2012-2013 Undergraduate Academic Assessment Plan

Computer Science in the College of Liberal Arts and Sciences

Computer Science

College of Liberal Arts and Sciences

Beverly Sanders, sanders@cise.ufl.edu

Computer Science Program
College of Liberal Arts and Sciences
Undergraduate Academic Assessment Plan

Mission Statement

The College of Liberal Arts and Sciences' computer science program exposes the student to a broad range of disciplines, including programming languages, theory of computer science, physical science, mathematics and software engineering. Students graduate with the ability to apply knowledge of science and mathematics to computer science problems, to design computer systems or components to satisfy users' needs and to communicate technical information regarding computer systems to other computer scientists.

The mission of the Computer Science Program is to produce a graduate who is adept at the discipline of Computer Science**.** The core of the Computer Science discipline is the ***algorithm*****,** a well-defined finite sequence of steps for solving a problem or performing a task. An algorithm takes as input data from a specified set, and for each input produces a desired output**,** which is the solution to the problem. The internal state of the algorithm as it processes it**s** data is represented in a ***data structure***. The ground-truth that the discipline relies upon **is *discrete mathematics***: logic, number theory, induction / recursion, coding theory, combinatorics, relations, graph theory, and formal methods. These mathematical elements define the terms and grammar of the discourse of computer science. A second foundation of computer science is ***architecture***, which also relies upon discrete mathematics. The architecture is an abstract model of a computational device, with sufficient detail for describing how the specific steps of an algorithm can execute its steps.

This algorithmic paradigm supports a wide range of applications, at the apex of the computer science discipline, with specific examples including graphics / image processing, computational science / numerical methods, simulation and modeling, aesthetic computing, artificial intelligence, databases, bioinformatics, cryptography / cyber-security, and the digital arts**.**

To enable these tasks requires a framework, both theoretical and practical, such as compilers and language theory, software engineering, operating systems, networks, and parallel / distributed computing. Compilers and their theory describe how high-level languages are translated into assembly language. Software engineering is the process of designing and executing a software system that meets desired specifications. Operating systems manage the resources of a computer system, provide an environment that enables a program to run, and a user-interface that enables the user to interact with the software. This framework, along with the foundation of discrete mathematics and architecture, enables an algorithm to be expressed and implemented.

This mission statement supports the three-fold UF Mission Statement: teaching, research & scholarship, and service, and the related Mission Statement of the College of Liberal Arts and Sciences (which is to expand our knowledge and practice in the most fundamental questions in the arts, humanities, social sciences, and natural and mathematical sciences through teaching, research and service), by:

- Teaching students in the computing discipline,
- Equipping them with the foundation for future graduate studies, as an integral part of the education process, and
- Enabling our students to serve the needs of the broader society

Student Learning Outcomes (SLOs)

The student acquires the following skills in the Computer Science major in the College of Liberal Arts and Sciences:

1. Apply knowledge of mathematics and science to computer science problems.
2. Design a computing system, component or process, analyzing and interpreting the data.
3. Use the techniques, skills and tools necessary for computer science practice.
4. Design a computing system, component or process to meet desired needs within realistic economic, environmental, social, political, ethical, and health and safety constraints.
5. Identify, formulate and solve computer science problems.
6. Communicate technical data and design information effectively in writing, in speech and in multidisciplinary teams to other computer scientists.

For the Academic Learning Compact for this program in the University of Florida catalog, see https://catalog.ufl.edu/ugrad/current/liberalarts/ALC/computer-science.aspx. See https://catalog.ufl.edu/ugrad/current/liberalarts/majors/computer-science.aspx for the major.

Curriculum Map

Curriculum Map for:

Computer Science Program College of Liberal Arts and Sciences

Key: **I**ntroduced **R**einforced **A**ssessed

Courses SLOs	COP 3504 Intro CS	COT 3100 Discrete Math	COP 3530 Data Struct	CDA 3101 Arch	CEN 3031 Software Eng.	COP 4600 Operating Sys.	COT 4501 Num. Methods	CIS 4914 Senior Design
Content Knowledge								
#1		I	R	R			A*	A*
#2							I, A*	A*
#3	I	R	R	R		A*		A*
Critical Thinking								
#4	I			R		R		A*
#5	I		R	R			R	A*
Communication								
#6					I, A*			A*

***See page 5 for a description of the Assessments for the courses marked 'A' in the above table.**

Assessment Cycle

We assess each course in the Fall semester of each year, analyze the results by February of each year, implement the improvements by April, and then disseminate the results in May.

