College of Arts and Sciences
Department of Scientific Computing | Computational Science Program
Table 3: Student Learning Outcomes

Doctorate in Computational Science

Outcome Type	Outcome	Assessment & Evaluation Process
Program Outcome: Cross-Disciplinary Training	A student that completes the doctoral program will possess cross- disciplinary training in scientific computing, mathematics and applied sciences, in preparation for entry into industry, government, or academics, in an era of fast-pace, changing technology. A student in the doctoral program is expected to complete a master thesis or project in passing, in which case, the doctoral thesis prospectus serves as a Masters project. A doctoral degree is distinguished by a requirement of originality. In additional to elements from the Master program, student training encompasses the major elements of research originality, writing papers, and exposition. Responsibility lies with Departmental faculty as a whole, and more specifically with the student's primary advisor and thesis committee.	We judge our success of our Doctoral program in two ways: 1) the enrollment trends into our program through positive feedback, 2) by the number of papers written by the doctoral students in reviewed journals, presented at conferences and appearing in conference proceedings. The above two criteria are measured and collected by our graduate committee on a yearly basis.
Learning Outcome: Research Competency	Upon completion of the course of instruction, the student will be able to create through research, new results and computational technologies for application to the solution of scientific and engineering problems.	Every year, students who desire to pursue their doctorate are required to pass our preliminary examination, given once a year. There are two components: a written exam and an oral exam. The written exam consists of 11 questions on our core courses (the students are graded on 10 questions, and must get a score of at least 70 out of 100 on 7 of the questions to pass), with a mixture of theory and computation. The oral exam concentrates on the sections of the written exam where the student underperformed, and on a specialized topic agreed to beforehand. The departmental faculty is responsible for creating the questions, evaluating the responses, and deciding on who passes, or not. All students must pass their preliminary exam prior to continuing their course of study. The capstone of the Ph.D. program is the dissertation for which it is required that students do original research that is publishable in appropriate venues such as scientific journals.
Learning Outcome: Research and Development	Upon completion of the doctoral program, the student will be able to perform research in broad areas of computational science including the design, analysis, implementation, and application of computer algorithms to problems in science and engineering.	As part of training for our graduate students, many of courses require that students complete a project, which includes coming up with a problem, formulating the problem, generating results, writing a report, and presenting the work at the end of class. The project often accounts for 20 to 40 percent of the total grade. Some courses that incorporate student projects include (in no particular order), ISC5228 (Monte- Carlo Methods), ISC5907 (Directed Individual Study), ISC5307 (Scientific Visualization), ISC5415

		(Computational Space Physics), and ISC3313 (Introduction to Scientific Computing). Some of our courses are more conventional with a laboratory component but no projects. Instead, laboratory reports have to be submitted and graded. These various courses assign from 20 to 40 percent of the class grade towards these projects. Projects can take the form of analysis and presentation of published research papers (ISC5907) or a step-by-step analysis of some application problem (Applied Computational Science I and II: ISC5315/ISC5316). Students get to improve their writing skills, analysis capability, independent research, and oral presentation.
Learning Outcome: Scientific Writing	Upon completion of the course of instruction, the student will be able to prepare a research paper, research proposal, and doctoral dissertation.	In the course of a doctoral degree, our students have multiple opportunities to exercise their written communication skills. These include: 1) writing a dissertation prospectus, typically one year after successfully passing a preliminary exam designed by our faculty to test the student's knowledge of course materials and their ability to conduct research; 2) writing a dissertation, including sections on problem presentation, section on previous work, details of research, results and benchmarks and results, directions for future research. A dissertation in the department must contain elements of modeling, mathematics, programming, and an applied discipline (the four elements that define computational science). All students must demonstrate these skills before their studies can proceed. Evaluation of the written materials is done primarily by the student's advisor and secondarily by the student's dissertation committee.
Learning Outcome: Admission to Candidacy	After the first year, the graduate doctoral student will demonstrate knowledge and research competency by taking and passing written (end of Spring) and oral (early fall) preliminary exams, which if successful will admit the student to the doctoral program.	Students who desire to pursue their doctorate are required to pass our preliminary examination, given once a year. There are two components: a written exam and an oral exam. The written exam consists of 11 questions on our core courses (the students are graded on 10 questions, and must receive a score of at least 70 out of 100 on 7 of the questions to pass), with a mixture of theory and computation. The oral exam concentrates on the sections of the written exam where the student underperformed, and on a specialized topic agreed to beforehand. The departmental faculty is responsible for creating the questions, evaluating the responses, and deciding on who passes, or not. All students must pass their preliminary exam to gain admission into the doctoral program. The capstone of the Ph.D. program is the dissertation, which presents original research conducted by the student, publishable in appropriate in appropriate peer-reviewed venues such as scientific journals or presented at scientific conferences.

