearly  
• Uses graphs, tables, and diagrams, but only in a few instances are they applied to support, explain or interpret information  
• Work is not neatly presented throughout  
• One or two spelling/grammar errors per page  
• Figures and/or table are present but are flawed – axes mislabeled, no data points, etc.  
• Style is informal or inappropriate, jargon is used, improper voice, tense…  
• The prescribed style guide or format is only followed in some portions of the paper/write-up | Text rambles, points made are only understood with repeated reading, and key points are not organized  
• Little or no structure or organization; no subheadings or proper paragraph structure used  
• Graphs, tables or diagrams are used, but no reference is made to them  
• Work is not presented neatly  
• Spelling/grammar errors present throughout more than 1/3 of the paper  
• No figures or graphics are used at all  
• The writing style is inappropriate for the audience and for the assignment  
• The prescribed style guide or format is not followed |

### CSCI Program Outcome 7b:

**An ability to communicate**

<table>
<thead>
<tr>
<th>Level 5</th>
<th>Level 3</th>
<th>Level 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance characterized by...</td>
<td>Performance characterized by...</td>
<td>Performance characterized by...</td>
</tr>
</tbody>
</table>
| Plans and delivers an oral presentation effectively; well organized  
• Presentation has enough detail appropriate | Presents key elements of an oral presentation adequately  
• Presentation contains excessive or | Talk is poorly organized, e.g. no clear introduction or summary of talk is presented |
<table>
<thead>
<tr>
<th>CSCI Program Outcome X</th>
<th>Level 5 Performance characterized by ...</th>
<th>Level 3 Performance characterized by ...</th>
<th>Level 1 Performance characterized by ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>effectively (oral)</td>
<td>and technical content for the time constraint and the audience&lt;br&gt;- Presents well mechanically&lt;br&gt;  - Makes eye contact&lt;br&gt;  - Can be easily heard&lt;br&gt;  - Speaks comfortably with minimal prompts (notecards)&lt;br&gt;  - Does not block screen&lt;br&gt;  - No distracting nervous habits&lt;br&gt;- Uses proper American English&lt;br&gt;- Uses visual aides effectively&lt;br&gt;- Professional appearance&lt;br&gt;- Listens carefully and responds to questions appropriately; is able to explain and interpret results for various audiences and purposes</td>
<td>insufficient detail for time allowed or level of audience&lt;br&gt;- Has some minor difficulties with the mechanical aspects of the presentation&lt;br&gt;  - Eye contact is sporadic&lt;br&gt;  - Occasionally difficult to hear or understand speaking&lt;br&gt;  - Overuses prompts or does not use prompts enough-occasionally stumbles or loses place; appears to have memorized presentation&lt;br&gt;  - Occasionally blocks screen&lt;br&gt;  - Some nervous habits (um, ah, clicking pointer, etc.)&lt;br&gt;- Occasionally uses an inappropriate style of English – too conversational&lt;br&gt;- Visual aides have minor errors or are not always clearly visible&lt;br&gt;- Appearance is too casual for the circumstances&lt;br&gt;- Sometimes misunderstands questions, does not respond appropriately to the audience, or has some trouble answering questions</td>
<td>Presentation is inappropriately short or excessively long; omits key results during presentation&lt;br&gt;- Major difficulties with the mechanical aspects of the presentation&lt;br&gt;  - No eye contact&lt;br&gt;  - Difficult to hear or understand speaking&lt;br&gt;  - Reads from prepared script&lt;br&gt;  - Blocks the screen&lt;br&gt;  - Distracting nervous habits (um, ah, clicking pointer, etc.)&lt;br&gt;- Uses poor English&lt;br&gt;- Multiple slides are unclear or incomprehensible&lt;br&gt;- Does not listen carefully to questions, does not provide an appropriate answer, or is unable to answer questions about presentation material</td>
</tr>
</tbody>
</table>