Assessment Cycle Chart

Assessment Cycle for:

Computer Science Program College of Liberal Arts and Sciences

Analysis and Interpretation: February 1 of each year
Improvement Actions: Completed by April 1 of each year
Dissemination: Completed by May 1 of each year

Year / SLOs	12-13	13-14	14-15	15-16	16-17	17-19
Content Knowledge						
#1	x	x	x	x	x	x
#2	x	x	x	x	x	x
Critical Thinking						
#3	x	x	x	x	x	x
Communication						
#4	x	x	x	x	x	x

Methods and Procedures

SLO Assessment Matrix

The SLO Assessment Matrix is new for the 2012-13 Academic Assessment Plans. We have populated the matrix to the extent possible with the information we have available. Please complete the matrix.

Assessment Method - For each SLO, please enter the assessment method you are using – exam (course, internal, or external), project, paper, presentation, performance, etc.

Measurement – list the measurement procedure you use for this outcome. It can be a faculty-developed rubric with the minimum acceptable level identified, an exam score and the minimum passing score, or other measurement. **Required for 2012-13: Include at least one example of a rubric used to assess an SLO.**

SLO Assessment Matrix for 2012-13

2012-13 Student Learning Outcome	Assessment Method	Measurement Procedure
Apply knowledge of mathematics and science to computer science problems.	Student course performance, in exams and/or projects, as determined by course instructor and a faculty committee	Faculty-developed rubric: Likert scale (1-5, with 2 as minimal achievement of the SLO)
Design a computing system, component or process, analyzing and interpreting the data.	Student course performance, in exams and/or projects, as determined by course instructor and a faculty committee	Faculty-developed rubric: Likert scale (1-5, with 2 as minimal achievement of the SLO)
Use the techniques, skills and tools necessary for computer science practice.	Student course performance, in exams and/or projects, as determined by course instructor and a faculty committee	Faculty-developed rubric: Likert scale (1-5, with 2 as minimal achievement of the SLO)
Design a computing system, component or process to meet desired needs within realistic economic, environmental, social, political, ethical, and health and safety constraints.	Student course performance, in exams and/or projects, as determined by course instructor and a faculty committee	Faculty-developed rubric: Likert scale (1-5, with 2 as minimal achievement of the SLO)
Identify, formulate and solve computer science problems.	Student course performance, in exams and/or projects, as determined by course instructor and a faculty committee	Faculty-developed rubric: Likert scale (1-5, with 2 as minimal achievement of the SLO)
Communicate technical data and design information effectively in writing, in speech and in multidisciplinary teams to other computer scientists.	Student in-class Presentation	Rubric

Assessment of the Student Learning Outcomes is performed via direct and indirect assessments.

Direct Individual Student Assessments

The process for direct assessment of outcomes has three components, described below.

1. Qualitative evaluation and quantitative measurement by the instructor

The quantitative measurement of achievement of each outcome is assessed in a subset of the required courses in the program. This analysis is performed and reported by the instructor of each course in the form of the per-course Course Outcomes Assessment report. Each semester, the instructor of each course that is charged with assessing outcomes, completes one Course Outcomes Assessment Form for each outcome that is assessed in the course. The instructor establishes the instrument(s) to be used to assess each outcome. These are typically questions embedded in student assignments, exams, quizzes, or other evaluative mechanisms. In consultation with the course committee, the instructor also establishes the Likert-scale threshold(s), which maps the instrument's scale to the 1-5 Likert scale for achieving each outcome. The instructor also supplies the relevant statistics for the course. These include the number of students, the grading scale and the average score for the embedded question, the score required to minimally achieve the outcome (Likert 2), the percentage of students who achieved the outcome, and the average Likert-scale value. Finally, the instructor makes any relevant comments regarding the achievement of the outcome. In addition, the instructor of each course prepares a set of course materials, which includes the course syllabus, copies of the Course Outcomes Assessment Reports, copies of the instruments used to assess the outcomes, and sample graded student work. This information is stored by the respective Departments. These materials are the primary source of information for the next level of the assessment process, the Course Committee Report.

1.1. Sample Rubric.

Below is a sample rubric for SLO #4, Communication, to be assessed in CEN-3031 Software Engineering, and in CIS 4914 or EEL 4924C, Senior Design.