Source: FSU Institutional Effectiveness Portal, 2018-19.

Masters in Computational Science

Outcome Type	Outcome	Assessment & Evaluation Process
Program Outcome: Training and Employment	By the end of the year, the program produces a slate of students with enhanced cross-disciplinary training in the scientific computing, mathematics and applied sciences, in preparation for entry into industrial or government labs in an era of fast-paced, changing technology or for admission to Ph.D. programs. Student training encompasses the major elements of STEM (Science, Technology, Engineering, and Mathematics). Responsibility lies with Departmental faculty as a whole. Our measure of success is gauged through continued enrollment, positive feedback, and good job postings.	We judge our success of our Masters program in two ways: 1) the enrollment trends and positive feedback from students in our program, and 2) by the ease with which our students find employment in industry and governmental positions or are admitted to Ph.D. programs in other institutions. The above two criteria are collected on a yearly basis. The information is collected by the Graduate Committee.
Learning Outcome: Research and Development	Upon completion of the course of instruction, the student will be able to perform research and development activities, often in a team environment, requiring knowledge and skill in computational approaches to solving problems in science and engineering.	As part of training for our graduate students, many of courses require that students complete a project, which includes coming up with a problem, formulating the problem, generating results, writing a report, and presenting the work at the end of class. The project often accounts for 20 to 40 percent of the total grade. Some courses that incorporate student projects include (in no particular order), ISC5228 (Monte- Carlo Methods), ISC5907 (Directed Individual Study), ISC5307 (Scientific Visualization), ISC5415 (Computational Space Physics), and ISC3313 (Introduction to Scientific Computing). Some of our courses are more conventional with a laboratory component that require that a final report be submitted and graded. These various courses assign from 20 to 40 percent of the class grade towards these projects. Projects take the form of analysis and presentation of published research papers (ISC5307) or a step-by-step analysis of some application problem (ISC5315). Students get to improve their writing skills, analysis capability, independent research, and oral presentation. All students are expected to perform satisfactorily on these tasks. We judge the research and development at the Masters level to be effective if all students achieve at least a B+ in the courses that are project based.
Learning Outcome: Master Competency Skills	Upon completion of the course of instruction, the student will be able to organize a research project, survey previous work done in the research area, write up and present the results, leading to a written document and an oral presentation of either a Master project or thesis.	The Master's thesis consists of an extended research project, one to two years in duration, to be completed and defended within two years of joining the Master's program. It is supervised by a Masters committee of three departmental faculty. The Masters thesis is a written compilation of the student's research, employing the skills and knowledge derived from the courses taken in the departmental program. These skills will translate to employment in almost any field dependent on computers and software development. The thesis presentation is public, while the associated written document is evaluated by the student committee. Students can instead choose the Master's project option which requires two additional courses and the presentation of a written and oral report about the project. The project, does not require the same

		level of originality as does the theses, but still requires a mastery of research skills. These skills will translate to employment in almost any field dependent on computers and software development. We measure success by the number of students who successfully defend their master project or thesis.

Source: FSU Institutional Effectiveness Portal, 2018-19.

Bachelors in Scientific Computing

Outcome Type	Outcome	Assessment & Evaluation Process
Program Outcome: First Year Students	By the end of the year, the program will produce a group of undergraduate students who have successfully completed their first year, typically as a junior.	We measure success through course grading. Students passing all our courses with a grade of 70% and above pass the course. If a student passes all our courses, we consider the student to have successfully completed his/her first year. We do not control the student's work in his/her liberal art studies.
Learning Outcome: Employment	Upon completion of the course of instruction, the student will be able to demonstrate adequate preparation for employment in industry, laboratory, and government jobs and for graduate computational science and other degree programs.	Through the totality of their courses, which include programming, numerical algorithmic components, algorithm applications in the sciences, discrete and continuous, and a practicum (to provide research experience), students will automatically be trained for a programming or computer modeling job in a national laboratory or in any institution requiring practical knowledge of computers and how they are used to solve problems. Since most disciplines currently depend on computer technology, and programming is becoming an increasing component of our society, our undergraduate student education provides good employment (for those students choosing not to continue with their graduate programs). It is expected that all our graduates will continue their studies or find employment.
Learning Outcome: Application of Algorithms	Upon completion of the course of instruction, the student will be able to demonstrate broad knowledge of the application of algorithms to problems in the sciences and of the interpretation of computational results through visualization and other techniques. The two courses most related to algorithm applications are ISC4223 (Computational Methods for Discrete Problems) and ISC4232 (Computational Methods for Continuous Problems). These two courses build on ISC4220 and ISC4221 (Continuous and Discrete Algorithm for Scientific Applications). All four courses are four credits each: 3 credits of lectures and a 1 credit lab component to provide the students with the opportunity to put into practice the methods learned during the lectures. We assess whether students have mastered these courses by requiring that 70 percent of our students achieve B+ and above.	The grades for ISC4223 and ISC4232 are determined by exams constructed by the instructor and the lab reports graded by either instructor or TA (if assigned). Labs are created either by the instructor or together with the TA. Often the labs will be identical to those created in previous years. Most labs lead to a written report, which answers the questions posed in a descriptive manner. This also provides students with practice in writing. All students are expected to demonstrate this competency successfully.
Learning Outcome: Discrete and Continuous Algorithms	Upon completion of the course of instruction, the student will be able to demonstrate broad knowledge of the basic continuous and discrete algorithms of scientific computing. This knowledge comes primarily from the classes Algorithms for Science Applications I (ISC4220), and Algorithms for Science Applications II (ISC4221).	ISC4220 and ISC4221 are required classes of all undergraduate students in our program. They are four credits each: 3 credits of instruction and one credit of lab. The labs, run weekly, provide the students with practical hands-on experience with writing and executing a variety of computer algorithms, both discrete and continuous, as applied to the various scientific disciplines. The labs are conducted by either the faculty member or a teaching assistant (TA). Students write lab reports, graded by either the instructor or the TA. The class has a midterm and final exam, constructed and administered by the instructor.