<p>| CSCI Program Outcome 8: The broad education necessary to understand the impact of computing solutions in a global and societal context | Is familiar with the current trends in the computing discipline&lt;br&gt;- Respects the historical aspects of “classic” (standard) algorithms and their impacts&lt;br&gt;- Reads and is familiar with the content of periodicals that are relevant to understanding the global and societal impact of computing&lt;br&gt;- Has a personal perspective on the impact and significance of computing in today’s | Is aware of current events in society&lt;br&gt;- Is aware of historical aspects of “classic” (standard) algorithms, but is not influenced by them&lt;br&gt;- Is aware of the existence of technical periodicals – would know where to look to find them&lt;br&gt;- Is interested in computing because of what the discipline offers him/her personally | Is unaware of current events&lt;br&gt;- Is unaware of historical aspects of “classic” (standard) algorithms&lt;br&gt;- Is not familiar with any technical periodicals&lt;br&gt;- Is not sure why he/she is studying computing |</p>
<table>
<thead>
<tr>
<th>CSCI Program Outcome X</th>
<th>Level 5 Performance characterized by ...</th>
<th>Level 3 Performance characterized by ...</th>
<th>Level 1 Performance characterized by ...</th>
</tr>
</thead>
</table>
| **CSCI Program Outcome 9:** A recognition of the need for, and an ability to engage in, life-long learning | • Demonstrates ability to learn independently  
• Goes beyond what is required in completing an assignment and brings information from outside sources into assignments  
• Learns from mistakes and practices continuous improvement  
• Demonstrates capability to think for one’s self  
• Demonstrates responsibility for creating one’s own learning opportunities  
• Is able to understand, interpret, and apply learned materials and concepts in a format different from that taught in class (e.g. different nomenclature, understand equation from different textbook)  
• Participates and takes a leadership role in professional and technical societies available to the student body | • Requires guidance as to expected outcome of task or project  
• Completes only what is required  
• Sometimes is able to avoid repeating the same mistakes  
• Does not always take responsibility for own learning  
• Seldom brings information from outside sources to assignments  
• Has some trouble using materials and concepts that are in a different format from that taught in class  
• Occasionally participates in the activities of local professional and technical societies | • Requires detailed or step-by-step instructions to complete a task  
• Has trouble completing even the minimum required tasks  
• Is unable to recognize own shortcomings or deficiencies  
• Assumes that all learning takes place within the confines of the class  
• Shows little or no interest in outside learning resources  
• Cannot use materials outside of what is explained in class  
• Does not show any interest in professional and/or technical societies |

| **CSCI Program Outcome 10:** A knowledge of contemporary issues | • Has knowledge of current events in the computing discipline and in society  
• Has a good perspective on the current job market  
• Able to discuss in-depth major political issues at national, state and local levels  
  – Can summarize essence of several issues; take and defend a position on them  
  – Is able to evaluate political solutions, or scenarios using a series of different measures - e.g., economic, quality of life; number | • Has some knowledge of current events  
• Has a somewhat narrow perspective on the current job market  
• Able to comment on major political issues, but is not familiar enough with them to defend a position on them  
  – Can summarize the facts of the issues  
  – Can only comment on possible alternative political solutions, or scenarios using a few different measures - e.g., economic, quality of life; number of individuals | • Has no clue about issues and events in the world  
• Hopes that a job will fall into his/her lab  
• Unable to comment on political solutions or is unaware of world and local happenings |
CSCI Program Outcome X
ABET Scoring Rubric

<table>
<thead>
<tr>
<th>Level 5 Performance characterized by ...</th>
<th>Level 3 Performance characterized by ...</th>
<th>Level 1 Performance characterized by ...</th>
</tr>
</thead>
</table>
| of individuals affected; political ramifications; etc. | affected; political ramifications; etc. | *Does not use computer-based and other resources in assignments/projects*
|  |  | *Does not seek additional information on problems from any resource*
|  |  | *Is unable to interpret and understand information from any resource*
|  |  | *Does not maintain currency in computing*
| *Uses computer-based and other resources effectively in assignments/projects* | *Seeks information on problems from multiple resources* | *Is unable to learn and implement process simulation software*
| *Seeks information on problems from multiple resources* | *Is able to interpret and understand information from a selection of resources* | *Does not understand the organization and use of the library*
| *Is able to interpret and understand information from a variety of resources* | *Maintains adequate abilities in computing* | *
| *Maintains current, state-of-the-art abilities in computing* | *Is able to somewhat learn and implement process simulation software* | *
| *Is able to learn and implement process simulation software* | *Somewhat understands the organization and use of the library* | *
| *Understands the organization and use of the library* |  | *

CSCI Program Outcome 11:
An ability to use the techniques, skills, and modern computing tools necessary for computing practice fields.

