SCHOOL OF COMPUTING & INFORMATION SCIENCES

Annual Assessment Summary 2010-2011
for the
Bachelor of Science in Computer Science

Prepared by
Norman Pestaina – Assessments Coordinator

November 1, 2011

I. INTRODUCTION

This report is prepared in accordance with the Assessment Plan adopted by the School of Computing & Information Sciences (then the School of Computer Science) in spring 2003. Its purpose is to summarize the results of the various assessment mechanisms utilized by the SCIS in support of the BS in Computer Science program, and to present the resulting findings and recommendations to the Undergraduate Committee, directors and faculty of the School.

The goals of the annual assessment process are to assess the extent to which the Student Outcomes and Program Educational Objectives of the BS in Computer Science program have been attained in the period under review, to identify specific areas of the program where a need for improvement is indicated, and to present a set of recommendations for achieving those improvements.

The period under review includes the spring, summer and fall semesters of 2010, and the spring semester of 2011.

II. OVERVIEW

A. Terminology

The BS in Computer Science *Program Educational Objectives* describe the overriding goals of the program relating to the cumulative persistent effects of the students' educational experiences. The objectives are broad in nature and define expected general characteristics of the program's graduates within some years after graduation.

The BS in Computer Science *Student Outcomes* are more specific in nature. These describe characteristics of students at the time of graduation, and define the specific knowledge, skills, and behaviors that students are expected to acquire as they complete the requirements of the program. Attainment of each Student Outcome enables the attainment of one or more of the Program Educational Objectives.

Additionally, the syllabus of each required and elective course of the BS in Computer Science program presents a set of *Course Outcomes*. The Course Outcomes identify specific knowledge units and levels of attainment (mastery, familiarity, awareness) expected of a student completing the course. Attainment by students of Course Outcome enables attainment of one or more of the Student Outcomes.

The *Program Educational Objectives* and *Student Outcomes* were amended in fall 2010. Until fall 2010, these were referred to as Program Objectives and Program Outcomes respectively. Both original and current Objectives and Outcomes are presented in Appendix A.

B. Assessment Mechanisms & Procedures

Consistent with current educational practice, the SCIS follows a systematic process of collecting and utilizing data on the degree of attainment of the Student Outcomes and Program Educational Objectives. The *SCIS Assessment Plan* specifies the participants and schedule for this process, and the means of evaluating the data and enacting program changes indicated by the evaluation. The *SCIS Assessment Mechanisms & Procedures* document specifies the implementation of the Assessment Plan. The SCIS Assessment Plan and Assessment Procedures and Mechanisms were adopted in 2003, and amended in 2010 to incorporate additional direct assessment measures.

The *SCIS Assessment Plan* and *SCIS Assessment Mechanisms & Procedures* documents are presented as Appendix B to this report.

The following indirect assessment mechanisms have been employed since spring 2003:

Mechanism	**Target**	**Frequency**
Course Outcomes Survey by Students	Course Outcomes	Semester
Course Outcomes Survey by Instructors	Course Outcomes	Semester
Graduating Student (Exit) Survey	Student Outcomes	Semester
Alumni Survey	Program Educational Objectives	Continual

The following direct assessment mechanisms have been employed since spring 2010:

Mechanism	**Target**	**Frequency**
Course Embedded Assessment	Course and Student Outcomes	Semester
Capstone Project Assessment	Student Outcomes	Semester

Additional input is solicited and may be received from other program constituents including:

- ACM Student Chapter,
- Upsilon Pi Epsilon Honor Society Chapter,
- SCIS Women In Computer Science group
- SCIS Industry Advisory Board.

C. **Process**

The required and elective courses of the BS in Computer Science are each assigned, based on subject area, to one of six groups: Communications and Ethics, Computer Organization, Computer Systems, Foundations, Programming, and Software Engineering.

Each subject area group is managed by a faculty Subject-Area Coordinator. Periodically, the assessment data and comments from Student and Instructor Course Outcome Surveys are considered by the Subject Area Coordinators. These provide the information for the Subject Area Coordinators' reports.

The Subject Area Coordinator reports and assessment data from all other sources are evaluated by the SCIS Assessments Coordinator whose evaluations and recommendations are presented in an assessment report.

The assessment report is considered by the SCIS Undergraduate Committee, and by the SCIS Undergraduate Program Director. The Undergraduate Committee's curricular recommendations are presented to the SCIS faculty for approval. Responsibility for enactment of approved recommendations rests with the SCIS Undergraduate Program Director.

III. DATA

A. Course Outcomes Survey by Students

This survey is completed by students in each section of a required or elective CS class. For each course outcome, the student states the extent to which s/he agrees with each of two assertions
1: *I believe that this is a valuable outcome for this course*, and
2: *The subject matter of this outcome was covered adequately in class*
To each assertion, the student responds on a 5-point scale as follows:
5: I agree strongly, 4: I agree moderately, 3: I am not sure, 2: I disagree moderately, 1: I disagree strongly
For each outcome, a weighted mean of the responses to each question is calculated. The means are provided for each course, cumulatively over all semesters of the period under review.

	BS in Computer Science Required or Elective Course	# Responding	Value of Outcome	Coverage Adequacy
CAP 4770	Introduction to Data Mining	15	4.58	4.27
CDA 4101	Structured Computer Organization	52	4.49	4.47
CEN 4010	Software Engineering I	53	4.65	4.55
CEN 4021	Software Engineering II	14	4.57	4.43
CEN 4072	Software Testing	17	4.56	4.34
CGS 1920	Introduction to Computing	149	4.68	4.62
CGS 3092	Ethics & Social Issues in Computing	197	4.65	4.66
CIS 4911	Senior Project	41	4.69	4.09
COP 2210	Computer Programming I	280	4.46	4.36
COP 3337	Computer Programming II	178	4.54	4.44
COP 3402	Fundamentals of Computer Systems	120	4.52	4.64
COP 3530	Data Structures	112	4.56	4.34
COP 4225	Advanced Unix Programming	19	4.16	3.89
COP 4226	Advanced Windows Programming	21	4.54	4.48
COP 4338	Computer Programming III	66	4.36	4.11
COP 4520	Introduction to Parallel Computing	11	4.43	4.23
COP 4540	Database Management	44	4.63	4.19
COP 4555	Principles Programming Languages	63	4.38	4.46
COP 4610	Operating Systems Principles	48	4.41	4.07
COT 3420	Logic for Computer Science	58	3.86	3.53
MAD 2104	Discrete Mathematics	12	4.37	4.30
MAD 3512	Theory of Algorithms	6	3.94	4.07
		======	======	======
			4.51	4.40

Table 1: Value & Adequacy of Coverage of Course Outcomes 01/10 – 04/11

Note: This if the first report for which data is available for CGS 1902, MAD 2104, MAD 3512.

Note: MAD 2104 and MAD 3512 are taught by Math faculty. Students in these sections complete the surveys on-line voluntarily, unlike students in CS sections who do so in class.

The semester data for each course are presented here grouped under the six subject areas. The Subject Area Coordinator (SAC) reports are included as Appendix C to this assessment report.

Subject Area: Communications & Ethics (SAC: Tiana Solis)
CGS1920 Introduction to Computing
CGS 3092 Professional Ethics and Social Issues in Computer Science
COM 3110 Business and Professional Communication
ENC 3213 Report and Technical Writing

CGS 1920	Introduction to Computing		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	53	4.59	4.50
Fall '10	51	4.71	4.69
Spring '11	45	4.76	4.68
	=======	=======	=======
	149	4.68	4.62

Table 2-CGS1920: Student Rating of Course Outcomes

CGS 3092	Professional Ethics and Social Issues in Computing		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	35	4.63	4.65
Summer '10	34	4.66	4.59
Fall '10	63	4.71	4.75
Spring '11	65	4.59	4.60
	=======	=======	=======
	197	4.65	4.66

Table 2-CGS3092: Student Rating of Course Outcomes

Subject Area: Computer Organization (SAC: Nagarajan Prabakar)
COP 3402 Fundamentals of Computer Systems … now renumbered to **CDA 3103**
CDA 4101 Structured Computer Organization
COP 4610 Operating Systems Principles

COP 3402	Fundamentals of Computer Systems		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	38	4.52	4.57
Summer '10	13	4.72	4.77
Fall '10	26	4.39	4.57
Spring '11	43	4.53	4.70
	=======	=======	=======
	120	4.52	4.64

Table 2-COP3402: Student Rating of Course Outcomes

CDA 4101	Computer Organization		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	22	4.46	4.33
Fall '10	19	4.44	4.51
Spring '11	11	4.64	4.67
	=======	=======	=======
	52	4.49	4.47

Table 2-CDA4101: Student Rating of Course Outcomes

COP 4610	Operating Systems Principles		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	16	4.34	4.23
Summer '10	16	4.71	4.63
Fall '10	10	3.98	3.82
Spring '11	6	4.53	2.57
	=======	=======	=======
	48	4.41	4.07

Table 2-COP4610: Student Rating of Course Outcomes

Subject Area: **Computer Systems** (SAC: Shu-Ching Chen)
COP 4540 Database Management systems … now renumbered to **COP 4710**
CAP 4770 Principles of Data Mining
COP 4225 Advanced UNIX Programming … now renumbered to **COP 4604**

COP 4540	Database Management		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	18	4.49	4.01
Fall '10	18	4.73	4.20
Spring '11	8	4.70	4.59
	44	4.63	4.19

Table 2-COP4540: Student Rating of Course Outcomes

CAP 4770	Introduction to Data Mining		
	# Responding	Value of Outcome	Coverage Adequacy
Fall '10	15	4.58	4.27
	15	4.58	4.27

Table 2-CAP4770: Student Rating of Course Outcomes

COP 4225	Advanced Unix Programming		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	4	3.91	3.79
Summer '10	11	4.39	4.32
Spring '11	4	3.79	2.83
	19	4.16	3.89

Table 2-COP4225: Student Rating of Course Outcomes

Subject Area: **Foundations** (SAC: Xudong He)
COP 4555 Principles of Programming Languages
COT 3402 Logic for Computer Science
MAD 2104 Discrete Mathematics
MAD 3512 Theory of Algorithms
Set 2 (Math) Electives (**MAD 3305**, **MAD 3402**, **MAD 4203**, **MHF 4302**)

COP 4555	Principles Programming Languages		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	22	4.44	4.49
Fall '10	21	4.45	4.48
Spring '11	20	4.25	4.41
	63	4.38	4.46

Table 2-COP4555: Student Rating of Course Outcomes

COT 3420	Logic for Computer Science		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	20	4.05	3.11
Summer '10	11	3.84	3.59
Fall '10	10	3.83	4.28
Spring '11	17	3.68	3.54
	58	3.86	3.53

Table 2-COT3420: Student Rating of Course Outcomes

MAD 2104	Discrete Mathematics		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	5	4.75	4.75
Fall '10	7	4.10	3.97
	12	4.37	4.30

Table 2-MAD2104: Student Rating of Course Outcomes

MAD 3512	Theory of Algorithms		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	3	4.07	3.87
Summer '10	2	3.20	3.90
Fall '10	1	5.00	5.00
	=======	=======	=======
	6	3.94	4.07

Table 2-MAD3512: Student Rating of Course Outcomes

Set 2 (Math) Electives

MAD 3305 Graph Theory
MAD 3402 Numerical analysis
MAD 4203 Introduction to Combinatorics
MHF 4302 Mathematical Logic

The Set 2 Elective courses are taught by faculty of the Mathematics Department. There are no assessment data for these courses.

Subject Area: Programming (SAC: Tim Downey)
COP 2210 Computer Programming I
COP 3337 Computer Programming II
COP 3530 Data Structures
COP 4226 Advanced Windows Programming
COP 4338 Computer Programming III

COP 2210 Computer Programming I

	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	125	4.46	4.38
Summer '10	30	4.07	3.67
Fall '10	72	4.56	4.49
Spring '11	53	4.56	4.53
	=======	=======	=======
	280	4.46	4.36

Table 2-COP2210: Student Rating of Course Outcomes

COP 3337 Computer Programming II

	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	25	4.42	4.48
Summer '10	20	4.40	4.03
Fall '10	75	4.60	4.56
Spring '11	58	4.57	4.41
	=======	=======	=======
	178	4.54	4.44

Table 2-COP3337: Student Rating of Course Outcomes

COP 3530 Data Structures

	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	40	4.50	4.27
Summer '10	15	4.75	4.67
Fall '10	23	4.45	4.15
Spring '11	34	4.62	4.39
	=======	=======	=======
	112	4.56	4.34

Table 2-COP3530: Student Rating of Course Outcomes

COP 4226	Advanced Windows Programming		
	# Responding	Value of Outcome	Coverage Adequacy
Fall '10	21	4.54	4.48
	=======	=======	=======
	21	4.54	4.48

Table 2-COP4226: Student Rating of Course Outcomes

COP 4338	Computer Programming III		
	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	16	4.51	4.24
Fall '10	30	4.58	4.30
Spring '11	20	3.91	3.73
	=======	=======	=======
	66	4.36	4.11

Table 2-COP4338: Student Rating of Course Outcomes

Subject Area: **Software Engineering** (SAC: Peter Clarke)
CEN 4010 Software Engineering I
CEN 4021 Software Engineering II
CEN 4072 Software Testing
CIS 4911 Senior Project

CEN 4010 Software Engineering I

	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	5	5.00	5.00
Summer '10	14	4.52	4.55
Fall '10	13	4.77	4.54
Spring '11	21	4.57	4.46
	53	4.65	4.55

Table 2-CEN4010: Student Rating of Course Outcomes

CEN 4021 Software Engineering II

	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	7	4.75	4.89
Spring '11	7	4.39	3.96
	14	4.57	4.43

Table 2-CEN4021: Student Rating of Course Outcomes

CEN 4072 Software Testing

	# Responding	Value of Outcome	Coverage Adequacy
Fall '10	17	4.56	4.34
	17	4.56	4.34

Table 2-CEN4072: Student Rating of Course Outcomes

CIS 4911 Senior Project

	# Responding	Value of Outcome	Coverage Adequacy
Spring '10	5	4.98	4.75
Fall '10	16	4.69	4.17
Spring '11	20	4.62	3.86
	41	4.69	4.09

Table 2-CIS4911: Student Rating of Course Outcomes

B. Course Outcomes Survey by Instructors

This survey is completed by each instructor of a required or elective CS course section.

- The Instructor separately rates the individual course outcomes in respect of two criteria
  - Appropriateness: *Essential* *Very Appropriate* *Appropriate* *Inappropriate*
  - Coverage: *Extensive* *Adequate* *Not Enough* *Not At All*
- The Instructor separately rates the course prerequisites in respect of two criteria
  - Relevance: *Irrelevant* *Incidental* *Useful* *Highly Useful*
  - Student Mastery: *Good* *Adequate* *Deficient* *Non-existent*
- The Instructor rates the students' overall preparation for taking the course
  - Student Preparation: *Good* *Adequate* *Deficient* *Non-existent*
- In addition, the Instructor may append general comments and suggestions specific to each course prerequisite or outcome.

These responses, comments and suggestions from the Instructor surveys, together with the data from the Student Course Outcomes surveys (see **Table 1**, above), form the basis of the Subject Area Coordinators' reports. The Coordinators' summaries as they relate to the Instructors' evaluations are represented here. As noted in the preceding section, the complete SAC reports from which these observations are taken are included as Appendix C. **Note:** The data here are qualitative; no numeric scores are assigned to responses.

Subject Area: Communications & Ethics (SAC Tiana Solis)

CGS1920 Introduction to Computing

- *All objectives were covered on an assignment or guest lectures or research activities.*
- *All objective were considered essential.*
- *Most objectives were covered extensively except for undergraduate research opportunities*

CGS 3092 Professional Ethics and Social Issues in Computer Science

- *All objectives were covered on an assignment or in an in class discussion.*
- *All objective were considered essential.*
- *Most objectives were covered extensively except for team problem solving.*
- *Most prerequisite objectives currently listed include specific programming skills that were considered incidental. Making ENC3211 Technical writing a pre-requisite for this course significantly improved the outcome of most objectives*

COM 3011 Business and Professional Communication

ENC 3213 Report and Technical Writing

- *COM 3011 and ENC 3211 are taught by other instructional units and consequently are not subject to the School's assessment mechanisms.*

Subject Area: Computer Organization (SAC Nagarajan Prabakar)

COP 3402 Fundamentals of Computer Systems … now renumbered to **CDA 3103**

- Only one faculty appraisal suggests to re-consider the proficiency level of each of the out*comes.*

CDA 4101 Structured Computer Organization

- *There is no significant concern expressed by the students or faculty.*

COP 4610 Operating Systems Principles

- *Also, the lack of adequate of prerequisite skills among a subset of students is cited in the faculty course appraisals.*

Subject Area: **Computer Systems** (SAC Shu-Ching Chen)

COP 4540 Database Management systems … now renumbered to **COP 4710**

- *Summary of Assessment: This course has seven outcomes, all of which has been indicated by the instructors as either essential or appropriate.*

CAP 4770 Principles of Data Mining

- *Summary of Assessment: This course has six outcomes, all of which has been indicated by the instructors as either essential or very appropriate.*

COP 4225 Advanced UNIX Programming … now renumbered to **COP 4604**

- *Summary of Assessment: This course has six outcomes, all indicated by the two instructors as essential.*

Subject Area: **Foundations** (SAC Xudong He)

COP 4555 Principles of Programming Languages

COT 3402 Logic for Computer Science

- *In the appraisals, Alex found the students' preparation deficient. Alex observed the continued decline of student quality and suggested to tighten the admission requirements.*

MAD 2104 Discrete Mathematics

- *…there are no instructor appraisals*

MAD 3512 Theory of Algorithms

- *…there are no instructor appraisals*

Set 2 (Math) Electives (**MAD 3305**, **MAD 3402**, **MAD 4203**, **MHF 4302**)

- *(Assessments Coordinator:) These courses are taught by the Math department faculty and consequently are not subject to the School's assessment mechanisms.*

Subject Area: **Programming** (SAC Tim Downey)

COP 2210 Computer Programming I

- *All objectives are considered essential or appropriate.*
- *Most of the instructors thought that the students' preparation for taking the course was adequate. One instructor felt that it was deficient.*

COP 3337 Computer Programming II

- *All objectives are considered essential or appropriate.*
- *All objectives were covered extensively or adequately. Student evaluations confirm the instructor's appraisals, except for one class.*
- *All prerequisite objectives were considered highly useful.*
- *Student prerequisite preparation was generally good and adequate; several instructors reported multiple deficiencies for their students. Three sections reported a deficiency in Strings/ArrayLists. One section reported a deficiency in Objects/Classes.*

COP 3530 Data Structures

- *All objectives are considered essential or appropriate.*
- *All objectives were covered extensively or adequately. Student responses supported this; except for a 3.97/5.00 for one semester in recursion.*
- *All but one instructor indicated that the mastery of the prerequisites was at least adequate.*

COP 4226 Advanced Windows Programming

- *All objectives are considered essential or appropriate.*
- *All objectives were covered extensively or adequately. Student responses supported this, except for the outcome "Database Connectivity, Serialization, Drag and Drop, Multithreaded Programming".*

COP 4338 Computer Programming III

- *Objectives were met less adequately as time progressed. By Spring 2011, Java and C++ were no longer being covered. In all semesters, reflection and STL were not covered adequately.*
- *The relevant prerequisite objectives were rated from highly useful to irrelevant. This corresponds to the change in the objectives.*
- *The preparation of the students was rated good, except for Spring 2011 where it was rated deficient due to lack of UNIX experience.*

Subject Area: **Software Engineering** (SAC Peter Clarke)

CEN 4010 Software Engineering I

- *All the course objectives were either extensively or adequately covered for all the semesters*.