	Unsatisfactory(1)	**Satisfactory(2-3)**	**Adept (4)**	**Exemplary (5)**
1. Prepares a written report for a project in a organized and professional manner	Presents information to the audience in an unprofessional or disorganized format with many errors in grammar and spelling.	Presents information to the audience in an organized report, but with grammar and spelling errors.	Communicates meaning to the audience in an organized, professional report with very few grammar and spelling errors.	Skillfully communicates meaning to the audience with excellent organization, format, wording, and virtually error free grammar and spelling.
2. Presents findings orally to audiences in an effective way	Presents in a manner that is disorganized or obviously unrehearsed, with poor quality visual aids or voice projection.	Delivers an oral presentation with supporting materials, but needs more work to help audience understand key points.	Delivers an organized presentation with effective central message and supporting materials	Confidently delivers a memorable, organized, and polished presentation with effective central message and supporting materials.
3. Uses appropriate	Figures or tables in written and oral reports	Incorporates figures and	Incorporates figures and tables	Incorporates well labeled and organized

graphs or tables to display and interpret results	have major errors, and are difficult to read or understand.	tables into written or oral reports with some errors in presentation and marginal discussion.	into written or oral reports, and discusses their interpretation	figures and tables into written or oral reports, and fully discusses their interpretation for the audience.

2. Qualitative evaluation by the course committee

The qualitative evaluation of the achievement of all outcomes is assessed in each course. This evaluation is performed and reported by a course committee, consisting of at least three faculty members who are involved in either teaching the course or otherwise have expressed interest in it. This committee makes recommendations and suggestions for improvements in the course and its relation to other courses in the curriculum, improvements in the achievement of the outcomes, and improvements in the process itself. They produce the Course Committee Evaluation report containing their evaluations and recommendations. Each semester, the course committee is convened by the instructor of each course in which program outcomes are assessed.

Each committee is tasked with the following:

1. To evaluate the course in terms of its contents and its place within the curriculum,
2. To perform a qualitative analysis of the quantitative data in the Course Outcomes Assessment Report and course materials supplied by the instructor,
3. To examine, evaluate, and ratify the quantitative criteria used, the instruments chosen, the statistics provided, the Likert scale values used, and the sample student graded work, and
4. To generate suggestions/recommendations in three categories:
    a. Recommendations to future instructors,
    b. Recommendations to curriculum governance, and
    c. Recommendations on improvement of the process.

The course committee fills in one table per outcome assessed in the course, with evaluative comments on the instruments chosen, the statistics provided, the Likert scale values used, and the sample student graded work. The Course Committee Evaluation Reports are collected by the SLO Coordinator, for the third and final component of the program outcomes assessment process.

3. Overall analysis of the results

Overall analysis of the achievement of each outcome is performed across all courses in which it is assessed. This analysis is performed by the SLO Coordinator, who analyzes the reports produced by each individual course committee, collects (and generates further) recommendations for improvements at all levels, directs those recommendations to the proper governance bodies, and follows up on actions triggered by those recommendations. Once per semester, the SLO Coordinator collects the Course Committee Evaluation Reports from the courses that assess outcomes, and takes the “birds-eye” view of each outcome, examining the results and recommendations across all courses that assess that outcome. He also gathers any feedback from other, program-level indirect assessment mechanisms. The SLO Coordinator refers suggestions and recommendations to the Joint CEN Curriculum Committee for

consideration and/or action. The Coordinator is also charged with following up in subsequent semesters on such actions, and determining whether recommendations initiated earlier to address any shortcomings have engendered program improvements.

Indirect Student Assessments

Indirect assessments are carried out via student focus groups and student exit surveys. These assessments provide feedback on the entire Program, including its Mission and Student Learning Outcomes.

Student focus groups

Students meet with one or more faculty members to discuss the Program Mission, their attainment of the Student Learning Outcomes, and ideas for new courses or modifications to existing courses.

Student exit surveys

Students are asked to complete an exit survey before they graduate. Students are asked regarding their future employment or graduate school plans, their experience in courses at the University of Florida, the effectiveness of undergraduate advising, and their ideas for improving the program.

Assessment Oversight

Name	Department Affiliation	Email Address	Phone Number
Beverly Sanders	Computer and Information Science and Engineering (CISE)	sanders@cise.ufl.edu	505-1563
Douglas Dankel	CISE	ddd@cise.ufl.edu	505-1578