- Uses computer-based and other resources effectively in assignments/projects
- Seeks information on problems from multiple resources
- Is able to interpret and understand information from a variety of resources
- Maintains current, state-of-the-art abilities in computing
- Is able to learn and implement process simulation software
- Understands the organization and use of the library

- Uses computer-based and other resources adequately in assignments/projects
- Seeks information on problems from some resources
- Is able to interpret and understand information from a selection of resources
- Maintains adequate abilities in computing
- Is able to somewhat learn and implement process simulation software
- Somewhat understands the organization and use of the library

- Does not use computer-based and other resources in assignments/projects
- Does not seek additional information on problems from any resource
- Is unable to interpret and understand information from any resource
- Does not maintain currency in computing
- Is unable to learn and implement process simulation software
- Does not understand the organization and use of the library
H. Survey Instruments

1. Educational Testing Services Major Field Test (MFT) in Computer Science

MAJOR FIELD TESTS

Colleges and universities use the Major Field Tests to measure student academic achievement and growth and to assess the educational outcomes of their major programs. In addition, academic departments use the Major Field Tests to evaluate their curricula and to measure the progress of their students. The tests also provide students with an assessment of their own level of achievement within a field of study compared to that of students in their program and to national comparative data.

Test Length

The tests are two-hour multiple-choice tests. The addition of optional, locally developed questions may require a longer testing period.

Test Administration

Departments or schools choose when and where to give the tests; however, the tests are normally administered during the senior year when students have completed the majority of courses in the major. Many institutions administer the tests as part of the requirements of a capstone course.

National Comparative Data

A Comparative Data Guide, published each year, contains tables of scale scores and percentiles for individual student scores, departmental mean scores, and any subscores or group assessment indicators that the tests may support. The tables of data are drawn from senior-level test takers at a large number of diverse institutions. More than 500 colleges and universities employ one or more of the Major Field Tests for student achievement and curriculum evaluation each year.

Scores

Major Field Test score reports are sent directly to the office within an institution that purchases them, such as a department chairperson, dean, or director of testing. Results of the tests are reported for the entire group of test takers, as well as for individual students. Overall student scores are reported on a scale of 120-200; subscores (which many of the tests include) are reported on a scale of 20-100. Another score reported for most of the tests is based on group-level achievement in subfields of the discipline. These "assessment indicators" report the average percent of a subset of test questions answered correctly by all students tested. On Major Field Tests, only correct answers are scored, so students are not penalized for omissions or guesses.

COMPUTER SCIENCE
COMPUTER SCIENCE (3XMF)

(Current form introduced in Spring 2002)

The Major Field Test in Computer Science consists of 60 multiple choice questions, some of which are grouped in sets and based on such materials as diagrams, graphs, and program fragments. The test is divided into four subareas with content distribution as follows:

I. Programming Fundamentals (26 percent)
   A. Fundamental programming constructs and data structures (8 percent)
      1. Conditional and iterative control structures
      2. Basic data structures (primitive, arrays, records, pointers)
      3. Expression evaluation and statement execution
   B. Problem-solving, algorithms, and recursion (6 percent)
      1. Problems-solving strategies (top-down, functional decomposition)
      2. Properties of algorithms
      3. Recursive procedures (towers of Hanoi, generating permutations, divide and conquer)
   C. Abstract data types (7 percent)
      1. Abstract programming interfaces and encapsulation
      2. Specific ADT structures (stacks, queues, symbol tables, graphs)
   D. Object-oriented programming (5 percent)
      1. Object-oriented design
      2. Classes, subclasses, and inheritance
      3. Polymorphism

II. Software Engineering (13 percent)
   A. Software development life cycle (10 percent)
      1. Software processes and metrics
      2. Requirements and specifications
      3. Design and implementation
      4. Verification and validation
   B. Software tools and development methodologies (3 percent)
      1. Development, modeling, testing, and project management tools
      2. Team management, project planning, and project engineering