CEN 4021 Software Engineering II

- *All the course objectives were either extensively or adequately covered for all the semesters*

CEN 4072 Software Testing

- *There was no assessment done for the course in Fall 2010. This was the first time the course was offered and the online instructor course assessment was not ready.*

CIS 4911 Senior Project

- *All except one of the course objectives were either extensively or adequately covered for all the semesters. The only course objective that was not adequately covered was Ethical Issues*.

C. Graduating Student (Exit) Survey of Student Outcomes

The Student Outcomes Survey is completed by students in the semester in which they expect to graduate. The student rates each outcome in respect of two criteria, attainment and relevance.

Attainment: *This program outcome has been met for me personally*

5: I agree strongly	*2: I disagree somewhat*
4: I agree moderately	*1: I disagree moderately*
3: I agree somewhat	*0: I disagree strongly*

Relevance: *How meaningful do you consider this outcome to be for you personally?*

5: Extremely meaningful	*2: Somewhat meaningless*
4: Moderately meaningful	*1: Moderately meaningless*
3: Somewhat meaningful	*0: Extremely meaningless*

The following table summarizes the responses of 28 graduating students completing the survey. The mean responses are expressed as percentages of 5, the maximum rating. The raw data from the 28 completed surveys are provided in Appendix D.

Exit Survey (Graduating Students) 28 Respondents	*Outcome Attainment*		*Perceived Relevance*	
Student Outcomes	*Mean*	*Percentage*	*Mean*	*Percentage*
a: Proficiency in foundation areas	4.46	89.2	4.46	89.2
b: Proficiency in core areas	4.50	90.0	4.79	95.8
c: Proficiency in problem solving	4.29	85.8	4.68	93.6
d: Proficiency in a programming language	4.57	91.4	4.75	95.0
e: Understanding of social & ethical issues	4.18	83.6	3.86	77.2
f: Ability to work cooperatively in teams	4.39	87.8	4.46	89.2
g: Effective communication skills	4.39	87.8	4.46	89.2
h: Experience with contemporary environments & tools	3.86	77.2	3.86	77.2
	====	====	====	====
	4.33	86.6	4.42	88.3
	====	====	====	====
Overall Satisfaction for CS Areas, Outcomes a..e			4.51	90.2

Table 3: Exit Survey of Attainment & Relevance of Student Outcomes

Note: Original Program Outcomes relating to understanding of the scientific method and familiarity with the arts and humanities, and Program Outcomes relating to success in employment or admission to graduate school, were dropped as part of the Fall 2010 revision. There are no corresponding Student Outcomes.

Note: Student Outcome h) has been renumbered from j) as part of the revision of Student Outcomes in Fall 2010. This Outcome was rephrased in response to a recommendation of the 2009 assessment cycle.

D. Alumni Survey of Program Educational Objectives

Alumni responding to the survey are asked to rate the contribution of their broad educational experience at FIU to their personal growth, capacity for life-long learning, communication skills, social and ethical awareness, career preparation, and preparation for graduate study. They rate their preparation in the major areas of the BS-CS curriculum. The respondents also provide "overall" ratings of their FIU educational experience and the student's preparation at graduation. Finally, the alumni provide a rating of their overall satisfaction with the BS in CS program.

Responses to the survey questions are on a the following scale

4: Excellent, 3: Good, 2: Satisfactory, 1: Poor and 0: Unsatisfactory

The following table summarizes the responses to this survey. The means for the current survey cycle, 5/26/2007 through 6/20/2011, are compared with corresponding means for earlier cycles, 2/11/2004 through 2/28/2007. The raw data for the current cycle are provided in Appendix E.

		2/11/2004 2/28/2007		5/26/2007 6/20/2011	
	Alumni Survey of Program Objectives	125 Respondents		16 Respondents	
		Outcome Attainment		*Outcome Attainment*	
	Program Educational Objective	*Average*	*Percentage*	*Average*	*Percentage*
1	Capacity for personal growth	3.35	83.75	3.31	82.75
1	Capacity for life-long learning	3.45	86.25	3.19	79.75
3	Development of communication skills	2.90	72.50	2.94	73.50
3	Awareness of social, ethical responsibility	2.94	73.50	3.25	81.25
4	Preparation for career in CS	3.18	79.50	3.19	79.75
4	Preparation for graduate study	3.08	77.00	2.88	72.00
4	Overall preparation upon graduation	3.10	77.50	3.06	76.50
2	Computer Programming	3.37	84.25	3.13	78.25
2	Systems Development	2.82	70.50	2.81	70.25
2	Data Structures & Algorithms	3.29	82.25	3.44	86.00
2	Computer Architecture & Organization	2.94	73.50	2.88	72.00
	Overall FIU educational experience	3.15	78.75	3.13	78.25
	Overall satisfaction with BS-CS program	3.14	78.50	3.11	77.75

Table 4: Alumni Survey of Attainment of Program Educational Objectives

Note: Original Program Objectives relating to student diversity and faculty excellence were dropped as part of the fall 2010 revision. Although the survey solicits ratings of those criteria, the related data are not relevant to this evaluation and are not presented here.

E. Course Embedded Assessment

SCIS began applying course-embedded assessment of the BS in CS program in fall 2010 in order to supplement the direct measures obtained via capstone assessment in the Senior Project (see the following section). This strategy was applied using multiple-choice (M-C) quizzes as shown below. The data presented in the following tables are reconstructed from the Direct Assessment Summaries for fall 2010 and spring 2011, included as Appendix E to this report. The raw data are provided as appendix F.

Student Outcome a)

Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.

Fall 2010 MAD 3512 Theory of Algorithms — 5 M-C questions, 4 samples*

Correct Answers	# of Students	Cumulative %	
5 = 100%	0	0.0	(0/4)
4 = 80%	0	0.0	(0/4)
3 = 60%	1	25.0	(1 /4)
2 = 40%	2	75.0	(3/4)
1 or 0	1	100.0	(4/4)

*The available data were considered insufficient for any meaningful analysis.

Table 5-MAD3512-Fall2010: Course-Embedded Data

Spring 2011 MAD 3512 Theory of Algorithms — 4 M-C questions, 12 samples

Correct Answers	# of Students	Cumulative %	
4 = 100%	6	50.0	(6 /12)
3 = 75%	3	75.0	(9 /12)
2 = 50%	2	92.0	(11/12)
1 or 0	1	100.0	(12/12)

Table 5-MAD3512-Spring2011: Course-Embedded Data

Student Outcome b)

Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.

Fall 2010 COP 4555 Principles of Programming Languages — 10 M-C questions, 19 samples

Correct Answers	# of Students	Cumulative %	
10 = 100%	2	10.5	(2 /19)
9 = 90%	1	15.8	(3/19)
8 = 80%	4	36.8	(7/19)
7 = 70%	5	63.2	(12/19)
6 = 60%	1	68.4	(13/19)
5 = 50%	4	89.5	(17/19)
4 = 40%	2	100.0	(19/19)

Table 5-COP4555-Fall2010: Course-Embedded Data

Spring 2011 COP 3530 Data Structures 10 M-C questions
25 samples

Correct Answers	# of Students	Cumulative %	
10 = 100%	0	0.0	(0 /25)
9 = 90%	2	8.0	(2/25)
8 = 80%	1	12.0	(3/25)
7 = 70%	6	36.0	(9/25)
6 = 60%	7	64.0	(16/25)
5 = 50%	4	80.0	(20/25)
4 = 40%	3	92.0	(23/25)
3 = 30%	2	100.0	(25/25)

Table 5-COP3530-Spring2011: Course-Embedded Data

Spring 2011 COP 4540 Database Management 5 M-C questions
13 samples

Correct Answers	# of Students	Cumulative %	
5 = 100%	5	38.0	(5/13)
4 = 80%	7	92.0	(12/13)
3 = 60%	1	100.0	(13/13)

Table 5-COP4540-Spring2011: Course-Embedded Data

Student Outcome e)
Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.

Fall 2010 CGS 3092 Professional Ethics and Social Issues in Computing 10 samples
Each of 10 samples was scored on an 8-point scale using a specialized rubric

Rubric Score	# of Students	Cumulative %	
8 = 100%	8	80.0	(8/10)
4 = 50%	2	100.0	(10/10)

F. Capstone Project Assessment

Current requirements of the BS in Computer Science include completion of a capstone course, CIS 4911 Senior Project. Beginning with the first offering of CIS 4911, SCIS has performed assessment of all Student Outcomes via evaluation of the presentations and artifacts of all completed projects. Each project is rated by 2 or more evaluators according to a rubric *Senior Project Assessment of Student Outcomes of the BS in Computer Science*, and scored on the following scale:

Rating	Criterion
n/a	The project does not provide clear evidence about this particular outcome
1	The project demonstrates **poor** attainment of this outcome
2	The project demonstrates **fair** attainment of this outcome
3	The project demonstrates **good** attainment of this outcome
4	The project demonstrates **very good** attainment of this outcome
5	The project demonstrates **excellent** attainment of this outcome

Based on experience gained with application in each of the spring 2010 and fall 2010 semesters, the rubric has evolved through 3 iterations to the version of spring 2011. The spring 2011 version of the rubric, and associated check-list and score grid are included as Appendix G of this report.

The data from these assessment events are summarized in the following table. The sources of these data are the Direct Assessment Summary documents of fall 2010 and spring 2011, and the tabulated (pilot) project ratings for spring 2010 included as Appendix E of this report.

	Outcome (a)	Outcome (b)	Outcome (c)	Outcome (d)	Outcome (e)	Outcome (f)	Outcome (g)	Outcome (h)
SP'10								
Proj.1	2.0	3.8	4.6	4.0	3.2	4.6	4.6	4.4
Proj.2	0.7	4.5	4.7	4.7	4.3	5.0	4.7	4.7
FL'10								
Proj.1	3.0	3.0	5.0	5.0	4.0	5.0	5.0	5.0
Proj.2	1.0	2.0	5.0	5.0	5.0	5.0	5.0	4.0
Proj.3	4.5	4.5	4.5	4.5	4.5	5.0	4.5	4.5
Proj.4	3.0	2.0	5.0	5.0	1.0	5.0	5.0	4.0
Proj.5	3.5	4.5	5.0	5.0	5.0	5.0	5.0	5.0
Proj.6	3.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0
Proj.7	3.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0
SP'11								
Proj.1	3.5	5.0	5.0	5.0	3.0	5.0	5.0	5.0
Proj.2	3.0	3.0	5.0	5.0	5.0	5.0	5.0	5.0
Proj.3	5.0	5.0	5.0	5.0	5.0	5.0	5.0	4.0
Proj.4	1.0	2.0	5.0	4.0	4.0	5.0	5.0	5.0
Proj.5	4.0	4.0	5.0	4.0	4.0	5.0	5.0	5.0

Table 6: Summarized Senior Project Assessment Ratings

IV. EVALUATION

In this section of the report, the data presented in the previous section are evaluated. For quantitative data, the threshold value at which SCIS deems a measured item to satisfy its criteria is 75% of the maximum attainable rating.

Measured Item	Scale	Threshold
Course Outcomes	1 to 5	3.75
Student Outcomes	0 to 5	3.75
Program Objectives	0 to 4	3.00

A. Course Outcomes

The SAC reports (Appendix C) present the data obtained for each course via surveys by students and instructors. The Course Outcomes for each required or elective course of the BS in Computer Science program are evaluated for relevance and attainment by the Subject Area Coordinators (SAC). The Subject Area Coordinators' evaluations are contained in their reports.

The Assessment Coordinator's evaluation of the Course Outcomes is based on the student ratings of the course outcomes summarized in Table 1.

AC-Evaluation-01: The response rates to the Survey of Course Outcomes by Students for the MAD courses are very low, especially considering the relatively large number of students taking MAD 2104 each semester.

AC-Evaluation-02: The Value of Course Outcomes rating of every course exceeds the 3.75 acceptability threshold. The value of Course Outcomes for COT 3420 Logic for Computer Science, and MAD 3512 Theory of Algorithms, are rated as **acceptable** at 3.86 and 3.94 respectively. With the exception of COT 3420 and MAD 3512, students ascribe at least **high** value (4.00 or higher) to the outcomes of every course. In fact, the rating of the Value of Course Outcomes of a majority of courses is **very high** (4.50 or higher).

AC-Evaluation-03: The student rating of the Adequacy of Coverage of Course Outcomes in COT 3420 is **low** at 3.53, below the acceptability threshold of 3.75. The student rating of the Adequacy of Coverage of Course Outcomes in COP 4225 is **acceptable** at 3.89. For all other CS courses, students rate the Adequacy of Coverage of Course Outcomes as at least **high**, with four courses, CEN 4010, CGS 1920, CGS 3092 and COP 3402, being rated as **very high** (4.50 or higher).

B. Student Outcomes

Evaluation of the level of attainment of the BS in CS Student Outcomes utilizes data obtained via several assessment mechanisms. Historically, these have been indirect measures…

- The Graduating Student (Exit) Survey,
- Course Outcomes Surveys by Students and by Instructors.

This is the first assessment cycle that incorporates data from direct sources…

- Capstone Project Assessment via CIS 4911 Senior Project presentations,
- Course-embedded Assessment by multiple-choice questions in selected required courses in the BS-CS major, MAD 3512 Theory of Algorithms, COP 3530 Data Structures, COP 4540 Database Management, and COP 4555 Principles of Programming Languages.
- Course-embedded Assessment by portfolio inspection in CGS 3092 Ethics and Social Issues in Computing.

The spring 2011 rubric for evaluation of Senior Project for assessment of Student Outcomes is provided in Appendix G. The initial version, applied in spring 2010, was improved for the fall 2010 application, and again for the spring 2011 application. The methodology, structure and scale have remained unchanged throughout the iteration. Quantitative metrics are provided in the Direct Assessment Summary of spring 2011(Appendix H) to substantiate the improvement.

The direct assessment events of fall 2010 and spring 2011 are documented in summaries provided in Appendix H, together with the data from the (pilot) assessment of the spring 2010 Senior Project presentations.

a) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.

Indicators

1.	Graduating Student Ratings	Relevance 89.2%	Attainment 89.2%	Sample: 28
2.	Course Outcomes COT 3420	Value: 77.2%	Coverage: 70.6%	Sample: 58
3.	Course Outcomes MAD 2104	Value: 87.4%	Coverage: 86.0%	Sample: 12
4.	Course Outcomes MAD 3512	Value: 78.8%	Coverage: 81.4%	Sample: 6

5. Course-Embedded Assessment MAD 3512
   Spring 2011 Event: 12 students completed a 5-question multiple choice quiz. Because of a typographical error in one question, the results from only 4 of the questions were considered.
   Criterion: At least 75% of students should score 3 of 4 (75%) or higher.
   Observation: Exactly **75%** (9) of students scored 3 of 4 or higher.
6. Senior Project Assessment
   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome a). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **1.33** Fall 2010: **3.00** Spring 2011: **3.46**

Evaluation

There appear to be issues with the delivery of COT 3420, so Indicator 2 is disregarded. Indicator 3 is disregarded because of the small sample size in relation to the number of students in MAD 2104. Inspection of the Direct Assessment Summaries (Appendix H) suggests that the Senior

Projects have so far not incorporated this curriculum component to a significant extent. The course MAD 3512 is taken close to graduation, so indicators 4 and 5, consistent with indicator 1, will be trusted. Attainment of Student Outcome a) is rated as **acceptable**.

b) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.

Indicators

1.	Graduating Student Ratings	Relevance 95.8%	Attainment 90.0%	Sample: 28
2.	Course Outcomes CDA 4101	Value: 89.8%	Coverage: 89.4%	Sample: 52
3.	Course Outcomes COP 3402	Value: 90.4%	Coverage: 92.8%	Sample: 120
4.	Course Outcomes COP 3530	Value: 91.2%	Coverage: 86.8%	Sample: 112
5.	Course Outcomes COP 4555	Value: 87.6%	Coverage: 89.2%	Sample: 63
6.	Course Outcomes COP 4540	Value: 92.6%	Coverage: 83.8%	Sample: 44
7.	Course Outcomes COP 4610	Value: 88.2%	Coverage: 81.4%	Sample: 48

8. Course-Embedded Assessment COP 4555
   Fall 2010 Event: 19 students completed a 10-question multiple choice assessment quiz.
   Criterion: 75% of students should score 7 of 10 or higher.
   Observation: **63%** of the students answered at least 7 of 10 questions correctly.
9. Course-Embedded Assessment COP 3530
   Spring 2011 Event: 25 students completed a 10-question multiple choice assessment quiz.
   Criterion: 75% of students should answer 7 or more questions correctly.
   Observation: 9 of 25 students (**36%**) answered 7 or more of 10 questions correctly.
10. Course-Embedded Assessment COP 4540
    Spring 2011 Event: 13 students completed a 5-question multiple choice assessment quiz.
    Criterion: 75% of students should answer 4 or 5 questions correctly.
    Observation: **92%** of students answered either 4 or 5 quiz questions correctly.
11. Senior Project Assessment
    Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome b). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
    Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
    Observation: Spring 2010: **4.15** Fall 2010: **3.88** Spring 2011: **4.00**

Evaluation

This Student Outcome addresses attainment in the core areas of the discipline.

- Exit Survey Indicator 1: Graduating students rate the importance and their achievement of this outcome as **very high**.
- Course Outcome indicators 2, 3, 4, 5, 6, 7: Students ascribe **very high** value to this outcome, and they rate delivery of the course outcomes as **very high**.
- Course-embedded Indicator 8, Programming Languages: **below acceptable**.
- Course-Embedded Indicator 9, Data Structures: **very low**
- Course-embedded Indicator 10, Database Management (Computer Systems): **very high**
- Senior Project Assessment Indicator 11: **acceptable**

On balance, attainment of Student Outcome b) is rated as **acceptable**. Clearly, there are concerns that must be addressed.

c) Demonstrate proficiency in problem solving and application of software engineering techniques.

Indicators

1.	Graduating Student Ratings	Relevance 93.6%	Attainment 85.8%	Sample: 28
2.	Course Outcomes CEN 4010	Value 93.0%	Coverage: 91.0%	Sample: 53
3.	Course Outcomes COP 3530	Value: 91.2%	Coverage: 86.8%	Sample: 112
4.	Course Outcomes CIS 4911	Value: 94.6%	Coverage: 85.0%	Sample: 37

5. Senior Project Assessment

   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome c). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **4.63** Fall 2010: **4.88** Spring 2011: **5.00**

Evaluation

All indicators suggest that attainment of Student Outcome c) is **excellent**.

d) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.

Indicators

1.	Graduating Student Ratings	Relevance 95.0%	Attainment 91.4%	Sample: 28
2.	Course Outcomes COP 2210	Value: 89.2%	Coverage: 87.2%	Sample: 280
3.	Course Outcomes COP 3337	Value: 90.8%	Coverage: 88.8%	Sample: 178
4.	Course Outcomes COP 3530	Value: 91.2%	Coverage: 86.8%	Sample: 112
5.	Course Outcomes COP 4338	Value: 87.2%	Coverage: 82.8%	Sample: 66

6. Senior Project Assessment

   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome d). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **4.33** Fall 2010: **4.82** Spring 2011: **4.64**

Evaluation

All indicators suggest that attainment of Student Outcome d) is **high**.

e) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.

Indicators

1.	Graduating Student Ratings	Relevance 77.2%	Attainment 83.6%	Sample: 28
2.	Course Outcomes CGS 3092	Value: 93.0%	Coverage: 93.2%	Sample: 197

3. Course Outcomes CIS 4911 Instructor Survey: "*The only course objective that was not adequately covered was Ethical Issues"*
4. Course-Embedded Assessment CGS 3092

   Fall 2010 Event: Each of 10 samples of student work was scored on an 8-point scale using a specialized rubric
   Criterion: Each topic should receive a minimum rating of 75% or a score of 6 from 8
   Observation: 8 topics were rated at 100% (8/8), 2 topics were rated at 50% (4/4).

5. Senior Project Assessment
   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome e). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **3.76** Fall 2010: **3.94** Spring 2011: **4.21**

Evaluation
On balance, attainment of Student Outcome e) is rated as **acceptable.**

f) Demonstrate the ability to work cooperatively in teams.

Indicators

1.	Graduating Student Ratings	Relevance 89.2%	Attainment 87.8%	Sample: 28
2.	Course Outcomes CEN 4010	Value 93.0%	Coverage: 91.0%	Sample: 53
3.	Course Outcomes CIS 4911	Value: 94.6%	Coverage: 85.0%	Sample: 37

4. Senior Project Assessment
   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome f). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **4.80** Fall 2010: **4.88** Spring 2011: **5.00**

Evaluation
All indicators suggest that attainment of Student Outcome f) is **excellent**.

g) Demonstrate effective communication skills.

Indicators

1.	Graduating Student Ratings	Relevance 89.2%	Attainment 87.8%	Sample: 28
2.	Course Outcomes CGS 3092	Value: 93.0%	Coverage: 93.2%	Sample: 197
3.	Course Outcomes CEN 4010	Value 93.0%	Coverage: 91.0%	Sample: 53

4. Senior Project Assessment
   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome a). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **4.63** Fall 2010: **4.88** Spring 2011: **5.00**

Evaluation
All indicators suggest that attainment of Student Outcome g) is **excellent**.

h) Have experience with contemporary environments and tools necessary for the practice of computing

Indicators

1. Graduating Student Ratings Relevance 77.2% Attainment 77.2% Sample: 28
2. Senior Project Assessment
   Event: Artifacts of all completed Senior Projects are assessed, by application of the *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, for attainment of outcome a). This event was replicated in spring 2010, fall 2010 and spring 2011 semesters.
   Criterion: Attainment should be rated at **75% or 3.75** on a 1—5 scale, or better.
   Observation: Spring 2010: **4.53** Fall 2010: **4.65** Spring 2011: **4.86**

Evaluation

All indicators suggest that attainment of Student Outcome h) is **acceptable**.

C. Program Educational Objectives

The principal means of assessing attainment of the Program Educational Objectives of the BS in Computer Science program is the Alumni Survey of Program Objectives. The alumni' responses are summarized in Table 4 (above), showing the averages of the 16 responses in the current survey cycle, and separately, all 125 responses received in earlier survey cycles. The alumni responses provide ratings of the specific facets of each objective, and overall ratings of some objectives. The Alumni Survey raw data are included in Appendix D.

Attainment of Student Outcomes enables attainment of the Program Educational Objectives, and so some Student Outcome data are again noted in this section where relevant. Additionally, the other constituent groups within the SCIS umbrella, WICS, ACM, IAB, and UPE may provide indicators of the attainment of the program objectives.

It must be noted that the number of responses to this survey, 16, is only 3 more than the 13 who responded during the preceding assessment cycle.

1. To provide our graduates with a broad-based education that will form the basis for personal growth and life-long learning.

Indicators

- Alumni Survey of Program Educational Objectives:
  Please rate how your educational experience at FIU contributed to your capacity for personal growth
  Current cycle: **82.75%** Previous cycles: **83.75%**
  Please rate how your educational experience at FIU contributed to your capacity for lifelong learning
  Current cycle: **79.75%** Previous cycles: **86.25%**

- ACM Chapter activities (Appendix H)
  Volunteer Tutoring Program, ACM Special Interest Groups, High School Programming Competition

Evaluation

It is not clear that attainment of this objective is directly enabled by specific courses in the Computer Science major. Rather, it is the collective breadth represented by the entire BS in Computer Science program that may have an enabling effect. In addition, the breadth component common to all FIU majors, the Core Curriculum and non-major elective courses, is a principal contributor to any graduates' realization of personal growth and capacity for life-long learning. Involvement with the School's student organizations is another excellent enabler of this objective, but these experiences are voluntary and are not exploited by a majority of our graduates.

While the Alumni Survey ratings of the current survey cycle are at acceptable levels, the decline relative to earlier cycles is noted.

Attainment of Program Educational Objective 1 is rated as **acceptable**.

2. To provide our graduates with a quality technical education that will equip them for productive careers in the field of Computer Science.

Indicators

- Alumni Survey of Program Educational Objectives:
  Please rate the quality of your preparation upon graduation in Computer Programming
  Current cycle: **78.25%** Previous cycles: **84.25%**
  Please rate the quality of your preparation upon graduation in Systems Development
  Current cycle: **70.25%** Previous cycles: **70.50%**
  Please rate the quality of your preparation upon graduation in Data Structures & Algorithms
  Current cycle: **86.00%** Previous cycles: **82.25%**
  Please rate the quality of your preparation upon graduation in Computer Architecture & Organization
  Current cycle: **72.00%** Previous cycles: **73.50%**

- Enabling Student Outcomes
  a) Proficiency in foundation areas – Graduating Student Rating: **89.2%**
  b) Proficiency in core CS areas – Graduating Student Rating: **90.0%**
  c) Proficiency in problem solving – Graduating Student Rating: **85.8%**
  d) Mastery of a programming language – Graduating Student Rating: **91.4%**

Evaluation

This Program Educational Objective is paramount. The ratings shown above for the current survey cycle are consistent with those reported in the 2010 assessment report:

Alumni Survey Period	**5/07 to 6/11**	**5/07 to 11/09**	**2/04 to 2/07**
# Responses	**16**	**13**	**138**
Computer Programming	78.25	77.00	84.25
Systems Development	70.25	69.25	70.50
Data Structures & Algorithms	86.00	84.50	82.25
Architecture & Organization	72.00	73.75	73.50

The ratings for preparation in the Systems Development and Computer Organization & Architecture areas have been consistently below acceptable while the ratings for Data Structures & Algorithms have consistently been high. The Computer Programming ratings have declined from high to acceptable.

The analysis of the 2010 assessment report remains pertinent but will not be repeated here. Based on the 2010 report, the SCIS has undertaken a major review of the BS in Computer Science program that has resulted in a proposal for significant changes in the program requirements. Implementation is anticipated for the academic year starting in fall 2012. Introduction into the required curriculum of units on Unix, C programming, and net-centric programming (among other program changes) should provide a major impact on the Systems Development and Architecture preparation of our graduates. As well, the common complaint of "too much Java" that possibly accounts for the decline in the Programming ratings should be mitigated by the addition of C programming as a required unit earlier in the course sequence.

Attainment of Program Educational Objective 2 is rated as **acceptable**.

3. To provide our graduates with the communication skills and social and ethical awareness requisite for the effective and responsible practice of their professions.

Indicators

- Alumni Survey of Program Educational Objectives:
  Please rate how your educational experience at FIU contributed to the development of your communication skills
  Current cycle: **73.50%** Previous cycles: **72.50%**
  Please rate how your educational experience at FIU contributed to the development of your awareness of social and ethical responsibility
  Current cycle: **81.25%** Previous cycles: **73.50%**

- Enabling Student Outcomes
  e) Understanding social and ethical concerns – Graduating Student Rating: **83.6%**
  g) Effective communication skills – Graduating Student Rating: **87.8%**

Evaluation

It is interesting that the perspective on this outcome/objective should differ in the interim from graduation to employment. While the enabling outcomes are rated as high by seniors, the alumni assign only acceptable ratings. It is reasonable to ascribe the adjustment to the real-world experiences of our graduates, but this is conjecture. This circumstance underscores the need to have continuing communication and dialog with our alumni. The upward trend in the rating of *awareness of social and ethical responsibility* is welcomed.

Attainment of Program Educational Objective 3 is rated as marginally **acceptable.**

4. To prepare students for BS level careers or continued graduate education.

Indicators

- Alumni Survey of Program Educational Objectives:
  Please rate how your educational experience at FIU contributed to your preparation for a career in computer science
  Current cycle: **79.75%** Previous cycles: **79.50%**
  Please rate how your educational experience at FIU contributed to your preparation for graduate study
  Current cycle: **72.00%** Previous cycles: **77.00%**

- ACM Chapter activities (Appendix H)
  ACM Special Interest Groups, Company Visits

Evaluation

There is a marked need for direct assessment of this objective.

Attainment of Program Educational Objective 4 is rated as **acceptable** .

V. RECOMMENDATIONS

A. Recommendations of the Subject Area Coordinators

Subject Area: Communications & Ethics (SAC: Tiana Solis)

CGS 1920:
The course name discourages students from taking the course, it implies very basic skills that they felt they already possessed. May be we should rename the course to be "Undergraduate Computer Seminar" or something else.

CGS 3092**:**
Continue the process of replacing this course with the proposed three credit hours course which will count toward the Global Learning requirement.

Subject Area: Computer Organization (SAC: Nagarajan Prabakar)

COP 3402:
Evaluate the proficiency level for each of the course outcomes in consultation with faculty who taught this course and change it appropriately

COP 4610:
Enforce the prerequisite Programming III for all students enrolled in the course (including non-CS majors). Also, the faculty needs to specify clearly about the expected C proficiency at the very first class. Furthermore, students may be given a quiz (about 10-20 short questions) in C during the first week of the term so that each students can gauge his/her ability to cope with the projects.

Subject Area: **Computer Systems** (SAC: Shu-Ching Chen)

COP 4540:
I recommend no changes to the outcome of this course. To better cover the number 5 outcomes, I suggest adding stored procedure to the syllabus

Subject Area: **Foundations** (SAC: Xudong He)

MAD 2104 & MAD 3512:
More student evaluations and instructor appraisals are needed in two Mathematics Department courses to make the assessment more meaningful and accurate

Subject Area: Programming (SAC: Tim Downey)

COP 3337:
Programming II instructors should be strongly encouraged to cover all of the objectives for Programming I, especially Stacks & Queues and the Java Collections.

COP 3530:
The low perception of the book can be attributed to the difficulty of the course. The book is one of the most popular books on the subject. We must stress to the COP3337 instructors to emphasize interfaces when possible and to be sure to cover recursion when covering stack, queues and linked lists.

COP 4338:
The course is being redesigned. The new outcomes should be posted to the CES.
The students do not have UNIX in the curriculum; hopefully, the redesigned course will realize this and not expect students to have experience in it.

Subject Area: **Software Engineering** (SAC: Peter Clarke)

CEN4010:
There is a need to have students take a programming course that contains web-based programming and learning technologies such as Tomcat, Apache Server, PHP/JSP/ASP.

CIS 4911:
1. If the senior project course is to be taken seriously then SCIS must find a way to get faculty involved in the course and the faculty must dedicate the time and effort in order for the course to be a success. The course cannot be treated solely as a way for undergraduate students to work on research projects or to do "on the side" projects for faculty members.
2. There area of ethical issues needs to be adequately covered in a prerequisite course. Students are creating software artifacts and must know how to acknowledge other people's work being used, and how to write the appropriate licenses to protect their own work. In addition, it is important for them to have some understanding of the privacy and security issues when they are writing software in some domains, e.g., healthcare.
3. Students need additional practice in both written and verbal communication

B. Recommendations of the Assessments Coordinator

AC-01: As the Foundations Area Coordinator indicated, the number of responses to the MAD 2104 and MAD 3512 Course Outcomes Surveys are too low to permit meaningful evaluation. It is noted elsewhere that student participation is voluntary, and outside of regular class hours. The feasibility of doing in-class evaluations should be considered. Failing that, other assessment means must be employed for the MAD 2104 and MAD 3512 courses on a regular schedule.

AC-02: The average ratings for the Value and Coverage of the Course Outcomes are 4.51 and 4.40 respectively (Please refer to Table 1). In fact, only 3 of 44 ratings fall below 4.00. It seems pointless, and possibly counter-productive, to maintain the acceptability level at 3.75. SCIS should consider raising the minimum acceptable rating for both Value and Coverage of Course Outcomes to at least 4.00.

AC-03: The Course Outcomes ratings for COT 3420 are 3.86 (Value) and 3.53 (Coverage), the latter being well below the acceptability threshold of 3.75. It is clear that students believe that the outcomes of this course are not adequately covered. Further, the Subject Area Coordinator's report for previous assessment cycles clearly indicate that the Course Outcomes are not followed consistently by the various instructors of COT 3420. The content and delivery of this course must be clearly specified and followed by all instructors. It would not be untimely to consider alternative implementations of COT 3420 to include knowledge units from applied logic areas, for example artificial intelligence, knowledge-based reasoning, robotics, game playing, etc.

AC-04: It is challenging to perform meaningful assessment of Student Outcome a) ***Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.*** First, SCIS does not deliver ***mathematics*** in its courses. Second, ***logic*** is included in ***discrete structures*** and other courses. This recommendation is to restate Student Outcome a) as ***Demonstrate proficiency in foundation areas of Computer Science including discrete structures, formal languages and automata.***

AC-05: SCIS has explicitly incorporated direct measures into its assessment processes. The broad generality of the Course Outcomes of many of the BS-CS required courses does not lend for focused outcome assessment via course-embedded assessment strategies. To facilitate more focused assessment and evaluation, finer granularity of outcomes is desirable. It is proposed to refine the Course Outcomes of all required and elective courses of the BS in CS into Learning Outcomes of sufficient granularity to support course-embedded assessment. Some examples of the refinement sought are provided as Appendix K to this report.

AC-06: There are anomalies in the assessment indicators of Student Outcome b). In particular, the indirect indicators of COP 3530 are very high while the course-embedded indicators are very low. It seems necessary to implement processes for course-embedded assessment that **a)** ensure instructor-neutral assessment instruments, **b)** promote consistency across repeated applications of an assessment procedure, and **c)** specify how to respond to the assessment indicators.

AC-07: Very little new data is available from the Alumni Survey, 3 over the current assessment cycle. The usefulness and administration of the Alumni Survey should be evaluated.

AC-08: It is impossible to conduct meaningful assessment of the Program Educational Objectives without collecting data on the accomplishments of our graduates in the workplace or in academia. This is also implied in a statement of concern in our most recent statement of accreditation. It is essential that SCIS implement and execute processes of gathering job placement and graduate attainment data.

AC-09: Adoption of direct assessment strategies may entail consistent course administration activities more so than hitherto obtained. For example, maintaining a pool of multiple-choice questions for course–embedded assessment, application of portfolio evaluation rubrics and evaluation of assessment data on an on-going basis. SCIS may consider instituting course coordination by designated faculty members, and re-evaluate the efficacy of subject-area coordination as it is currently defined.

AC-10: There seems to be a need to improve the delivery of CIS 4911 Senior Project. The Software Engineering SAC has made some recommendations for doing so. A majority of students in CIS 4911 have had little prior project management experience, and are sometimes unfamiliar with the application domain of their project. SCIS should consider incorporating an instructional component into this important course to include the following: 1) domain-specific knowledge and testing strategies, 2) project management, 3) proprietorship-related issues,4) technical presentation critique.

VI. CONCLUSION

The BS in Computer Science program continues to deliver high quality preparation for entry into the computing work-force, or admission to graduate programs in computing. The delivery of its required coursework continues to receive very high ratings from students as expressed in the Surveys of Course Outcomes (4.40/5, 88%, Table 1). Evaluations of attainment of its Student Outcomes (90.2%, Table 3) and Program Educational Objectives (77.75%, Table 4) uniformly meet or exceed the minimum acceptability criteria.

In continuing to strive to ensure students' educational experiences are relevant to the reality of the work-force they enter, SCIS has introduced a capstone experience in the Senior Project course, and has redesigned the Software Development track to include instruction in software testing techniques. The program assessment processes are continually being strengthened by introduction and fine-tuning of more direct assessment strategies.

The ACM and UPE chapters have seen increased membership, and have expanded their activities to include several Special Interest Groups and a vigorous Volunteer Tutoring Program.

This is a landmark report is two ways. In future, a similar report will be issued biennially rather than annually as has been the practice since SCIS initiated formal program assessment in 2003. Of even greater significance, arising out of the assessment process, the School has completed a major revision of the program's outcomes and curriculum. The revised program, which is expected to be operational by fall 2012, will strengthen students' technical preparation by bringing knowledge units in computer systems, net-centric computing, and global learning experiences into the required curriculum.