III. Computer Organization, Architecture and Operating Systems (23 percent)
   A. Logic design and data representation (4 percent)
      1. Logic gates and expressions, address
      2. Number representation (floating-point, signed, and two's-complement)
   B. Assembly level organization (3 percent)
      1. Assembly/machine language programming
      2. Instruction formats and addressing modes
C. Processes, memories, and communication (4 percent)
   1. CPU organization and implementation
   2. Memory organization and operation
   3. I/O control methods, bus, and switch

D. Operating Systems (8 percent)
   1. Scheduling and dispatch
   2. Virtual memory and device management
   3. Security and protection

E. Concurrency and communication (4 percent)
   1. Communication and synchronization
   2. Networking and distributed systems

IV. Algorithms, Theory and Computational Mathematics (27 percent)

   A. Fundamental algorithmic strategies and algorithms (5 percent)
      1. Algorithmic strategies (greedy, divide and conquer, backtrack)
      2. Basic algorithms (searching, sorting, BST, graph algorithms)

   B. Analysis, complexity, and correctness of algorithms (7 percent)
      1. Asymptotic analysis of upper and average space and time complexity bounds
      2. Using recurrence relations to analyze recursive algorithms
      3. Using formal specifications and assertions

   C. Automata and language theory (7 percent)
      1. Models of computation (finite automata, pushdown automata, Turing machines)
      2. Formal languages (regular languages, context-free languages)
      3. Modern computer language principles (types, encapsulation, etc.)

   D. Discrete structures (9 percent)
      1. Mathematical logic
      2. Elementary combinatorics, including graph theory and counting arguments
      3. Elementary discrete mathematics, including number theory, discrete probability, and recurrence relations

V. Special Topics (11 percent)

   A. Intelligent systems (3-4 percent)
      1. Search and optimization methods
      2. Knowledge representation and reasoning

   B. Information management (3-4 percent)
      1. Database systems
      2. Data modeling and relational model

   C. Human-computer interaction (2 percent)

   D. Net-centric computing (3 percent)
Scores on the Computer Science Test are reported as follows:

**Total Score**
Reported for each student and summarized for the group.

**Assessment Indicators**
Reported for the group* only.
- Programming Fundamentals (16)
- Computer Organization, Architecture and Operating Systems (14)
- Algorithms, Theory and Computational Mathematics (17)

Numbers in parentheses are approximate number of questions in each category.

*A minimum of five students is required for Assessment Indicators to be reported.
COMPUTER SCIENCE (4CMF)

(Courses introduced in January 2006)

The Major Field Test in Computer Science consists of 66 multiple choice questions, some of which are grouped in sets and based on such materials as diagrams, graphs, and program fragments. The outline below shows the content areas covered on the test and the approximate distribution of questions among content areas.

I. Discrete structures (15–21 percent)
A. Functions, relations, and sets
B. Basic logic
C. Proof techniques
D. Basics of counting and number theory
E. Graphs and trees
F. Discrete probability

II. Programming (21–27 percent)
A. Programming fundamentals
   1. Fundamental programming constructs
   2. Basic algorithms and problem solving
   3. Fundamental data structures
   4. Recursion
   5. Event-driven programming
   6. Object-oriented programming
B. Programming languages
   1. Features, paradigms, implementation techniques

III. Algorithms and complexity (16–22 percent)
A. Advanced data structures and algorithms (including graph algorithms)
B. Algorithmic strategies
C. Distributed algorithms
D. Basic computability and complexity
E. Automata theory

IV. Systems (16–24 percent)
A. Architecture
   1. Digital logic and digital systems
   2. Machine level representation of data
   3. Assembly level machine organization
   4. Interfacing and communication
B. Operating systems
   1. Operating system principles
   2. Concurrency
   3. Scheduling and dispatch
   4. Memory management
C. Networking

V. Software engineering (3–9 percent)
A. Software requirements, specifications, design, validation, and management

VI. Information management (3–8 percent)
A. Database systems
B. Data modeling

VII. Other (3–8 percent)
A. Human computer interaction
B. Graphics
C. Intelligent systems
D. Social and professional issues
E. Web computing

MFT Computer Science Pseudocode Statement

Scores on the Computer Science Test are reported as follows:

Total Score
Reported for each student and summarized for the group.

Assessment Indicators
Reported for the group* only.
- Programming (22)
- Discrete Structures and Algorithms (25)
- Systems (Architecture, Operating Systems, Networking, Database) (16)

Numbers in parentheses are approximate number of questions in each category.