VII. APPENDICES

A. BS in CS *Program Educational Objectives* and *Student Outcomes*
B. The SCIS BS in CS *Assessment Plan* and *Assessment Mechanisms & Procedures*
C. Subject Area Coordinator Reports
D. Exit (Graduating Student) Survey and Alumni Survey raw data
E. Direct Assessment Summaries, Fall 2010 & Spring 2011
F. Course-Embedded Assessment Data, Fall 2010 & Spring 2011
G. *Senior Project Assessment of Student Outcomes of the BS in Computer Science* rubric, check-list and score-grid.
H. Student Organization reports (ACM, UPE)
I. Learning Outcomes examples

APPENDIX A: BS in CS *Program Educational Objectives* and *Student Outcomes*

BS in Computer Science Program Educational Objectives

1. To provide our graduates with a broad-based education that will form the basis for personal growth and life-long learning.
2. To provide our graduates with a quality technical education that will equip them for productive careers in the field of Computer Science.
3. To provide our graduates with the communication skills and social and ethical awareness requisite for the effective and responsible practice of their professions.
4. To prepare students for BS level careers or continued graduate education.

BS in Computer Science Student Outcomes

To complete the program of study for the BS in Computer Science, every student will

a) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.
b) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.
c) Demonstrate proficiency in problem solving and application of software engineering techniques.
d) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.
e) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.
f) Demonstrate the ability to work cooperatively in teams.
g) Demonstrate effective communication skills.
h) Have experience with contemporary environments and tools necessary for the practice of computing.

As currently stated http://www.cis.fiu.edu/programs/undergrad/cs/assessment/bsoutcomes.php :

BS-CS Program Objectives

1. To provide our graduates with a broad-based education that will form the basis for personal growth and life-long learning.
2. To provide our graduates with a quality technical education that will equip them for productive careers in the field of Computer Science.
3. To provide our graduates with the communication skills and social and ethical awareness requisite for the effective and responsible practice of their professions.
4. To prepare students for BS level careers or continued graduate education.
5. To maintain a diverse student population and actively promote an environment in which students from all groups, including the traditionally under-represented, may successfully pursue the study of Computer Science.
6. To maintain a qualified and dedicated faculty who actively pursue excellence in teaching.

Proposed Modification: *Remove items 5 and 6.*

BS in Computer Science Program Educational Objectives

1. To provide our graduates with a broad-based education that will form the basis for personal growth and life-long learning.
2. To provide our graduates with a quality technical education that will equip them for productive careers in the field of Computer Science.
3. To provide our graduates with the communication skills and social and ethical awareness requisite for the effective and responsible practice of their professions.
4. To prepare students for BS level careers or continued graduate education.

Note that items 5 and 6 of the existing BS-CS Program Objectives are already expressed as part of the SCIS Mission on the SCIS web page http://www.cis.fiu.edu/about.php :

High Quality Undergraduate Program

To offer an undergraduate degree program leading to the BS in Computer Science which includes high quality teaching by properly trained and experienced faculty.

…

Effective quality teaching is an essential part of the responsibilities of all faculty in the School, and special efforts are made to find ways of increasing that quality and effectiveness.

…

Because students from minority and other disadvantaged and/or underrepresented groups have been historically underrepresented in technological areas, the School has a special responsibility to recruit, encourage, and support undergraduate major students from those groups.

As currently stated http://www.cis.fiu.edu/programs/undergrad/cs/assessment/bsoutcomes.php

BS-CS Student Outcomes

To complete the program of study for the BS in Computer Science, every student will

a) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.
b) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.
c) Demonstrate proficiency in problem solving and application of software engineering techniques.
d) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.
e) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.
f) Demonstrate the ability to work cooperatively in teams.
g) Demonstrate effective communication skills.
h) Demonstrate understanding of the scientific method.
i) Demonstrate familiarity with fundamental ideas and issues in the arts, humanities and social sciences.
j) Have experience with contemporary environments and tools necessary for the practice of computing.
k) Be successful in applying for computer science related entry-level positions in business, industry or government.[*Computer Science track graduates*]
l) Be successful in gaining admission to graduate programs in Computer Science.

Proposed Modifications:
Remove outcomes h), i), k) and l). Retain outcomes a) through g). Renumber outcome j) to h).

BS-CS Student Outcomes

To complete the program of study for the BS in Computer Science, every student will

i) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.
j) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.
k) Demonstrate proficiency in problem solving and application of software engineering techniques.
l) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.
m) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.
n) Demonstrate the ability to work cooperatively in teams.
o) Demonstrate effective communication skills.
p) Have experience with contemporary environments and tools necessary for the practice of computing.

Justification

- *Student Outcomes h) and i) of the existing BS-CS Student Outcomes are enabled by other components of the students' educational programs that are not delivered by SCIS.*
- *Student Outcomes k) and l) of the existing BS-CS Student Outcomes are already expressed as part of the SCIS Mission on the SCIS web page* http://www.cis.fiu.edu/about.php :

High Quality Undergraduate Program

To offer an undergraduate degree program leading to the BS in Computer Science which includes high quality teaching by properly trained and experienced faculty. This program prepares students for either continued graduate education or for BS level careers in business, industry, or government….

APPENDIX B: BS in CS *Assessment Plan* and *Assessment Mechanisms & Procedures*

SCHOOL OF COMPUTING AND INFORMATION SCIENCES

ASSESSMENT PLAN
of the
Bachelor of Science in Computer Science

I. INTRODUCTION

The document, Assessment Mechanisms and Procedures, of the School of Computing and Information Sciences (SCIS), describes the means by which the School conducts the annual assessment of its BS in Computer Science program. The instruments employed for assessment, and the SCIS administrative structure for performing the assessment are described in that document. These means include

- Survey Instruments
    1. Course Outcomes Survey by Students
    2. Course Outcomes Survey by Instructors
    3. Survey of Graduating Students
    4. Survey of Alumni
- Recommendations from constituents
    1. Industry Advisory Board (IAB)
    2. Women in Engineering and Computer Science (WIECS)
    3. ACM Student Chapter
- Direct Measures
    1. Senior Project Assessment
    2. Course-Embedded Assessment

The administrative structure for conducting the assessment comprises

- The Undergraduate Program Director (UPD)
- The Assessments Coordinator (AC)
- The Subject Area Coordinators (SACs)

The assessment procedures are performed by the SCIS Subject Area Coordinators and the SCIS Assessments Coordinator. Their findings are reported to the SCIS Undergraduate Committee for evaluation, resulting in a set of recommendations to the SCIS faculty.

This document, the SCIS Assessment Plan, defines the implementation of the entire assessment cycle. It specifies the roles of all participants in the process, and sets out a timetable for execution of those roles.

II. PARTICIPANTS

1) The Undergraduate Program Director (UPD)

The Undergraduate Program Director is appointed by the Dean of the School of Computing and Information Systems. The UPD bears overall responsibility for the administration of all SCIS undergraduate programs.

The role of the UPD relevant to the assessment process is

- To designate the chair of the SCIS Undergraduate Committee (below)
- To ensure that the assessment timetable is followed and that the procedures are otherwise executed as set forth in this document and in the Assessments Mechanisms and Procedures Document
- To document and implement program adjustments arising from the annual assessment process that are approved by the SCIS faculty and, if necessary College and University Curriculum Committees.

2) The Subject Area Coordinators (SACs)

The Subject Area Coordinators may be appointed by the UPD or elected by the SCIS faculty. Each SAC bears responsibility for a group of courses in the BS in Computer Science curriculum:

Foundations Subject Area courses:
MAD 2104, MAD 3512, COT 3420, COP 4555
List 2 electives: MAD 3305, MAD 3401, MAD 4203, MHF 4302

Programming Subject Area courses:
COP 2210, COP 3337, COP 3530, COP 4338

Software Engineering Subject Area courses:
CEN 4010, CEN 4012, CEN 4021, CEN 4023, CIS 4911

Computer Systems Area courses:
COP 3402, CDA 4101, COP 4540, COP 4610
List 1 electives: CAP 4710, CAP 4770, CDA 4400, CNT 4403,
CNT 451, COP 4225, COP 4226

Communications & Ethics Area courses:
CGS 1920, CGS 3092, COM 3011, ENC 3211

Calculus and Physics Area courses:
MAC 3311, MAC 3312, PHY 2048(L), PHY 2049(L), STA 3033

The role of a Subject Area Coordinator is:

- To maintain a common syllabus for each SCIS course in their area.
- To maintain the instruments and rubrics for course-embedded assessment in their area
- To liaise with the academic unit teaching a non-SCIS course that is a required or elective course in the BS in CS program.
- To interpret the data from the Student and Instructor Course Outcomes surveys for each course in their area.
- To prepare an annual report presenting the findings from the course surveys, and to make recommendations based on these findings.

3) The Assessments Coordinator (AC)

The Assessments Coordinator is appointed by the SCIS Dean. The role of the AC is:

- To interpret the data from the Survey of Graduating Students, Senior Project assessment, and Alumni survey.
- To prepare the SCIS annual assessment report. The report presents the data from these assessment mechanisms and resulting findings and recommendations, and summarizes the recommendations from the several SAC annual reports.
- To monitor the BS in CS program for compliance with the ABET accreditation criteria.
- To prepare the ABET accreditation self-study report, and program documentation as may be required by ABET.

The Assessments Coordinator should not simultaneously be a Subject Area Coordinator, except for the Calculus and Physics area (liaison).

4) The Undergraduate Committee (UGC)

The Undergraduate Committee may be appointed by the SCIS Dean or elected by the SCIS faculty. The UGC Chair convenes and conducts all UGC meetings as necessary. The Undergraduate Program Director and Assessments Coordinator are ex-officio members of the Undergraduate Committee.

The UGC has the responsibility of considering proposed changes to the existing SCIS undergraduate courses and programs, and of making recommendations, based on these considerations, to the full SCIS faculty.

The role of the UGC in the assessment process specifically, is to consider the AC's annual assessment report. Each AC or SAC recommendation contained in the annual report is evaluated by the UGC. Where helpful, the UGC may require further input or clarification from the author (AC or SAC) of a recommendation. At the conclusion of their deliberations, the UGC chair prepares a summary of recommendations for presentation to the SCIS faculty. In the summary:

- The UGC may endorse an AC or SAC recommendation for adoption by the SCIS faculty.
- The UGC may endorse an AC or SAC recommendation and propose to the SCIS faculty a means of enacting the recommendation.
- The UGC may decline to act on a recommendation, setting forth reasons for its decision.
- The UGC may author its own recommendations to the SCIS faculty.

5) The SCIS Faculty

The SCIS faculty, collectively, has sole responsibility for promulgating and modifying its academic programs. The SCIS faculty approves or rejects any recommendations for adjustments to the BS in Computer Science program. Adoption of SCIS approved program adjustments may be subject to final approval of College and University Curriculum Committees.

III. SCHEDULE

1) Surveys

The schedule for administering Course Outcomes, Graduating Students and Alumni surveys is set out in the SCIS Assessment and Mechanisms document. All surveys are carried out on-line. The SCIS Director for IT and Business Relations has the responsibility of ensuring that the data from any survey is available within one month of conclusion of the survey.

2) Direct Measures Assessment

Senior Projects are presented at the end of every semester. The resulting assessment data are collected by the Senior Project coordinator and are available by the start of the following semester. Data from the course-embedded assessments are prepared by the SAC's and are made available by the start of the next semester.

3) Subject Area Coordinator Annual Reports

The SAC annual reports cover the Spring, Summer, and Fall semesters of one calendar year. The SAC annual reports are made available to the Assessments Coordinator by the end of January of the following year.

4) Recommendations from Constituents

Recommendations from IAB, WEICS, ACM Chapter, or other constituent group are provided to the assessments Coordinator no later than the end of January of each year.

5) Assessment Coordinator Annual Report

The AC annual report incorporates data and recommendations from all of the sources listed above. The report covers the period of one calendar year and is made available to the Undergraduate Committee by the end of February of the following year.

6) Undergraduate Committee Summary of Recommendations

UGC meetings to consider the annual assessment report are conducted during the months of February, March and April. UGC concludes all deliberations, and the UGC summary of recommendations is made available to the SCIS faculty, no later than two weeks prior to the end of the Spring semester.

The UGC chair should prioritize recommendations for adjustments to the BS in CS program that require further approval by the College Curriculum Committee. The SCIS Dean and/or UPD should expedite SCIS faculty consideration of such recommendations, bearing in mind the deadlines of the College Curriculum Committee, and with a view to implementation at the start of the next academic year.

7) SCIS Faculty Assessment Meeting

The SCIS Dean convenes a meeting of the SCIS faculty to consider the UGC recommendations prior to the end of the Spring semester, but no sooner than one week following receipt of the UGC summary of recommendations. Should matters be left over from this meeting, such matters should be addressed during the first meeting of the full SCIS faculty in the following Fall semester.

IV. ENACTMENT

- UGC recommendations not requiring faculty approval must be enacted by the responsible entity, SAC or UPD, immediately and reported to the next meeting of the full SCIS faculty.
- UGC recommendations approved by the SCIS faculty during the Spring meeting, and not requiring further approval by the College, must be enacted by the UPD as soon as practicable, and by the start of the following Summer semester if at all possible.
- Recommendations approved by the SCIS faculty during the Fall meeting, and not requiring further approval by the College, must be enacted by the UPD as soon as practicable during the Fall semester.
- Recommendations for BS in CS program adjustments approved by the SCIS faculty, and subsequently approved by the College and/or University Committees, must be enacted at the earliest possible date following approval by the highest Committee.

The Undergraduate Program Director has overall responsibility for enactment of all program adjustments resulting from the annual assessment process. The UPD is charged with documentation and publication of program adjustments.

Revised: November 16, 2010

SCHOOL OF COMPUTING AND INFORMATION SCIENCES
ASSESSMENT MECHANISMS AND PROCEDURES
of the
Bachelor of Science in Computer Science

(Revised November 16, 2010)

I. INTRODUCTION

The School of Computer and Information Sciences (SCIS) at Florida International University uses several mechanisms to assess the extent to which its undergraduate program outcomes and objectives are being met. Further, the School has defined procedures to evaluate the assessment results and to identify ways to improve its curriculum based on the assessment results, as deemed necessary and appropriate by its faculty.

SCIS currently uses four survey instruments:

1. Course Outcomes Survey by Students
2. Course Outcomes Survey by Instructors
3. Survey of Graduating Students
4. Survey of Alumni

Direct measure of attainment of the program outcomes is performed by assessment of student performance in the Senior Project course (Capstone course) taken in the students' final semester.

In addition to the data from the survey instruments and Senior Project assessment, SCIS seeks recommendations from other constituents of the BS in CS program, including the Industrial Advisory Board, Women in Engineering and Computer Science group, and the ACM student chapter.

II. ADMINISTRATIVE STRUCTURE

To administer and evaluate these assessments, SCIS has created an administrative structure that includes:

- the Undergraduate Program Director (UPD),
- the Assessments Coordinator (AC),
- the Subject Area Coordinators (SACs)

The Undergraduate Program Director is appointed by Dean of the School.
The Assessments Coordinator and the Subject Area Coordinators are appointed by the Undergraduate Program Director.

Each course in the BS in Computer Science program falls under one of five subject areas, each with its own SAC: Programming, Software Engineering, Computer Systems, Foundations, and Communication & Ethics. Each Subject Area Coordinator is responsible for writing an annual report detailing recommendations for modifications pertaining to all courses in their respective subject area.

The Assessments Coordinator is responsible for writing an annual report summarizing the recommendations of the SACs, and recommendations received from the other program constituents. The AC's report is submitted to the SCIS Undergraduate Committee for consideration.

On consideration of the AC and SAC reports, the SCIS Undergraduate Committee may subsequently make recommendations to the full SCIS faculty. Recommendations adopted by the SCIS faculty are implemented via the normal academic procedures of the university.

The Undergraduate Program Director bears the overall responsibility for assessing the undergraduate programs of the School as well as ascertaining that defined procedures are followed in a timely fashion.

III. ASSESSMENT INSTRUMENTS AND PROCEDURES

As indicated earlier, SCIS utilizes data from the survey instruments and Senior Project evaluation, and recommendations from its constituent groups, to assess whether the program outcomes and objectives of the BS in Computer Science program are being met. The details of these assessment mechanisms, and their application, are described below.

A. SURVEY INSTRUMENTS:

SCIS currently uses four survey instruments. All surveys are conducted online. The Associate Director for Computing Technologies is responsible for ensuring that meaningful statistics for each survey are available within a month after the semester concludes.

The student and instructor Course Outcomes Survey statistics are analyzed and reported in the annual reports of the Subject Area Coordinators.

The Graduating Students and Alumni survey statistics are analyzed and reported in the annual report of the Assessments Coordinator.

1. Course Outcomes Survey by Students

This survey is undertaken during the final two weeks of every semester.

Students of every class offered during the semester are asked to rate each course outcome from two perspectives by indicating the extent to which they agree or disagree with two assertions about that outcome:

- *I believe that this is a valuable outcome for this course*
- *The subject matter of this outcome was covered adequately in class*

Responses are given on a scale of 1 to 5 with 5 indicating strong agreement with the assertion, and 1 indicating strong disagreement. The students' responses from both perspectives, *value of outcome* and *adequacy of coverage*. are averaged across the class, individually for each outcome, and cumulatively for all outcomes

2. Course Outcomes Survey by Instructors

This survey is undertaken at the conclusion of every semester.

For each class offered during any semester, the instructor of the class completes a grid showing how course assignments and tests relate to the individual course outcomes. The instructor rates each course outcome from two perspectives:

- The *appropriateness* of the outcome is rated as one of *essential*. *appropriate*, or *inappropriate*.
- The in-class coverage of the outcome is rated as one of *extensively*, *adequately*, *not enough*, or *not at all*.

The instructor also provides ratings of the *relevance* and *student mastery* of the *course prerequisite outcomes*, and may choose to provide recommendations for additional prerequisite outcomes.

3. Survey of Graduating Students (Program Outcomes)

This survey is undertaken every semester, beginning during the final two weeks of the semester.

The graduating student is asked to rate each of the BS in Computer Science (curricular) Program Outcomes, *a* through *j*, from 2 perspectives.

- The graduating student indicates the extent to which they agree or disagree with the following assertion:
- *This program outcome has been met for me personally*
- The graduating student indicates how meaningful they consider the outcome to be:
- *How meaningful do you consider this outcome to be for you personally?*

Program outcomes *k* and *l* relate to the success of the graduating student in finding CS-related employment, and admission to graduate school respectively. For each of these 2 outcomes, *k* and *l*, the student indicates how successful they have been, and how their CS education has contributed to that success.

Responses to all questions are given on a scale of 0 through 5, with 0 being least favorable, and 5 being most favorable, and are averaged across all students completing the survey.

4. Survey of Alumni (Program Objectives)

This survey is undertaken by graduates of the BS in Computer Science program, and is conducted every three years.

Alumni completing this survey are asked to provide ratings of the several facets of the BS in Computer Science Program Objectives under four broad areas:

- quality of Educational Experience (6 facets)
- quality of Faculty and Instruction (4 facets)
- quality of preparation in the Curricular Areas (4 facets)
- promotion of Diversity and Healthy Environment (4 facets)

Each facet is rated on a scale of 0 (Unsatisfactory) through 4 (Excellent). The ratings are averaged for each individual facet (18), for each area (4), and cumulatively across all facets.

B. RECOMMENDATIONS:

Periodically, we seek out recommendations for curricular changes from diverse bodies and interest groups. In all cases, curriculum modifications based on these recommendations will be included in the annual report submitted by the AC to the School's curriculum committee.

1) Industry Advisory Board (IAB):

The IAB of the School is expected to meet once a year to discuss among other things, how we can prepare our students better to face the current challenges in the field. The Dean of the School, the UPD, and the AC will review these formal and informal recommendations of the Board.

2) Women in Engineering and Computer Science (WIECS) group:

The WIECS women's forum meets occasionally throughout the year under the leadership of a faculty member of the School. The problems faced by women in science areas of endeavor are unique, and we take the recommendations of this group to address their concerns about our curriculum and how can we assist them to perform better and attract more women into our program. The AC and the UPD review the recommendations of the group on an annual basis.

3) ACM Student Chapter:

The members of our ACM Student Chapter meet periodically throughout the year. Recommendations made by this group through their faculty advisor are reviewed by the AC and the UPD on an annual basis.

C. DIRECT MEASURES

1. Senior Project Assessment

For the purpose of assessing the BS in CS Program Outcomes via the Senior Project, the UPD, in consultation with the faculty, constitutes an evaluation team(s) of at least 3 persons to include

1. The Senior Project course coordinator/instructor (faculty),
2. A second faculty member not associated with the project,
3. A non-faculty representative from the SCIS Industry Advisory Board, or person with similar experience nominated by the Board.

Several such teams may be constituted, based on the number of student projects to be evaluated.

The evaluation team observes the students' oral presentations and/or demonstrations of their project. The evaluation team has access to all artifacts produced by the student team to satisfy the requirements of the Senior Project course.

The members of the evaluation team complete a suitable instrument to indicate their assessment of the extent to which the students' work demonstrates attainment of the BS in Computer Science Program Outcomes. The instrument includes rubrics to guide their evaluations. The instrument and included rubrics must be published.

The completed evaluation instruments, together with the project artifacts, become components of the annual assessment process, and must be maintained until at least the following ABET accreditation site visit.

2. Course-Embedded Assessment

In addition to assessment via the Senior Project, the Undergraduate Program Director and Assessments Coordinator, in consultation with the relevant Subject Area Coordinators, may designate courses for sampling of student work (exams and/or projects), for the purpose of assessing attainment of Student Outcomes. The particular courses to be sampled may be determined from semester to semester. The Subject Area Coordinators will maintain suitable sampling mechanisms and rubrics for assessment of Student Outcomes via the courses in their areas.

IV. IMPLEMENTING CURRICULUM CHANGES:

The Assessment Coordinator's annual written report is submitted to the SCIS Undergraduate Committee by the end of February of each year. The report includes recommended curriculum modifications based on all of the assessment mechanisms. The SCIS Undergraduate Committee completes all internal deliberations in the School by the end of the Spring semester so that the faculty approved changes in our curriculum can be submitted to the College Curriculum Committee's first meeting in the Fall semester. The University approved curriculum modifications are implemented no later than in the subsequent Fall semester.

APPENDIX C: Subject Area Coordinator Reports

Subject Area Report for 2010

Subject Area: Communications & Ethics (Reported by Tiana Solis)

CGS1920 Introduction to Computing
CGS 3092 Professional Ethics and Social Issues in Computer Science
COM 3011 Business and Professional Communication
ENC 3211 Report and Technical Writing

COM 3011 and ENC 3211 are taught by other instructional units and consequently are not subject to the School's assessment mechanisms. The Subject Area Coordinator's report thus addresses CGS 1920 and CGS 3092 only.

CGS 1920

All objectives were covered on an assignment or guest lectures or research activities.
All objective were considered essential.
Most objectives were covered extensively except for undergraduate research opportunities.

Recommendations: *The course name discourages students from taking the course, it implies very basic skills that they felt they already possessed. May be we should rename the course to be "Undergraduate Computer Seminar" or something else.*

CGS 3092

All objectives were covered on an assignment or in an in class discussion.
All objective were considered essential.
Most objectives were covered extensively except for team problem solving.
Most prerequisite objectives currently listed include specific programming skills that were considered incidental. Making ENC3211 Technical writing a pre-requisite for this course significantly improved the outcome of most objectives.

Recommendations: *Continue the process of replacing this course with the proposed three credit hours course which will count toward the Global Learning requirement.*

Computer Organization: Area Coordinator Report

Nagarajan Prabakar

October 27, 2011

1. **Introduction:**

The Computer Organization area consists of the following three courses: CDA-3103 (Fundamentals of Computer Systems), CDA-4101 (Structured Computer Organization) , and COP-4610 (Operating Systems Principles). The assessment report given below for each of these courses is based on student responses about the course outcomes and the faculty course appraisals.

2. CDA-3103: Fundamentals of Computer Systems

The following table shows a summary of the course assessment evaluations:

	No. of Student Responses	Value of Outcome	Coverage Adequacy	Professor
Spring '10	38	4.52	4.57	Pestaina
Summer '10	13	4.72	4.77	Pestaina
Fall '10	26	4.39	4.57	Pestaina
Spring '11	43	4.53	4.70	Pestaina
	=======	=======	=======	
	120	4.52	4.64	

For all five outcomes of the course, most of the students (more than 80%) agree either strongly or moderately. There is no significant concern expressed in the Students Suggestions section. Only one faculty appraisal suggests to re-consider the proficiency level of each of the outcomes.

Recommendation: Evaluate the proficiency level for each of the course outcomes in consultation with faculty who taught this course and change it appropriately.

3. CDA-4101: Structured Computer Organization

The following table shows a summary of the course assessment evaluations:

	No. of Student Responses	Value of Outcome	Coverage Adequacy	Professor
Spring '10	22	4.46	4.33	Barton
Fall '10	19	4.44	4.51	Barton
Spring '11	11	4.64	4.67	Barton
	=======	=======	=======	
	52	4.49	4.47	

For all five outcomes of the course, most of the students (more than 80%) agree either strongly or moderately. There is no significant concern expressed by the students or faculty.

Recommendation: No change is needed on the course outcomes or syllabus.

4. **COP-4610: Operating Systems Principles**

The following table shows a summary of the course assessment evaluations:

	No. of Student Responses	Value of Outcome	Coverage Adequacy	Professor
Spring '10	16	4.34	4.23	Zhao
Summer '10	16	4.71	4.63	Barton
Fall '10	10	3.98	3.82	Zhao
Spring '11	6	4.53	2.57	Wei
	=======	=======	=======	
	48	4.41	4.07	

For all five outcomes of the course, most of the students (more than 75%) agree either strongly or moderately for all terms with the exception of Spring'11 which had only six student responses. Students suggested consistently about their inadequate preparation in C for this course. Non-CS majors take only one introductory level C programming course and they are unable to complete projects with system calls,
whereas CS major students who complete COP-4338 Programming III before they enroll in COP-4610, have adequate C proficiency to complete projects. Also, the lack of adequate of prerequisite skills among a subset of students is cited in the faculty course appraisals. Jinpeng Wei taught this course first time in Spring'11 and the lack of students' C proficiency together might be the reason for the low assessment score in Spring'11.

Recommendation: Enforce the prerequisite Programming III for all students enrolled in the course (including non-CS majors). Also, the faculty needs to specify clearly about the expected C proficiency at the very first class. Furthermore, students may be given a quiz (about 10-20 short questions) in C during the first week of the term so that each students can gauge his/her ability to cope with the projects.

Subject Area: Computer Systems (Reported by Shu-Ching Chen)
COP 4540 Database Management
CAP 4770 Introduction to Data Mining
COP 4604 Advanced UNIX Programming

COP 4540 Database Management

- *Appraisal and Course Evaluation Reports Status: This course was taught three times by two instructors during the past year. The instructors have submitted all of the course appraisals for all the sessions. The student evaluation for all of the three sessions is available in the system.*
- *Summary of Assessment: This course has seven outcomes, all of which has been indicated by the instructors as either essential or appropriate.*
- *Embedded assessment questions to cover seven are done in Spring 2011. The assessment results show that the students have good course outcome.*
- *Recommendation: I recommend no changes to the outcome of this course. To better cover the number 5 outcomes, I suggest adding stored procedure to the syllabus.*

CAP 4770 Introduction to Data Mining

- *Appraisal and Course Evaluation Reports Status: This course was taught one time by one instructor during the past year. The instructors have submitted the course appraisal for all this session. Also, the student evaluation for this session is available in the system.*
- *Summary of Assessment: This course has six outcomes, all of which has been indicated by the instructors as either essential or very appropriate.*
- *Recommendation: I recommend no changes to the outcome of this course.*

COP 4225 Advanced UNIX Programming

- *Appraisal and Course Evaluation Reports Status: This course was taught two times by two instructors during the past year. The instructor has submitted the course appraisal for all the two sessions. The student evaluation for all of the two sessions is available in the system.*
- *Summary of Assessment: This course has six outcomes, all indicated by the two instructors as essential.*
- *Recommendation: I recommend no changes to the outcome of this course.*

Assessment of 2010 Foundations Courses

Xudong He

September 18, 2011

1 Introduction

The Foundations courses are COT 3420 (Logic for Computer Science), COP 4555 (Principles of Programming Languages), MAD 2104 (Discrete Mathematics), MAD 3512 (Theory of Algorithms), and the math electives. For the first time, Mathematics Department has done assessments for their courses; however the students' responses are very low. There were a total of 12 student responses from two sections of MAD-2104, and a total of 6 student responses from three sections of MAD-3812. Furthermore, there are no instructor appraisals from these two Math Department courses.

2 COT 3420 Logic for Computer Science

Alex Pelin taught a section of COT 3420 in Summer 2010, another in Fall 2010. Christine Lisetti taught a section in Spring 2010.

The following table shows a summary of the student evaluations:

	# Responding	Outcome Value	Coverage Adequacy
Spring 10	20	4.05	3.11
Summer 10	11	3.84	3.59
Fall 10	10	3.83	4.28
	=======	=======	=======
Year 2010	41	3.92	3.79

Overall the evaluations went down a bit compared to last years'. The drop seems to be most significant in Spring 10 mainly due to Christine's illness, which was reflected in students' negative comments about missing classes and substituting professors. Some students suggested including homework and quizzes in the class.

In the appraisals, Alex found the students' preparation deficient. Alex observed the continued decline of student quality and suggested to tighten the admission requirements.
Christine was ill and did not give appraisal in Spring 2010.

3 COP 4555 Principles of Programming Languages

In 2010, Geoff Smith taught one section of COP 4555 in Spring 2010 and another in Fall 2010. Students submitted 22 evaluations in Spring and 21 in Fall, again reflecting a much improved response rate with the new "netbook" procedure.

The following table shows a summary of the student evaluations:

	# Responding	Outcome Value	Coverage Adequacy
Spring 10	22	4.44	4.49
Fall 10	21	4.45	4.48
	=======	=======	=======
Year 2010	43	4.45	4.48

The student evaluations are positive. A few students questioned whether a standard functional language such as ML or Lisp should be used instead of F#.

In his Fall 2010 appraisals, Geoff was concerned about the motivation and effort that students put into the course. "Typically about half of the students showed up at each class meeting." With regard to the group homework assignments, significant copying between the groups existed. The lack of effort manifested itself in disappointing results on the final exam.

4 MAD-2104 Discrete Mathematics

There are student evaluations for one section of MAD-2104 taught in Spring 2010 and another in Fall 2010.Since there are no instructor appraisals, it is unknown who taught these courses.

The following table shows a summary of the student evaluations:

	# Responding	Outcome Value	Coverage Adequacy
Spring 10	5	4.75	4.75
Fall 10	7	4.10	3.97
	=======	=======	=======
Year 2010	12	4.33	4.26

The submitted student evaluations were generally positive. The average score for the valuation of the course outcomes was 4.33 out of 5, and for the adequacy of coverage was 4.26 out of 5. But the low participation rate makes firm conclusions unwarranted.

Some student suggestions include the re-organization of materials to avoid putting all hard materials at the very beginning; and the need of more time to cover recursion.

5 MAD-3512 Introduction to Theory of Algorithms

There are student evaluations for one section of MAD-3512 taught in Spring 2010, one section in Summer 2010, and another in Fall 2010.Since there are no instructor appraisals, it is unknown who taught these courses.

The following table shows a summary of the student evaluations:

	# Responding	Outcome Value	Coverage Adequacy
Spring 10	3	4.07	3.87
Summer 10	2	3.20	3.90
Fall 10	1	5.00	5.00
	=======	=======	=======
Year 2010	6	3.93	4.07

The submitted student evaluations were generally positive. The average score for the valuation of the course outcomes was 3.93 out of 5, and for the adequacy of coverage was 4.07 out of 5. But the low participation rate makes firm conclusions unwarranted.

6 Recommendations

More student evaluations and instructor appraisals are needed in two Mathematics Department courses to make the assessment more meaningful and accurate.

Subject Area: Programming (Reported by Tim Downey) 5/8/2012

COP 2210 Computer Programming I
COP 3337 Computer Programming II
COP 3530 Data Structures
COP 4338 Computer Programming III

COP 2210 Computer Programming I

All objectives are covered on an assignment and/or an exam.
All objectives are considered essential or appropriate.
Most of the instructors thought that the students' preparation for taking the course was adequate. One instructor felt that it was deficient.
Several students feel that the course should have a programming class as a prerequisite. One student would like to use GUI instead of command line.
Summer offering had low survey results; students did not feel that most topics were covered adequately; several comments that instructor did not follow text.
Several negative comments about the book Big Java.

Recommendation:

GUI could be used in course as an alternative to the command line, but not very advanced; since the HCI course is a co-requisite, GUI would be better taught in COP 3804.
Upon follow up, the instructor who thought students were un prepared was only commenting on the lack of math skills. Since this course is considered a math class in the University curriculum, he felt that the preparation was adequate for a first level math class.
Summer offerings do not have as many class sessions as regular semesters. Encourage instructors to follow the recommended curriculum and use the recommended text.
Even though some students complained about the book, the instructors felt that it was a good book for the course. The negative comments from the students were not wide spread.

COP 3337 Computer Programming II

All objectives are covered on an assignment and/or an exam; except one instructor did not cover interfaces adequately in one term.
All objectives are considered essential or appropriate.
All objectives were covered extensively or adequately. Student evaluations confirm the instructor's appraisals, except for one class. In that class, the student evaluation of the coverage of outcomes was low. The areas that were not covered were recursion, interfaces, stacks & queues and problem solving.
All prerequisite objectives were considered highly useful.
Student prerequisite preparation was generally good and adequate; several instructors reported multiple deficiencies for their students. Three sections reported a deficiency in Strings/ArrayLists. One section reported a deficiency in Objects/Classes.
Please also see the COP-3530 Data Structures comments.

Recommendation:
Programming II instructors should be strongly encouraged to cover all of the objectives for Programming I, especially Stacks & Queues and the Java Collections.

COP 3530 Data Structures

All objectives are covered on an assignment and/or an exam.
All objectives are considered essential or appropriate.
All objectives were covered extensively or adequately. Student responses supported this; except for a 3.97/5.00 for one semester in recursion.
An instructor rated the prep as deficient and noted interfaces, recursion, stacks and queues.
All but one instructor indicated that the mastery of the prerequisites was at least adequate.
Students think the course is too difficult.
Two semesters, the students were not entirely positive about the text: 3.85 and 3.88/5.00.

Recommendation:
The low perception of the book can be attributed to the difficulty of the course. The book is one of the most popular books on the subject.
We must stress to the COP3337 instructors to emphasize interfaces when possible and to be sure to cover recursion when covering stack, queues and linked lists.

COP 4226 Advanced Window Programming

All objectives are covered on an assignment and/or an exam.
All objectives are considered essential or appropriate.
All objectives were covered extensively or adequately. Student responses supported this, except for the outcome "Database Connectivity, Serialization, Drag and Drop, Multithreaded Programming". This outcome was rated 3.84/5
Students felt that the homework was not appropriate 3.6/5.
A student commented that programs should be larger part of grade and another that they should not build upon each other.

Recommendation:
The outcome "Database Connectivity, Serialization, Drag and Drop, Multithreaded Programming" is too broad. Database connectivity and multithreaded programming are entire courses in themselves. They do not seem appropriate. The outcome should be changed to "Data Sources, Serialization, Drag and Drop, Multithreaded Interfaces". The level of proficiency is 'familiar', so it seems that these topics could be introduced in the course.

COP 4338 Computer Programming III

Objectives were met less adequately as time progressed. By Spring 2011, Java and C++ were no longer being covered. In all semesters, reflection and STL were not covered adequately.
The textbook has a low rating by the students.
The relevant prerequisite objectives were rated from highly useful to irrelevant. This corresponds to the change in the objectives.

The preparation of the students was rated good, except for Spring 2011 where it was rated deficient due to lack of UNIX experience.
Students commented that it was difficult to cover Java and C/C++ in the same semester; did not like the textbook.

Recommendation:

The course is being redesigned. The new outcomes should be posted to the CES.
The students do not have UNIX in the curriculum; hopefully, the redesigned course will realize this and not expect students to have experience in it.

October 7, 2011

Software Engineering Area Report – Spring 2010 – Spring 2011

This report contains the assessment of the courses in the Software Engineering area taught in the School of Computing and Information Sciences (SCIS) for the review period that includes the semesters: Spring 2010, Summer 2010, Fall 2010, and Spring 2011. This area contains the courses: CEN 4010 Software Engineering I, CEN 4021 Software Engineering II, CEN 4076 Fundamentals of Software Testing and CIS 4911 Senior Project.

During the review period the following courses and sections were offered:

- CEN 4010 – one section each Spring 2010, Summer 2010, Fall 2010 and Spring 2011;
- CEN 4021 Software Engineering II - one section each in Spring 2010 and Spring 2011;
- CEN 4072 Fundamentals of Software Testing – one section in Fall 2010
- CIS 4911 – one section each in Spring 2010, Summer 2010, Fall 2010 and Spring 2011.

This report was prepared using the results from the online student course evaluation system and the instructor course appraisal system for the review period.