* A minimum of five students is required for Assessment Indicators to be reported.
MFT Computer Science Pseudocode Statement

We currently do not use any specific programming languages in questions on the MFT Computer Science Exam. Instead, we use a simple pseudocode that we believe will be easily understood by any computer science student. See the examples below.

Example 1. Class declaration and object instantiation

```plaintext
class StudentInfo
  int studentID
  string name
end class StudentInfo

StudentInfo x ← new StudentInfo()
x.studentID ← 1234 // the value 1234 is assigned to x.studentID
x.name ← "John"
print ( x.studentID )
print ( x.name )
```

Example 2. The following procedure swaps the values of two parameters.

```plaintext
swap ( pass-by-reference int x, pass-by-reference int y )
  int temp ← x
  x ← y
  y ← temp
end swap
```

Example 3. SelectionSort

Preconditions: A is an array of integers.
The length of array A is n.
The index of array A starts at 0.

```plaintext
int[] selectionSort ( pass-by-reference int[] A, int n )
  int min
  int j
  int i ← 0
  while ( i ≤ n - 1 )
    min ← i
    j ← i + 1
    while ( j ≤ n - 1 )
        min ← j
      end if
      j ← j + 1
    end while
    if ( min ≠ i )
      swap ( A[min], A[i] )
    end if
    i ← i + 1
end while
return A // returns the sorted array
end selectionSort
```
Major Field Test in Computer Science
Sample Questions

The following questions illustrate the range of the test in terms of the abilities measured, the
disciplines covered, and the difficulty of the questions posed. They should not, however, be
considered representative of the entire scope of the test in either content or difficulty. An answer
key follows the questions.

1. If D is the accepting state of the nondeterministic finite automation above, which of the
following does the automaton accept?

   (A) 001
   (B) 1101
   (C) 01100
   (D) 000110
   (E) 100100

2. If a node in the binary search tree above is to be located by binary tree search, what is the
expected number of comparisons required to locate one of the items (nodes) in the tree chosen
at random?

   (A) 1.75
   (B) 2
   (C) 2.75
   (D) 3
   (E) 3.25
Questions 3 and 4 are based on the following information.

If the variables are suitably initialized, and if \( i \) remains within appropriate bounds, then the following code implements the stack operations \( \text{Push} \) and \( \text{Pop} \) when the stack is represented as an array \( V[1...N] \) with an index variable \( i \).

**Push:** begin \( V[i] := x \); \( i := i + 1 \); end

**Pop:** begin \( i := i - 1 \); \( x := V[i] \); end

3. Which of the following gives the correct initialization for this stack implementation?

   (A) \( i := 0 \)
   (B) \( i := 1 \)
   (C) \( i := N - 1 \)
   (D) \( i := N \)
   (E) \( i := N / 2 \)

4. If it is assumed that suitable changes in the initialization code were also made, which of the following changes to \( \text{Push} \) and \( \text{Pop} \) would yield a correct implementation of stacks?

   I. Replacing the code for \( \text{Push} \) with that for \( \text{Pop} \) and vice versa
   II. Making \( \text{Push} \) decrement \( i \) and \( \text{Pop} \) increment \( i \)
   III. Reversing the order of the statements in both \( \text{Push} \) and \( \text{Pop} \)

   (A) I only
   (B) II only
   (C) III only
   (D) I and II only
   (E) II and III only

5. In a computer with a cache memory interposed between the processor and the primary memory, the cache is \( k \)-way set-associative (for some fixed \( k \)); i.e., each location in primary memory “maps to” (can be cached in) any of \( k \) locations in the cache. Let there be \( P \) locations in primary memory and \( C \) locations in the cache.

   On the average, how many different locations in primary memory map to a particular location in the cache if \( k = 2 \)?

   (A) 1
   (B) 2
   (C) \( P / C \)
   (D) \( 2P / C \)
   (E) \( P \)
6. Which of the following regular expressions generate(s) no string with two consecutive 1's? 
(Note that \(\varepsilon\) denotes the empty string.)