Course outcomes:

CEN 4010:
1. Be familiar with the Software Development Life Cycle
2. Master the techniques to gather and specify the requirements of a medium-size software system using UML,
3. Master the techniques to design and implement a medium-size software system
4. Be familiar with software testing techniques
5. Be familiar with software documentation
6. Be familiar with working in a small software development team
7. Be familiar with system walkthroughs

CEN 4021:
1. Master techniques of planning and monitoring the progress of a software project
2. Master software project cost estimation techniques
3. Be familiar with software architectures
4. Be familiar with software development team structures

CEN4072
1. Be familiar with creating, evaluating and implementing a test plan for a medium-size code segment.
2. Be familiar with program inspections.
3. Master the techniques used to perform specification-based testing and implementation-based testing on programs.
4. Be familiar with the techniques that apply test adequacy coverage criteria to the implementation model.
5. Be familiar with GUI testing.
6. Be exposed to program debugging.
7. Be familiar with tools to support testing, coverage analysis and debugging.

CIS 4911:

1. Mastery of problem formulation.
2. Demonstrate mastery of specifying the requirements of a problem.
3. Demonstrate mastery of designing the solution to a problem.
4. Demonstrate mastery of realizing the solution to a problem.
5. Demonstrate the ability to validate and evaluate the solution to a problem.
6. Demonstrate the ability to manage a semester long project.
7. Demonstrate the ability to work effectively in a project team.
8. Demonstrate the ability to think logically and critically when developing the solution to a given problem.
9. Demonstrate the ability to apply concepts learned in various courses when developing the solution to a given problem.
10. Demonstrate the ability to communicate the details of the technical solution through verbal and written modes.
11. Demonstrate the ability to incorporate ethical issues into the project development and documentation process.

Student Course Assessments:

The summary for the software engineering courses for the review period includes the results of the survey on course delivery, course outcomes and student suggestions. The course delivery criteria included (1) the student's preparation for taking the course, (2) the level of difficulty of the course, (3) an evaluation of the required text, and (4) the amount of home work required for the course. The course outcomes are listed in the previous section.

CEN 4010:

A total of 53 students completed the online course evaluations for the sections of CEN 4010 taught during the review period. The following table summarizes the results of the student surveys:

Table 1: Summary of the student evaluations for CEN4010.

Sem.	Resps.	Part A: Survey of course delivery (%)					Part C: Overall valuation of outcomes (%)					Part C: Overall adequacy of coverage of the outcomes (%)				
		SA	MA	NS	MD	SD	SA	MA	NS	MD	SD	SA	MA	NS	MD	SD
Sp '10	5	75	15	0	0	0	100	0	0	0	0	100	0	0	0	0
Su '10	14	43	32	11	11	4	60	33	6	1	0	59	38	2	1	0
Fa '10	13	48	25	8	13	6	71	24	4	0	0	51	22	19	6	2
Sp '11	21	56	33	4	5	0	62	35	1	2	0	60	32	4	4	0
	53	**55**	**26**	**8**	**7**	**2**	**73**	**23**	**3**	**1**	**0**	**67**	**23**	**6**	**3**	**1**

SA – Strongly Agree; MA- Moderately Agree; NS – Not Sure; MD- Moderately Disagree; SD – Strongly Disagree. The first column contains the semesters: Sp – Spring, Su - summer, Fa – Fall. The last row of the second column contains the total number of student over the review period. The remaining columns in the last row are the average percentages for the various catergories.

The majority of the students surveyed (81%) either strongly agreed (55%) or moderately agreed (26%) that the course delivery was good, see Table 1. When compared to the 2009 reporting period, the percentage of students who either strongly agreed or moderately agreed that the course delivery was good increased by 9%. A majority of the students strongly or moderately agreed that the level of difficulty was adequate (87%). The overall mean for the level of difficulty was 4.43/5.00. There was an increase in this category by 11% as compared to 2009. A majority (70%) of the students strongly (52%) or moderately (18%) agreed that the required textbook was suitable. The mean score for the suitability of the textbook was 4.1/5.0, this was an increase from 2009 (3.55/5.00). Note that the book was changed in Spring 2011 (mean of 4.45/5.0) for one semester. In 2008 the score (mean of 4.28/5.00) was higher than 2009 and the current review period. The lowest scoring attribute in the course delivery criteria was the preparation for taking the course (mean 4.0/5.00).

The results obtained for the course outcomes showed a similar trend. Approximately 96% of the students strongly agreed (73%) or moderately agreed (23%) that overall the course outcomes were valuable. Approximately 90% of the students strongly agreed (67%) or moderately agreed (23%) that the course outcomes were adequately covered in class. Both these values showed an increase from 2009, which scored 84% and 80% respectively.

CEN 4021:

Five (7) students completed the online survey for the course.

Table 2: Summary of the student evaluations for CEN4021.

Sem.	Resps.	Part A: Survey of course delivery (%)					Part C: Overall valuation of outcomes (%)					Part C: Overall adequacy of coverage of the outcomes (%)				
		SA	MA	NS	MD	SD	SA	MA	NS	MD	SD	SA	MA	NS	MD	SD
Sp '10	**7**	**82**	**11**	**0**	**4**	**0**	**82**	**11**	**7**	**0**	**0**	**89**	**11**	**0**	**0**	**0**

The majority of the students surveyed (93%) either strongly agreed (82%) or moderately agreed (11%) that the course delivery was good. The lowest attribute was the preparation for taking the course which recorded a mean value of 4.43/5.00. There was a significant improvement for both the course delivery and the course text (mean 2.80/5.00) over 2009. Note that in 2009 this course was taught as a cross listed course with CEN 5064 Software Design (graduate course).

The results obtained for the course outcomes were very positive. An estimated 93% of the students strongly agreed (82%) or moderately agreed (11%) that the course outcomes were valuable. Over 100% of the students strongly agreed or moderately agreed that the course outcomes were adequately covered in class, see Table 2. These survey results are a significant improvement over 2009. Maybe this is due to the fact that the course is no longer cross listed with the CEN 5064 Software Design course.

CEN 4072:

Seventeen (17) students completed the online survey for the course.

Table 3: Summary of the student evaluations for CEN4072.

Sem.	Resps.	Part A: Survey of course delivery (%)					Part C: Overall valuation of outcomes (%)					Part C: Overall adequacy of coverage of the outcomes (%)				
		SA	MA	NS	MD	SD	SA	MA	NS	MD	SD	SA	MA	NS	MD	SD
Sp '10	**17**	**43**	**35**	**7**	**7**	**7**	**62**	**34**	**2**	**1**	**1**	**53**	**38**	**3**	**3**	**3**

The majority of the students surveyed (78%) either strongly agreed (43%) or moderately agreed (35%) that the course delivery was good, see Table 3. The lowest attribute was the suitability of the textbook which recorded a mean value of 3.35/5.00.

The results obtained for the course outcomes were positive. An estimated 96% of the students strongly agreed (62%) or moderately agreed (34%) that the course outcomes were valuable. Approximately 91% of the students strongly agreed (53%) or moderately agreed (38) that the course outcomes were adequately covered in class. It should be noted that this is the first time the class was taught at FIU.

CIS 4911:

Forty-one (41) students completed the student evaluation during the review period.

Table 4: Summary of the student evaluations for CIS4911.

Sem.	Resps.	Part A: Survey of course delivery (%)					Part C: Overall valuation of outcomes (%)					Part C: Overall adequacy of coverage of the outcomes (%)				
		SA	MA	NS	MD	SD	SA	MA	NS	MD	SD	SA	MA	NS	MD	SD
Sp '10	5	50	35	10	0	5	98	2	0	0	0	75	25	0	0	0
Fa '10	16	28	33	16	9	14	73	26	1	1	1	56	29	0	6	9
Sp '11	20	38	35	14	9	5	71	24	4	0	0	51	22	19	6	2
	41	**39**	**34**	**13**	**6**	**8**	**80**	**17**	**2**	**0**	**0**	**61**	**26**	**6**	**4**	**4**

The majority of the students surveyed (73%) either strongly agreed (39%) or moderately agreed (34%) that the course delivery was good, see Table 4. The overall mean of the course delivery was 3.89/5.00. A majority of the students (84%) strongly (45%) or moderately (39%) agreed that the level of difficulty was adequate (mean = 4.1/5.00). A majority (93%) of the students strongly (55%) or moderately (38%) agreed that they were adequately prepared for the course (mean = 4.35/5.00). The lowest score was related to the homework needed for the course, a majority of the students (75%) strongly (32%) or moderately (43%) agreed that the amount of homework for the course was appropriate (mean = 3.77/5.00). There is no text book for the course although the evaluation included the question related to the suitability of the textbook. The students gave this question the lowest mean score, 3.35/5.00.

Suggestions/Comments (Students):

CEN 4010:

- The student suggestions were generally positive with respect to the course instructors.
- Several students stated that taking a Database course and a Web Development (php, ajax,or jsp programming) course would better prepare them for this class.
- Some students from Computer Engineering stated that the course seems unnecessary.
- One student thought that having groups of size 7 was too large.

CEN 4021

- The project needs to be more realistic since some students have not yet worked in industry.

CEN 4072

- Students liked how they learned to use several testing tools.

- Course is very useful for Computer Science students since most entry-level positions in software development companies begin with some sort of testing.
- Regular homework assignments may be better than a large project.
- More examples should be done in class.
- There should be in class demos on how to use the software testing tools.

CIS 4911
- Too much work in one semester.
- Suggest that the course be more organized and several professors be involved in the process of designing the course.
- Was not prepared enough for the course on the programming perspective. Would not have been able to complete the course using the knowledge I gained from school.
- Allow students to work on individual projects, sometimes working in teams make the project harder.
- Create a course where students can learn php, jsp, or asp as part of the computer science curriculum.
- The course was too much software engineering and enough computer science. Ability to select mentors, teammates, and projects would be nice.
- The course was excellent and provided me with a solid chance to reinforce the knowledge gained in the Software Engineering course.
- Class should be a year long. Should work on projects without focusing on too much documentation rather the focus should be on perfecting the project.
- Have a seminar for the advisors to really explain to them what is expected and maybe some tips on dealing with undergraduate students, especially those that don't teach undergraduate level courses.
- The course has an extremely strong software engineering focus as ran by Clarke. Some software engineering classes taught by other professors do not cover all the material expected by Clarke. The mentor system is not working for some groups. There was some confusion with respect to the document formats that were required. Some mentors wanted more of a research format for the documents but did not provide an outline of the format. The mentors need to be on board with Senior Project and treat it as a real class; they need to devote time to "teaching".
- Professor Clarke does not do a good job teaching the material, he goes very quickly over all the slides. More appropriate teaching methodology would be very good for this course.

Instructor Course Assessments:

CEN 4010:
The instructors for the sections taught in the Summer 2010, Fall 2010, and Spring 2011 semesters reported that the course objectives were covered using a variety of evaluation methods including tests, assignments, review papers, project presentations and deliverables. All the course objectives were either extensively or adequately covered for all the semesters. In one cases it was stated that note enough on testing techniques was covered in the class. There was no assessment for Spring 2010.

CEN 4021:
The instructor for the sections taught in the Spring 2010 and Spring 2011 semesters reported that the course objectives were covered using a variety of evaluation methods including tests and assignments. All the course objectives were either extensively or adequately covered for all the semesters.

CEN 4072:
There was no assessment done for the course in Fall 2010. This was the first time the course was offered and the online instructor course assessment was not ready.

CIS 4911

The instructor for the sections taught in the Spring 2010, Fall 2010, and Spring 2011 semesters reported that the course objectives were covered using a variety of evaluation methods including project presentations and deliverables. All except one of the course objectives were either extensively or adequately covered for all the semesters. The only course objective that was not adequately covered was Ethical Issues.

Prerequisite Outcome Suggestions (Instructors):

CEN 4010:

- Knowledge on using server-side technologies such as Tomcat, Apache Server, PHP/JSP/ASP. Student exposure to graphical user interface design technologies would also be beneficial to students taking this course.
- This class has significant interactions with programming and databases. Perhaps a database class would enhance the students' abilities in the class. Too many students have little to no programming experience in the technologies used to implement the class project according to their feedback.

CIS4911

- There needs to be more preparation in the area of ethical issues, particularly how to document ethical concerns. For example, how to cite copyright and trademark information and how to write their licenses for nay artifacts created.
- Students need to be better trained in verbal communication, particularly in their presentation skills.
- Students need to be better prepared in the area of systems, particularly in the area of hardware and software required to implement a "real" system. For example, the hardware and systems software (servers) required to support web-based applications.
- Students need to be exposed to more project management concepts and practice more of these concepts by working in teams.

General Comments (Instructors):

CEN 4010:

- Course should be divided into two separate courses and maybe combined with the Senior Project course.
- Students don't get a chance to truly master design techniques and methodologies. There should spend one semester mastering the design techniques and another semester applying them.
- Students were not aware of the type of projects being developed in the "real" world.
- There should be a mentoring program where alumni who have field experience can provide the students with some guidance.
- If the requirements to teach students web technologies it may be necessary to revert back to students implementing stand alone applications for their course project.

CIS4911:

- There needs to be more participation by industry during the development and mentoring of student projects.

Recommendations:

CEN4010:

1. There is a need to have students take a programming course that contains web-based programming and learning technologies such as Tomcat, Apache Server, PHP/JSP/ASP.
2. The problem with the database prerequisite seems to have been solved now that the Database course is a co-requisite for the software engineering class.

CEN4021:

1. Now that the CEN 4021 course is using the syllabus as stated in the catalog many of the problems from the 2009 report seem to have been solved.

CEN 4072:

1. The second edition of the course is being taught in Fall 2011 and it may be too early to make any changes to the textbook.

CIS 4911

1. If the senior project course is to be taken seriously then SCIS must find a way to get faculty involved in the course and the faculty must dedicate the time and effort in order for the course to be a success. The course cannot be treated solely as a way for undergraduate students to work on research projects or to do "on the side" projects for faculty members.
2. There area of ethical issues needs to be adequately covered in a prerequisite course. Students are creating software artifacts and must know how to acknowledge other people's work being used, and how to write the appropriate licenses to protect their own work. In addition, it is important for them to have some understanding of the privacy and security issues when they are writing software in some domains, e.g., healthcare.
3. Students need additional practice in both written and verbal communication.

Peter J. Clarke
Software Engineering Area Coordinator

APPENDIX D: Exit (Graduating Student) Survey and Alumni Survey raw data

APPENDIX E: Course-Embedded Assessment Summaries, Fall 2010 & Spring 2011

Fall 2010

Summary of Direct Measure Assessment Data for the BS in Computer Science

In accordance with the SCIS Assessment Plan for the BS in Computer Science, direct measures of attainment of Student Outcomes were performed as follows:

1. Embedded Assessment of BS in CS Student Outcome (a) (Foundations area) in MAD 3512 Theory of Algorithms.
2. Embedded Assessment of BS in CS Student Outcome (b) (Computer Science core) in COP 4555 Principles of Programming Languages.
3. Assessment of BS in CS Student Outcome (e) (Social & Ethical concerns) in CGS 3092 Professional Ethics and Social Issues in Computing.
4. Assessment of all BS in CS Student Outcomes, (a) through (h), via observation of the 7 Senior Projects presented in Fall 2010.

BS in CS Student Outcomes (Revised Fall 2010)

To complete the program of study for the BS in Computer Science, every student will

q) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.
r) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.
s) Demonstrate proficiency in problem solving and application of software engineering techniques.
t) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.
u) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.
v) Demonstrate the ability to work cooperatively in teams.
w) Demonstrate effective communication skills.
x) Have experience with contemporary environments and tools necessary for the practice of computing.

Embedded Assessment of Outcome (a) in MAD 3512 Theory of Algorithms

4 students enrolled in MAD3512 completed a 5-question multiple choice assessment quiz. The quiz and results are attached. Due to scheduling anomalies, only 4 students completed MAD 3512 in Fall 2010.

Expectation: 75% of students completing the assessment quiz should score 3.5/5 or higher.

Observation: 1 student scored 3/5, 2 students scored 2/5, 1 student scored 0/5.

Conclusion: The available data are insufficient. This assessment will be repeated in Spring 2011.

Embedded Assessment of Outcome (b) in COP4555 Principles of Programming Languages

19 students enrolled in COP 4555 completed a 10-question multiple choice assessment quiz. The quiz and results are attached.

Expectation: 75% of students completing the assessment quiz should score 7/10 or higher.

Observation: The average score was 6.84 out of 10, and the median score was 7 out of 10. 63% of the students answered at least 7 out of 10 questions correctly.

Conclusion: Student attainment on this facet of outcome (b) is **marginally lower** than expected.

Embedded Assessment of Outcome (e) in CGS 3092 Professional Ethics and Social issues

From each of 10 assigned topics, one oral (PowerPoint) student presentation and one written (paper) student presentation were analyzed to determine whether the presentation addressed issues of Social Concern and/or issues of Ethical Concern. For each facet (Social, Ethical), the analysis identified whether an assertion under that facet (Social, Ethical) was supported by evidence, and whether counter arguments on that assertion were provided. For each topic, this analysis yielded 8 binary (0/1) scores for an overall rating in the range 0 .. 8.

Expectation: Each topic should receive a minimum rating of 75% or a score of 6 from 8.

Observation: 8 topics were rated at 100% (8/8), 2 topics were rated at 50% (4/4).

Conclusion: Student attainment of outcome (e) evidenced in CGS 3092 is at a **very high** level.

Assessment via Senior Project

7 projects were observed for the purpose of obtaining ratings of attainment of BS-CS outcomes by at least 2 faculty members. The ratings are on a scale of 1..5, or 0 if the project provided insufficient evidence about a particular outcome. A mediation rating was obtained when the initial ratings differed by more than 1 point, or when a rater did not respond prior to preparation of this summary. The scoring rubric followed by the raters is attached. The project ratings are summarized in the following table. The mediation ratings (if any) are in **bold**.

	Outcome (a)	Outcome (b)	Outcome (c)	Outcome (d)	Outcome (e)	Outcome (f)	Outcome (g)	Outcome (h)
Project 1	3	2	2	2	4	5	5	5
	5	5	5	5	5	5	5	5
	3	**3**	**5**	**5**	**4**	**5**	**5**	**5**
Project 2	1	0	5	5	4	5	5	5
	2	4	5	5	3	5	5	4
	1	**2**	**5**	**5**	**5**	**5**	**5**	**4**
Project 3	5	5	5	5	5	5	5	5
	4	4	4	4	4	5	4	4
Project 4	2	2	4	3	2	3	4	4
	2	4	5	5	1	5	5	4
	3	**2**	**5**	**5**	**1**	**5**	**5**	**4**
Project 5	4	5	5	5	5	5	5	5
	3	4	5	5	5	5	5	5
Project 6	3	5	5	5	5	5	5	5
	3	**5**	**5**	**5**	**5**	**5**	**5**	**5**
Project 7	4	5	5	5	4	5	5	5
	3	**5**	**5**	**5**	**5**	**5**	**5**	**5**

	Outcome (a)	Outcome (b)	Outcome (c)	Outcome (d)	Outcome (e)	Outcome (f)	Outcome (g)	Outcome (h)
Mean	***3.00***	***3.88***	***4.88***	***4.82***	***3.94***	***4.88***	***4.88***	***4.65***

The means expressed in the final row of the table are averaged over all ratings including the mediation ratings, and excluding any 0 rating (0 = not applicable).