1. \((1 + \varepsilon)(01 + 0)^*\)
2. \((01 + 10)^*\)
3. \((0 + 1)^*(0 + \varepsilon)\)

(A) I only
(B) II only
(C) III only
(D) I and II only
(E) II and III only

7. Which of the following C++ expressions does NOT always correctly compute the mathematical average of the integer variables \(a, b, c, \) and \(d\)?

(A) `float (a + b + c + d) / 4.0`
(B) `(float (a + b + c + d)) / 4`
(C) `(a + b + c + d) / 4`
(D) `(a + b + c + d) / 4.0`
(E) `(a + float(b) + c + d) / 4`

8. Consider the following recursive function.

```c
int Fun ( int n )
{
    if ( n == 4 )
        return 2;
    else
        return 2 * Fun ( n + 1 );
}
```

What is the value returned by the function call `Fun(2)`?

(A) 2
(B) 4
(C) 6
(D) 16
(E) 24
9. If A is an array with n elements and procedure Swap exchanges its arguments, then the following code segment sorts A in descending order.

```c
for ( int j = 0; j < n - 1; j++ )
for ( int k = 0; k < n - j - 1; k++ )
    Swap ( A[k] , A[k + 1] );
```

How many calls to Swap are made if initially, A[i] = i, for i = 0, 1, 2, ..., n - 1?

(A) n - 1  
(B) n  
(C) n(n - 1)/2  
(D) (n - 1)(n - 2)  
(E) n(n - 1)

10. Which of the following statements about static RAM (SRAM) and/or dynamic RAM (DRAM) is true?

(A) SRAM is implemented using transistors and capacitors that must be periodically refreshed.  
(B) DRAM has a faster access time than SRAM.  
(C) DRAM is less expensive than SRAM.  
(D) SRAM is capable of operating at speeds closely approximating processors.  
(E) DRAM is capable of operating at speeds closely approximating processors.

11. Suppose that \( S_1, S_2, ..., S_N \) is a set of fixed-length strings ordered alphabetically so that \( S_1 < S_2 < ... < S_N \). If these strings are held in order, in an array of \( N \) elements, then the time required to find the location of \( S_{i+1} \), given the location of \( S_i \), 1 ≤ i < N has order

(A) 1  
(B) \( \log_2 N \)  
(C) \( N \)  
(D) \( N \log_2 N \)  
(E) \( N^2 \)

12. Suppose that \( V \) is a vector with indices from \( a \) to \( b \) and that each element of \( V \) occupies two words. If the elements of \( V \) are stored in consecutive words of memory and \( a \) \( V[\alpha] \) is the address of word 1 of \( V[\alpha] \), then the address of word 1 of \( V[i] \), where \( a \leq i \leq b \), is

(A) \( (a \mod V[\alpha] ) + i \)  
(B) \( 2(a \mod V[\alpha] ) + 2i \)  
(C) \( (a \mod V[\alpha] ) - 2i \)  
(D) \( (a \mod V[\alpha] ) - 2a + i \)  
(E) \( (a \mod V[\alpha] ) - a + 2i \)
13. If A, B, C are Boolean variables, and if "\(\land\)" and "\(\lor\)" denote Boolean "and" and "or," respectively, which of the following is (are) true?

   I. \(A \land (B \lor C) = (A \land B) \lor (A \land C)\)
   II. \(A \lor (B \land C) = (A \lor B) \land (A \lor C)\)
   III. \((B \land A) \lor C = C \lor (A \land B)\)

   (A) I only  
   (B) II only  
   (C) I and II only  
   (D) II and III only  
   (E) I, II, and III

**Answer Key**

1. E  
2. C  
3. B  
4. E  
5. D  
6. A  
7. C  
8. C  
9. C  
10. C  
11. A  
12. C
2. College of ECC Graduating Senior Survey

Graduating Senior Survey
College of Engineering, Computer Science, and Construction Management
CSU, Chico

Dear Graduating Senior,

The College of ECC has developed the enclosed survey to give you a forum for letting us know what you think of your experience at CSU, Chico, and to help us to continually improve the curriculum and services we offer. We care a great deal about the programs and your feedback is essential to helping us provide the highest quality education we can deliver. Thank you in advance for your time and attention to this survey.

We hope the years you have spent with us have enriched your life and provided you with the foundation for a successful career. Please stay in touch!