Reliability: Prior to mediation, 5 of the 7 projects were each rated across all 8 student outcomes by 2 raters. The consistency of the outcome attainment ratings is summarized in the following table.

Identical Ratings	Ratings differing by 1	Ratings differing by 2+
18/40	16/40	6/40
45%	40%	15%

85% of the ratings are either identical or differ by 1. The greatest disparity occurs in the ratings of outcomes (a) and (b) where 4 ratings differ by more than 1, and 5 differ by exactly 1, indicating that the rubric for rating outcomes (a) and (b) must be refined to decrease rating subjectivity.

Expectation: Attainment of all outcomes should be 75% or 3.75 on a 1—5 scale, or better.

- Outcome (a): *Demonstrate proficiency in the foundation areas of Computer Science…*

Observation: 2 of 16 raters scored attainment of outcome (a) as *excellent (5)*, and 3 scored it as *very good (4)*; 7 raters scored it as *good (3)*; 5 raters scored attainment as *either fair (2)* or *poor (1)*.

Conclusion: Attainment of outcome (a) evidenced by the Senior Projects is **below the acceptable level**. It is probable that several projects incorporated few elements of this outcome.

- Outcome (b): *Demonstrate proficiency in various areas of Computer Science…*

Observation: 7 of 16 raters scored attainment of outcome (b) as *excellent (5)* and 4 scored it as *very good (4)*; 1 rater scored it as *good (3)*; 4 raters scored it as *fair (2)*; 1 rater of project 2 thought that attainment of outcome (b) was not demonstrated.

Conclusion: Attainment of outcome (b) is demonstrated in the Senior Projects at an **acceptable level**. The ratings of projects 3, 5, 6 and 7 all indicate very high attainment. It is probable that the lower ratings of other projects reflect a greater emphasis on the Software Engineering components of those projects.

- Outcome (e): *Demonstrate understanding of the social and ethical concerns …*

Observation: Only project 4 received ratings of *fair (2)* or *poor (1)* for attainment of outcome (e). 1 project 2 rater assigned a rating of *good (3)*; 5 raters scored it as *very good (4)*; 8 raters scored it as *excellent (5)*.

Conclusion: There is clear indication of **very high** student attainment of outcome (e).

- Outcome (c): *Demonstrate proficiency in problem solving and application of software engineering…*
- Outcome (d): *Demonstrate mastery of at least one modern programming language…*
- Outcome (f): *Demonstrate the ability to work cooperatively in teams.*
- Outcome (g): *Demonstrate effective communication skills.*
- Outcome (h): *Have experience with contemporary environments and tools…*

Observation: Attainment of outcomes (c), (d), (f), (g) and (h) as demonstrated in the Senior Projects is almost uniformly rated as *very good (4)* or *excellent (5)* across all seven projects.

Conclusion: There is clear indication of **very high** student attainment of outcomes (c), (d), (f), (g) and (h).

Spring 2011 Summary of Direct Measure Assessment Data for the BS in Computer Science

Prepared by Norman Pestaina, SCIS Undergraduate Programs Assessments Coordinator.
September 12, 2011

In accordance with the SCIS Assessment Plan for the BS in Computer Science, direct measures of attainment of Student Outcomes were performed as follows:

5. Course-embedded Assessment of BS in CS Student Outcome (a) (Foundations area) in MAD 3512 Theory of Algorithms.
6. Course-embedded Assessment of BS in CS Student Outcome (b) (Computer Science core) in COP 4540 Database Management.
7. Course-embedded Assessment of BS in CS Student Outcome (b) (Computer Science core) in COP 3530 Data Structures.
8. Assessment of all BS in CS Student Outcomes, (a) through (h), via observation of the five Senior Projects presented in Spring 2011.

The data obtained via these direct measures are summarized here. The following documents are referenced in this summary and are attached:

- Email communication of the MAD 3512 data from Dr. Dev Roy
- COP 4540 Embedded Assessment Report prepared by Dr. Shu-Ching Chen
- COP 3530 Results of Assessment Quiz prepared by Dr. Jai Navlakha
- Mapping of COP 3530 quiz questions to course outcomes prepared by Mr. Norman Pestaina
- Spring 2011 Rubric for assessing BS-CS Student Outcomes in Senior Projects

For reasons of confidentiality, the MAD 3512 and COP 3530 quizzes are not included here. These, as well as the Senior Project raw data, may be made available for inspection as needed.

BS in CS Student Outcomes (Revised Fall 2010)

To complete the program of study for the BS in Computer Science, every student will

y) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms.
z) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.
aa) Demonstrate proficiency in problem solving and application of software engineering techniques.
bb) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.
cc) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.
dd) Demonstrate the ability to work cooperatively in teams.
ee) Demonstrate effective communication skills.
ff) Have experience with contemporary environments and tools necessary for the practice of computing.

Embedded Assessment of BS-CS Student Outcome (a) in MAD 3512 Theory of Algorithms

Course Outcomes:

1. Be familiar with formal languages
2. Master finite state machines
3. Master Turing machines
4. Be familiar with primitive recursive and recursive functions
5. Be exposed to recursive unsolvability

12 students enrolled in MAD3512 completed a 5-question multiple choice assessment quiz. Because of a typographical error in one of the questions, the results from only 4 of the questions are considered. The results are summarized as follows:

Correct Answers	# of Students	Cumulative %	
4 = 100%	6	50	(6 / 12)
3 = 75%	3	75	(9 /12)
2 = 50%	2	92	(11/12)
1 or 0	1	100	

MAD 3512 TABLE 1: Number of Correct Answers by Number of Students

Expectation:

75% of students completing the assessment quiz should score 3/4 or higher.

Observation:

Exactly 75% of students completing the assessment quiz scored 3/4 or higher.

Conclusion:

The available data suggests that attainment of Outcome (a) is **very high**.

Discussion:

The MAD 3512 Instructor indicated that the same quiz was given in both Spring 2011 and Fall 2010. Because of the small number of students (4) completing the quiz in Fall 2010, no conclusions were drawn from student performance in Fall 2010. Nonetheless, a cursory comparison of the scores in these two semesters suggests a significantly improved indication in the current data for Spring 2011.

The MAD 3512 instructor's summary does not permit a question-specific analysis. This would have been helpful to allow course outcome analysis with a view to identifying where improvement is needed.

Embedded Assessment of BS-CS Student Outcome (b) in COP 4540 Database Management

Course Outcomes

1. Be exposed to information systems
2. Be familiar with database system and database architecture
3. Master the design conceptual schemas
4. Master normalization theory and the mapping of a conceptual schema to a relational schema
5. Master the expression of queries in SQL, relational algebra, and relational calculus
6. Be familiar with physical database design
7. Be familiar with writing application programs that use SQL

13 students enrolled in COP 4540 completed a 5-question multiple choice assessment quiz. The quiz and scores are attached. The results may be summarized as follows:

Correct Answers	# of Students	Cumulative %
5 = 100%	5	38 (5 / 13)
4 = 80%	7	92 (12 / 13)
3 = 60%	1	100

COP 4540 TABLE 1: Number of Correct Answers by Number of Students

Expectation:
75% of students completing the quiz should answer 4 or 5 questions correctly..

Observation:
92% of students answered either 4 or 5 quiz questions correctly.

Conclusion:
Student attainment of Outcome (b) evidenced in COP 4540 is at a **very high** level.

Discussion:
The following table summarizes the COP 4540 quiz results by individual question and shows a mapping of quiz questions to course outcomes:

Question #	**1**	**2**	**3**	**4**	**5**
# of Correct Answers (of 13)	*13*	*13*	*12*	*7*	*11*
% of Answers Correct	*100*	*100*	*92*	*54*	*85*
Course Outcomes mapped	1,2	3,4,6	5	5	7

COP 4540 TABLE 2: Number of Correct Answers to each Question

With the exception of one facet of Course Outcome 5, there is evidence of high student attainment of all COP 4540 course outcomes. The course instructor's report includes an observation relevant to this that may be considered by the Systems Subject Area Coordinator.

Embedded Assessment of Outcome (b) in COP3530 Data Structures

25 students enrolled in COP 3530 completed a 10-question multiple choice assessment quiz. The quiz and scores are attached. The results may be summarized as follows:

# of Correct Answers	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**
# of Students	*0*	*2*	*1*	*6*	*7*	*4*	*3*	*2*
Cumulative # of Students	*0*	*2*	*3*	*9*	*16*	*20*	*23*	*25*

COP 3530 TABLE 1: Number of Correct Answers by Number of Students

Expectation:

75% of students completing the assessment quiz should answer 7 or more questions correctly.

Observation:

9 of 25 students (36%) answered 7 or more of 10 questions correctly;
16 of 25 students (64%) answered 6 or more of 10 questions correctly.
20 of 25 students (80%) answered 5 or more of 10 questions correctly

Conclusion:

Student attainment on this facet of Outcome (b) as evidenced in COP 3530 is **significantly lower** than expected.

Discussion:

The following table summarizes the COP 3530 quiz results by individual question and shows a mapping of quiz questions to course outcomes:

Question #	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
# of Correct Answers	*15*	*13*	*15*	*11*	*22*	*21*	*22*	*6*	*13*	*10*
% of Answers Correct	*60*	*52*	*60*	*44*	*88*	*84*	*88*	*24*	*52*	*40*
Course Outcomes mapped	*1, 4*	*1*	*1*	*2, 3*	*4*	*3*	*3, 4*	*6*	*1, 5*	*1, 5*

COP 3530 TABLE 2: Number of Correct Answers to each Question

Of the 10 questions, 3 questions (#'s 5, 6 and 7) were answered correctly by at least 80% of the students and only a further 2 questions (#'s 1 and 3) were answered correctly by at least 50% of the students. Put together, only about half of the quiz questions were answered correctly by at least half of the students taking the quiz. Fully half of the quiz questions (#'s 2, 4, 8, 9 and 10) were answered incorrectly by more than about half of the students.

There are clearly some indicators here that should be addressed by the Subject Area Coordinator and/or faculty who teach COP 3530.

Assessment via Senior Project

5 projects were observed for the purpose of obtaining ratings of attainment of BS-CS outcomes by at least 2 faculty members. The ratings are on a scale of 1 .. 5, or 0 if the project provided insufficient evidence about a particular outcome. A mediation rating was obtained when the initial ratings differed by more than 1 point. The scoring rubric followed by the raters is attached. The project ratings are summarized in the following table. The mediation ratings (if any) are in **bold**.

	Outcome (a)	Outcome (b)	Outcome (c)	Outcome (d)	Outcome (e)	Outcome (f)	Outcome (g)	Outcome (h)
Project 1	3	5	5	5	3	5	5	5
WResTT	4	5	5	5	3	5	5	5
Project 2	3	5	5	5	5	5	5	5
P-Care-2	3	1	5	5	5	5	5	5
(M)	**3**	**3**	**5**	**5**	**5**	**5**	**5**	**5**
Project 3	3	5	5	5	5	5	5	5
P-NEXUS	5	5	5	5	5	5	5	4
(M)	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**4**
Project 4	0	2	5	4	4	5	5	5
Geon-DB	3	5	5	5	5	5	5	5
(M)	**1**	**2**	**5**	**4**	**4**	**5**	**5**	**5**
Project 5	5	5	5	4	4	5	5	5
Data Vis	3	4	5	4	2	5	5	5
(M)	**4**	**4**	**5**	**4**	**4**	**5**	**5**	**5**
	Outcome (a)	Outcome (b)	Outcome (c)	Outcome (d)	Outcome (e)	Outcome (f)	Outcome (g)	Outcome (h)
Mean	***3.46***	***4.00***	***5.00***	***4.64***	***4.21***	***5.00***	***5.00***	***4.86***

The means expressed in the final row of the table are averaged over all ratings including the mediation ratings, and excluding any 0 rating (0 = not applicable).

Reliability: Prior to mediation, all 5 projects were each rated across all 8 student outcomes by 2 raters. The consistency of the un-mediated outcome attainment ratings is summarized in the following table.

Identical Ratings	Ratings differing by 1	Ratings differing by 2+
29/40	5/40	6/40
72.5%	12.5%	15%

85% of the paired ratings are either identical or differ by 1. Of these 34 paired ratings, 5 differ by 1 and 29 are identical. Of the remaining 6 divergent paired ratings, 3 pairs differ by exactly 2 rating points and another 3 pairs differ by more than 2 (including one pair where a rater judged the project to provide insufficient evidence about an outcome (a)). It is noted that the previous Fall 2010 summary reported an identical 85-15% split between paired ratings differing by fewer than 2 points or differing by 2+ points. **Then however, only 45% of paired ratings were identical, compared with 72.5% in Spring 2011**. The scoring rubric was refined prior to the Spring 2011 application. There is still room for improvement.

The following standard is applied to all BS-BC Student Outcome ratings via the Senior Project.
Expectation: Attainment of all outcomes should be **75% or 3.75** on a 1—5 scale, or better.

- Outcome (a): *Demonstrate proficiency in the foundation areas of Computer Science…* ***3.46***

Observation: 3 of 14 raters scored attainment of outcome (a) as *excellent (5)*, and 2 scored it as *very good (4)*; 7 raters scored it as *good (3)*; 1 raters scored attainment as *poor (1)*. 1 rater of project 4 thought that attainment of outcome (a) was not demonstrated in that project.

Conclusion: Attainment of outcome (a) evidenced by the Senior Projects is **marginally below the acceptable level**. It is probable that several projects incorporated few elements of this outcome.

- Outcome (b): *Demonstrate proficiency in various areas of Computer Science…* ***4.00***

Observation: 8 of 14 raters scored attainment of outcome (b) as *excellent (5)* and 2 scored it as *very good (4)*; 1 rater scored it as *good (3)*; 2 raters scored it as *fair (2)*; 1 rater scored it as *poor (1)*.

Conclusion: Attainment of outcome (b) is demonstrated in the Senior Projects at an **acceptable level**.

- Outcome (e): *Demonstrate understanding of the social and ethical concerns …* ***4.21***

Observation: 7 of 14 raters scored attainment of outcome (e) as *excellent (5)* , 4 raters scored it as *very good (4)*, and 2 rater scored it as *good (3)*; Only project 5 received a *fair (2)* score by 1 rater.

Conclusion: There is clear indication of **high** student attainment of outcome (e).

- Outcome (c): *Demonstrate proficiency in problem solving and application of software engineering…**5.00***
- Outcome (d): *Demonstrate mastery of at least one modern programming language…* ***4.64***
- Outcome (f): *Demonstrate the ability to work cooperatively in teams…* ***5.00***
- Outcome (g): *Demonstrate effective communication skills…* ***5.00***
- Outcome (h): *Have experience with contemporary environments and tools…* ***4.86***

Observation: Attainment of outcomes (c), (d), (f), (g) and (h) as demonstrated in the Senior Projects is uniformly rated as *excellent (5)* or *very good (4)* across all five projects.

Conclusion: There is clear indication of **very high** student attainment of outcomes (c), (d), (f), (g) and (h).

APPENDIX F: Course-Embedded Assessment Data, Fall 2010 & Spring 2011

Results of Assessment Quiz
COP 4555 Principles of Programming Languages
Geoffrey Smith
November 30, 2010

On November 29, 2010 I gave a 10-question multiple-choice quiz of programming languages concepts in my COP 4555 class. There were 19 students present, out of 34 enrolled in the class. (Attendance has been a problem throughout the semester.)

The complete results are shown in the following table, which has a row giving the results for each student (labeled 'A' through 'S') and a column giving the results for each question, using '1' for 'correct' and '0' for 'incorrect'.

	1	2	3	4	5	6	7	8	9	10	# correct
A	1	1	1	1	1	1	1	1	1	1	10
B	1	1	1	1	1	1	1	1	1	1	10
C	1	1	1	1	1	1	1	1	0	1	9
D	1	1	1	1	1	0	0	1	1	1	8
E	1	1	1	1	1	0	1	0	1	1	8
F	1	1	0	0	1	1	1	1	1	1	8
G	1	0	1	1	0	1	1	1	1	1	8
H	1	1	0	0	1	0	1	1	1	1	7
I	0	1	1	0	1	1	1	1	0	1	7
J	1	1	1	1	0	1	1	0	0	1	7
K	1	1	0	1	1	0	1	1	0	1	7
L	0	1	1	1	0	0	1	1	1	1	7
M	0	0	1	0	1	1	1	0	1	1	6
N	1	1	0	1	1	0	0	0	0	1	5
O	0	1	0	0	0	1	1	0	1	1	5
P	0	0	0	0	0	1	1	1	1	1	5
Q	0	0	0	1	0	1	1	0	1	1	5
R	1	1	1	0	0	0	0	1	0	0	4
S	0	1	0	0	0	0	1	1	0	1	4
# correct	12	15	11	11	11	11	16	13	12	18	130

The average score was 6.84 out of 10, and the median score was 7 out of 10. Moreover, 63% of the students answered at least 7 out of 10 quesions correctly.

MAD 3512 Assessment Quiz				
	Student 1	Student 2	Stuident 3	Student 4
Problem 1	0	1	0	0
Problem 2	1	0	0	0
problem 3	0	1	0	0
Problem 4	1	0	0	1
Problem 5	1	0	0	1
	------	------	------	------
Raw Score	3	2	0	2
% Score	60	40	0	40

Fall 2010

Results of Assessment Quiz
COP 3530
Introduction to Data Structures
Jai Navlakha
April 26, 2011

On April 26, 2011, I gave a 10-question multiple-choice quiz of Data Structures concepts as part of my final examination for my COP 3530 class. There were 25 students present out of 34 enrolled in the class with 7 students having dropped out and 2 being absent. The complete results are shown in the following table, which has a row giving the results for each student (labeled `A' through 'Y') and a column giving the results for each question, using `1' for `correct' and `0' for `incorrect'.