With best wishes, The College of ECC Faculty

1. Major
   - □ APOG
   - □ CMPE
   - □ CS
   - □ EE
   - □ ME
   - □ MECA
   - □ MFT

2. Graduation date
   - □ Spring, Year 2009
   - □ Fall, Year 2009

3. Did you come to Chico State as a...
   - □ First-time freshman
   - □ Transfer

4. How many semesters did you attend Chico State?
   - □ 1-3
   - □ 4-6
   - □ 7-9
   - □ 10-12
   - □ 13+

5. What is your overall GPA?
   - □ Below 2.25
   - □ 2.25-2.49
   - □ 2.50-2.74
   - □ 2.75-2.99
   - □ 3.00-3.24
   - □ 3.25-3.50
   - □ 3.51-3.74
   - □ 3.75-4.00

6. If you had an internship, co-op, or job related to your major while in school, how valuable was the experience?
   - □ Did not have internship, co-op, or job
   - □ Very valuable
   - □ Valuable
   - □ Somewhat valuable
   - □ Not valuable

7. If you were involved in any student/professional society, activities, or clubs, how valuable was the experience?
   - □ Was not involved in societies, activities, or clubs
   - □ Very valuable
   - □ Valuable
   - □ Somewhat valuable
   - □ Not valuable

8. Immediately after graduating are you planning to...
   - □ Attend graduate school
     - Yes
     - No
   - □ Begin working
     - Yes
     - No
   - □ If you are NOT planning to work full-time, or if you have not begun looking for a job, please skip to Question 13.

9. How many job offers have you received?
   - □ None
   - □ One
   - □ Two
   - □ Three
   - □ Four
   - □ Four+

10. Do you currently have a job offer that you are likely to accept?
    - □ Yes
    - □ No

If 'Yes,' please provide:
   Company name
   Your job title
   Starting annual salary
   □ Less than $30K
   □ $30-40K
   □ $41-50K
   □ $51-60K
   □ $61-70K
   □ $71K or more

11. If you interviewed through the campus Career Planning & Placement Office, how helpful was it?
    - □ Did not interview through campus office
    - □ Very helpful
    - □ Helpful
    - □ Somewhat helpful
    - □ Not helpful

12. If you found a job that you are likely to accept, how did you find it?
    - □ Campus Career Planning & Placement Office
    - □ Faculty/department referral
    - □ Online posting
    - □ Mailed resume
    - □ Personal connections
    - □ Other

13. Did you take a comprehensive exam (FE, CMGT, MFT or other) for your discipline?
    - □ No, did not take
    - □ Yes, and passed
    - □ Yes, and did not pass
    - □ Yes, and waiting for results

14. If you took a comprehensive exam, did you also attend a review course to prepare you for the exam?
    - □ Yes
    - □ No

If 'Yes,' how valuable was the course?
   - □ Very valuable
   - □ Valuable
   - □ Somewhat valuable
   - □ Not valuable
### Educational Satisfaction Questions

At Chico State, how satisfied were you with the...

15. Quality of teaching by faculty in your department
16. Quality of teaching by other faculty
17. Access to faculty in your department
18. Availability of courses in your department
19. Quality of courses in your department
20. Access to laboratory facilities and equipment
21. Quality of laboratories and equipment
22. Access to computer facilities
23. Quality of computer facilities
24. Academic advising from your major advisor
25. Academic advising from the University Advising Office
26. Career information from your department
27. Availability of General Education courses
28. Quality of General Education courses
29. The overall quality of your education
30. Your overall experience at Chico State

### Program Outcomes Questions

Based on your educational experience here at Chico State how well prepared are you to...

31. Apply knowledge of math, science, engineering, or technology to solve problems
32. Design and conduct experiments
33. Analyze and interpret experimental data
34. Design a component or system to meet desired needs
35. Function on a multidisciplinary team
36. Identify, formulate, and solve technical problems
37. Communicate technical matters in writing
38. Communicate technical matters orally
39. Understand and apply professional and ethical principles
40. Understand contemporary issues facing society
41. Use modern tools and technology
42. Enter the workplace
43. Continue learning

44. I would recommend my major program at CSU, Chico to others.

### Supplemental Questions


Please locate the supplemental questions on the sheet provided. Enter your responses to the right.

Thank you for completing the survey, and please stay in touch with us!