	1	2	3	4	5	6	7	8	9	10	# Correct
A	1	1	1	1	1	1	1	1	1	0	9
B	0	1	1	1	1	0	1	1	0	1	7
C	1	0	0	1	1	1	1	1	1	0	7
D	0	1	0	0	0	1	1	0	1	0	4
E	1	0	1	0	1	1	1	0	0	0	5
F	1	0	1	0	1	1	1	0	1	0	6
G	0	0	1	0	1	1	1	0	1	0	5
H	1	1	1	0	1	0	1	1	0	0	6
I	0	0	0	0	1	1	1	1	1	1	6
J	0	1	1	1	1	1	1	0	0	0	6
K	1	1	1	0	1	1	0	0	1	0	6
L	1	0	1	0	1	1	1	0	1	1	7
M	1	0	1	0	0	1	1	0	1	1	6
N	0	0	1	1	0	1	1	0	0	1	5
O	1	1	1	1	1	1	1	0	0	0	7
P	1	1	0	0	1	1	1	0	1	1	7
Q	1	0	0	1	1	1	1	0	0	0	5
R	0	0	0	0	1	1	1	0	0	0	3
S	1	1	1	1	1	1	1	0	0	1	8
T	1	1	1	1	1	0	0	0	1	0	6
U	1	1	1	0	1	1	1	1	1	1	9
V	0	0	0	1	1	1	1	0	0	0	4
W	0	0	0	0	1	1	1	0	0	0	3
X	1	1	0	1	1	1	0	0	1	1	7
Y	0	1	0	0	1	0	1	0	0	1	4
# Correct	15	13	15	11	22	21	22	6	13	10	148

The average score was 5.92 out of 10, and the median score was 6 out of 10. Moreover, 64% of the students answered at least 6 out of 10 questions correctly.

COP4540 Embedded Assessment Report
Spring 2011
Prepared by Professor Shu-Ching Chen

On April 18, 2011, embedded assessment was conducted in the class. Thirteen (13) in-class students answer the five questions (Table 1) in order to check whether the learning results match with the course outcomes (Table 2).

Table 1: Questions

****** The test questions were edited out for this publication**
****** Norman Pestaina**

Table 2: Course Outcomes

Course Outcomes:

1. Be exposed to information systems
2. Be familiar with database system and database architecture
3. Master the design conceptual schemas
4. Master normalization theory and the mapping of a conceptual schema to a relational schema
5. Master the expression of queries in SQL, relational algebra, and relational calculus
6. Be familiar with physical database design
7. Be familiar with writing application programs that use SQL

Table 3 corresponds the questions to the course outcomes.

Table 3: Examination Questions Corresponding to Course Outcomes

Question	**Course Outcomes**
1	1, 2
2	3, 4, 6
3	5
4	5
5	7

Table 4 shows the results of students' answers to these five questions: correct answer (denoted as C) and wrong answer (denoted as W).

Table 4: Examination Results and Performance

Student PID	Question 1	Question 2	Question 3	Question 4	Question 5	# Correct / Total Questions	Correct Percentage (%)
2112921	C	C	C	W	C	4/5	80
1284668	C	C	C	W	C	4/5	80
2342960	C	C	C	W	C	4/5	80
2129261	C	C	C	W	W	3/5	60
2641540	C	C	C	C	C	5/5	100
3016141	C	C	C	C	C	5/5	100
2606594	C	C	C	W	C	4/5	80
2983006	C	C	W	C	C	4/5	80
3123497	C	C	C	C	C	5/5	100
2903992	C	C	C	W	C	4/5	80
3001363	C	C	C	C	C	5/5	100
2930093	C	C	C	C	C	5/5	100
2427645	C	C	C	C	W	4/5	80

Assessment:

From the results, it clearly shows that the students have good course outcomes: 5 students with 100% performance, 7 students with 80% performance, and 1 student with 60% performance. Students did very well for questions 1, 2, 3, and 5 (corresponding to course outcomes 1,2, 3, 4, 6, and 7). Six (6) students answered question 4 wrong. This question is related to transform Universal and Existential Quantifiers and some students don't have the enough background in this area though this topic was also discussed in the class.

I gave the students a 5-question exam, but one of the questions had a typo. So I used only four questions, with the following results.

4 correct: Excellent 6 students
3 correct: Good 3 students
2 correct: Pass 2 students
0 or 1 correct: Fail 1 student

Getting Norman and Mark off your back: Priceless.

BTW I'm not teaching MAD 3512 this fall. Taje Ramsamujh is.

Dev

On 9/6/2011 4:39 PM, Norman Pestaina wrote:
> Hello Dev,
>
> For Fall 2010, you sent us a grid showing how each student did on each
> question (see attached). Could you send us a similar grid for the Spring
> 2011 students? That would help greatly.
>
> Also, your results (Excellent, Good, etc.) suggest that your students did
> much better in the Spring. That's great.
>
> Norman
>
>> Thanks Dev!
>>
>> --Mark
>>
>> ---------- Forwarded message ----------
>> Date: Tue, 10 May 2011 17:39:01 -0400
>> From: Dev K. Roy<royd@fiu.edu>
>> To: Mark Allen Weiss<weiss@cs.fiu.edu>
>> Subject: Exit exam results
>>
>> Mark: Here are the results of the exit exam.
>> Excellent: 6 students
>> Good: 3 students
>> Pass: 2 students
>> Fail: 1 student
>>
>> I used the same exam as for fall 2010.
>>
>> Dev
>>

APPENDIX G: Senior Project Assessment Instruments

Rating-Sheet

Senior Project
Assessment of Student Outcomes of the BS in Computer Science of the FIU School of Computing and Information Sciences

Project Title _____

Number of team members: _____ Semester & Year _____

Project origination: _____

_____

Evaluator	Affiliation
_____	_____
_____	_____
_____	_____
_____	_____

==========

Your responses to this survey instrument will be used solely for the purpose of assessing the Student Outcomes of the BS in Computer Science program of the School of Computing and Information Sciences at FIU. The survey is expressly NOT for assessment of student performance in the SCIS Senior Project course, nor for assessment of the instructor(s).

For each Student Outcome, decide whether this project provides sufficient evidence to make a judgment about the students' attainment of that Student Outcome. If so, please indicate your assessment of the level of attainment of that Student Outcome demonstrated in this project:

Rating	Criterion
n/a	The project does not provide clear evidence about this particular outcome
1	The project demonstrates **poor** attainment of this outcome
2	The project demonstrates **fair** attainment of this outcome
3	The project demonstrates **good** attainment of this outcome
4	The project demonstrates **very good** attainment of this outcome
5	The project demonstrates **excellent** attainment of this outcome

BS in CS Student Outcomes Assessment via Senior Project

Student Outcomes		Rating
a)	Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms	
b)	Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems	
c)	Demonstrate proficiency in problem solving and application of software engineering techniques	
d)	Demonstrate mastery of at least one modern programming language and proficiency in at least one other.	
e)	Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.	
f)	Demonstrate the ability to work cooperatively in teams.	
g)	Demonstrate effective communication skills.	
j)	Have experience with contemporary environments and tools necessary for the practice of computing.	

Rubric (Spring 2011)

Senior Project
Assessment of Student Outcomes of the BS in Computer Science
of the
School of Computing and Information Sciences
Florida International University

The School of Computing and Information Sciences evaluates the Senior Projects of its graduating seniors for the purpose of assessing the level of attainment of the Student Outcomes of the BS in Computer Science program.

Your responses to this survey will be used solely for the purpose of assessing the Student Outcomes of the BS in Computer Science program of the School of Computing and Information Sciences at FIU. This survey is expressly NOT for assessment of student performance in the SCIS Senior Project course for assignment of letter grade, nor for assessment of the instructor(s).

Rating Instructions

For each program outcome, you are provided with a check-list of 7 or more criteria that evidence attainment of that outcome. Please check all criteria that are presented in this project. You may include additional criteria that are not explicitly listed; if so, please record the additional criteria in the spaces provided. Unless noted otherwise, the number of checked criteria, up to a maximum of 5, should be recorded as your rating of attainment of that outcome evidenced in the project.

Project Title _____

Semester & Year _____

Moderator (Faculty / Industry Sponsor): _____

Evaluators: _____

_____

_____

Student Outcome *(a)*: *Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms*

____ Project incorporates elements of mathematical reasoning or proof
(Lemma, Theorem, Propositional Logic, First Order Logic, Mathematical Induction)

____ Project utilizes elements of discrete mathematics
(Set Theory, Boolean Algebras, Combinatorics, Graph Theory)

____ Project utilizes some statistical procedure(s) to represent or summarize test data
(Mean, Standard Deviation, Stem Plot/Histogram, Box Plot/Percentile-Graph)

____ Project utilizes some statistical measure(s) of system behavior or performance
(Probability Distributions, Confidence Intervals, Hypothesis Testing)

____ Project design utilizes finite state diagrams to model system behavior

____ Project utilizes some aspect(s) of formal computer science
(Automata, Turing Machines, Recursive Function Theory, Recursive Unsolvability)

____ Project utilizes some technique(s) of numerical analysis
(Error Estimation, Interpolation, Numerical Calculus, Linear Systems, Matrix Algebra)

____ OTHER: _____

____ OTHER: _____

Student Outcome (b): *Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.*
Data Structures & Algorithms

____ Project utilizes an advanced data structure, e.g. search tree, hash table, priority queue

____ Project utilizes some graph algorithm, e.g. shortest path, minimum spanning tree

____ Project documents runtime analysis of selected algorithms

Concepts of Programming Languages

____ Project utilizes knowledge of programming language syntax
(Context-Free Grammars, Parse Trees, Ambiguity, Recursive Descent)

____ Project utilizes knowledge of programming language semantics
(Natural Semantics, Interpreters, Expressions, L- and R- Value, Environments)

____ Project demonstrates familiarity with design issues such as scoping rules, dynamic type checking, static type checking

Computer Systems (Database)

____ Project utilizes or designs an appropriate database management system

____ Project utilizes conceptual and/or relational schema

____ Project utilizes a database query language such as SQL

Computer Systems (Operating Systems)

____ Project implementation utilizes knowledge of memory management

____ Project implementation utilizes knowledge of process synchronization

____ Project documents analysis of tradeoffs in selection of system characteristics

____ OTHER: _____

____ OTHER: _____

Student Outcome *(c)*: *Demonstrate proficiency in problem solving and application of software engineering techniques.*

____ Project demonstrates knowledge of the Software Development Life Cycle

____ Project deliverables include Project Specification

____ Project deliverables include Feasibility Study and/or Project Plan

____ Project deliverables include Requirements Documentation

____ Project deliverables include Design Documentation

____ Project documents testing and/or evaluation of the implementation

____ Project incorporates system walkthroughs

____ OTHER: _____

____ OTHER: _____

Student Outcome (d): *Demonstrate mastery of at least one modern programming language and proficiency in at least one other.*

____ Project is implemented using an appropriate high level language

____ Project implementation is reasonably efficient rather than "brute force"

____ Project implementation is modular and/or re-usable

____ Project implementation uses a modern API or Tool-Kit

____ Project implementation utilizes recursion

____ Project implementation utilizes some advanced features, e.g. polymorphism

____ A project sub-system or module utilizes an appropriate programming language other than the primary implementation language, e.g. SQL, ML, assembly language

____ OTHER: _____

____ OTHER: _____

Student Outcome *(e)*: *Demonstrate understanding of the social and ethical concerns of the practicing computer scientist*

____ Project documents sources and references

____ Project identifies and addresses any relevant social issues

____ Project identifies and addresses any relevant ethical issues

____ Project identifies and addresses relevant legal issues

____ Project identifies and addresses any relevant privacy issues

____ Project documents anticipated impact on users/clients

____ Project documents and addresses any anticipated technology impact issues

____ OTHER: _____

____ OTHER: _____

Student Outcome (f): *Demonstrate the ability to work cooperatively in teams*

____ Project completion evidences equitable participation by team members

____ Project presentation(s) included all team members

____ Project team activity is documented

____ Project team set out and followed a schedule for timely completion

____ Project team negotiated consensus when needed

____ Team members roles were clearly defined and executed

____ Team members shared responsibility for success and failure

____ OTHER: _____

____ OTHER: _____

Program Outcome *(g)*: *Demonstrate effective communication skills*

____ Presentations described the essential features of the project

____ Presentations utilized good quality slides and presentation aids

____ Presenters utilized their time effectively

____ Presenters spoke directly to the audience

____ Technical features were communicated clearly

____ Project artifacts clearly document all project features

____ Project reports are well organized and written

____ OTHER: _____

____ OTHER: _____

Program Outcome (j): *Have experience with contemporary environments and tools necessary for the practice of computing*

____ Project utilized contemporary design tools

____ Project implementation utilized a modern IDE(s)

____ Project utilized appropriate validation/testing tools

____ Project was demonstrated using appropriate presentation tools

____ Project utilized appropriate project management tools (e.g., MS Project)

____ Project utilizes appropriate version control/document sharing tools

____ Project documents consideration of trade-offs in selection of tools

____ OTHER: _____

____ OTHER: _____

ABET Student Outcome

The program must enable students to attain, by the time of graduation:
(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices. [CS]

Please comment on how this project "*demonstrates comprehension of the tradeoffs involved in design choices"*:

APPENDIX H: Student Organization Reports

2011 Upsilon Pi Epsilon Report

Upsilon Pi Epsilon (UPE) is the international honor society for students in computer science, information technology, computer engineering, and management information systems. UPE student members must have completed at least 60 credits towards their academic degrees and have earned a 3.0 GPA or above.

During the 2011 Spring and Fall academic semesters, the members of the FIU chapter of UPE focused on organizing events to promote community service outreach.

Below we list the accomplishments and activities of UPE during 2011:

Robotics Course at Sweetwater Elementary School

Starting in Spring 2011, UPE and the FIU Honors College worked together to create a robotics course at Sweetwater Elementary school. The objective with the 4th grade robotics program at Sweetwater Elementary school was to spark children's imagination, creativity, and interest in technology. This was achieved through a series of "hands on" activities with four small electric and completely customizable robots. The class was divided into four teams - one robot and one student volunteer per team. Throughout a period of three months of weekly classes, the teams progressed from building the robots to programming them to race through small obstacle courses and compete in 'SumoBot' competitions.

The children learned the basics of team work and problem solving within the realm of technology. Each volunteer guided the children through every step of building and programming the robot. Towards the end of the semester, the students created PowerPoint presentations on their robots and presented them to the class, the school principal, and faculty from the FIU Honors College and School of Computing and Information Sciences. Since every team's robot turned out to be completely different, the presentations were very fun and interesting for children and adults alike!

IBM Mastering the Mainframe Contest

In Fall 2011, UPE hosted IBM's Mastering the Mainframe Contest for FIU students. UPE collaborated with IBM's Michael Todd, creator of the contest, and Juan Caraballo, Program Director of IBM's Latin American Grid program to provide FIU students with the opportunity to have their questions answered regarding skills necessary for a career working with mainframes. Participation in the event was high and FIU was one of the top ten schools in the country with the most student participants.

IEEE Advisory Board Meeting

UPE had the honor of engaging in a roundtable discussion with the Institute of Electrical and Electronic Engineers (IEEE) Computer Society Industry Advisory Board (IAB). UPE presented the research and educational activities of its members. This was the first time the Board met at FIU and it was an excellent opportunity to expose them to the educational setting at SCIS and leave a strong impression.

Interview with SCIS Director Candidate

Two members from UPE's executive board had the distinct opportunity to collect student concerns and questions to be addressed to Dr. Divyakant Agrawal, a former candidate for the SCIS Director position. The executive board engaged Dr. Agrawal in discussion and later presented their opinions based on the conversation to the SCIS faculty.

Chapter Report at National Upsilon Pi Epsilon Headquarters Meeting

In March 2011, the UPE president traveled to the 2011 ACM Special Interest Group for Computer Science Education (SIG CSE) conference to present at the National UPE Headquarters meeting. FIU's UPE chapter was one of ten UPE chapters across the nation to present the club's academic activities to a crowd of CS faculty including Dr. Donald Knuth, who later followed the chapter presentations with a question and answer session.

Jairo Pava
UPE Chapter President
2010-2011

APPENDIX I: Examples of Learning Outcomes

CDA 3103 Fundamentals of Computer Systems

Course Outcomes:

6. Master the representations of numeric and character data
7. Master the implementation of some basic combinational circuits, registers and memories
8. Be familiar with the data path of a simple von Neumann architecture and its relation to the instruction execution cycle
9. Master simple machine and assembly language programming
10. Master the implementation of high-level language constructs in lower levels: selection, iteration, function call/return

Learning Outcomes:

1.1 Derive and interpret the two's-complement representation of signed integers
1.2 Derive and interpret at least one representation of real numbers, e.g. IEEE Short Real
1.3 Interpret the representation of character data in some standard format, e.g. ASCII

2.1 Demonstrate the effect of NOT, AND, OR and XOR operations on binary data
2.2 Analyze a simple circuit using fundamental building blocks
2.2 Characterize the operation of the decoder, multiplexer, adder and simple memory circuits

3.1 Describe the organization and components of a simple von Neumann architecture
3.2 Demonstrate the implementation of simple machine language instructions using register transfer notation

4.1 Write programs in machine and assembly language employing flow-of-control and subroutine call and return constructions
4.2 Describe the operation of a simple 2-pass assembler

10.1 Demonstrate how conditional operations and transfer of control are implemented at the machine level
10.2 Demonstrate how parameters are passed to subroutines and how local workspace is created and accessed at the assembly language level

Sources:

CDA 3103 Syllabus: http://www.cis.fiu.edu/programs/undergrad/courses/COP_3402.pdf
ACM CS 2008: http://www.acm.org//education/curricula/ComputerScience2008.pdf

COP 4710 (COP 4540) Database Management

Course Outcomes

8. Be exposed to information systems
9. Be familiar with database system and database architecture
10. Master the design conceptual schemas
11. Master normalization theory and the mapping of a conceptual schema to a relational schema
12. Master the expression of queries in SQL, relational algebra, and relational calculus
13. Be familiar with physical database design
14. Be familiar with writing application programs that use SQL

Learning Outcomes

1.1 Explain basic information storage and retrieval concepts
1.2 Describe issues of information privacy, integrity, security and preservation

2.1 Describe the goals, components and functions of a database system
2.1 Explain the concept of data independence and its importance in a database system

3.1 Characterize the various data models
3.2 Design the conceptual schema for a database

4.1 Prepare a relational schema from a conceptual model

5.1 Demonstrate queries in relational algebra using union, intersection, difference, and Cartesian product operations
5.2 Demonstrate queries in tuple relational calculus, domain relational calculus, and SQL

6.1 Evaluate functional dependencies between two or more attributes in a relation

14.1 Describe database queries (insert, update, retrieve, and delete) using SQL statements

Sources

COP 4710 (COP 4540) Syllabus: http://www.cis.fiu.edu/programs/undergrad/courses/COP_4540.pdf
ACM CS 2008: http://www.acm.org//education/curricula/ComputerScience2008.pdf