		We assess performance through grading. Our aim is that 70 percent of our students get a grade of B and above.
Learning Outcome: Programming Techniques	Upon completion of the course of instruction, the student will be able to demonstrate broad knowledge of the programming techniques needed to implement algorithms for scientific computing.	The following courses serve to teach students the broad range of programming techniques necessary for algorithm implementation: ISC 3313 (Introduction to Scientific Computing, 3 credits, given each semester (including summer), and rotating between C++, Fortran, and Java, with Python during the summer), ISC4304 (Programming Scientific Applications, 4 credits, given every year), ISC3222 (Symbolic and Numerical Computations, 3 credits, given every year). Each course aims to teach the students some aspect of programming. The courses cover elements Java, Python, C++, Fortran, Matlab and a symbolic programming language. All the courses provide ample illustrations in a range of applied sciences. Exams and quizzes (written by each instructor) evaluate the student's progress. We expect that students who complete all the lab reports, and all the homework, will get a grade of 70/100 or above. The exercises leave lots of room for independent thinking, and thus cannot be scored in a deterministic manner.
Learning Outcome: Research Experience	Upon completion of the course of instruction, the student will be able to demonstrate research experience in the design and application of computational algorithms to a scientific application and/or research experience in the design and analysis of computational algorithms across scientific applications. This experience is provided by a practicum, a 3 credit course through which a student works on a project under the supervision of one of our faculty, a faculty from another department, or a researcher outside FSU.	The practicum occurs in the senior year. All students are expected to successfully complete the practicum.
Learning Outcome: New Student Learning Outcome	Upon completion of the course of instruction, the student will be able to demonstrate specialized knowledge in a natural or mathematical science discipline as it relates to the design of scientific computing algorithms. A major tenet of Scientific Computing is to combine computer programming and mathematics with knowledge in at least one science discipline. Students achieve this knowledge by taking at least 18 credits (typically 3 courses in our department and 3 courses in outside departments) Courses might include (within our department) visualization, Verification and Validation, Multiscale Methods, Genomics, and Molecular Dynamics, or courses offered in other departments (assuming any prerequisites are satisfied) in the areas of biology, physics, chemistry, etc.. Usually, these courses have a computational component, but not always. Each course is the responsibility of its instructor, who creates exams, homeworks, and/or quizzes.	Assessment is through homeworks, exams and class participation. Each instructor is responsible for creating his/her homeworks and exams, taking as a starting point, those of previous years. The goal is to have at least the same results as the previous year, if not an increase in student achievement, but not at the expense of learning.

Learning Outcome: Solving Problems	Upon completion of the course of instruction, the student will be able to demonstrate understanding of the connections between computational, theoretical, and experimental approaches to solving problems in the sciences.	All our courses are designed to cover a range of topics whether programming to solve problems, algorithms (which must be converted into problems), and applications (what are the algorithms used for). The various courses mix these three elements to various degrees. On occasion, we also cover experimental techniques, which go hand in hand with computation and experiment. Courses that cover experimental approaches are usually part of courses that students take in their specialized discipline, for example in GFDI. Some of these courses are offered in our department as specialized elective courses, and others are taken in other departments. The assessment for outside courses follows that of the other departments. Sometimes students might take a course that covers experimental techniques. In that case, the student will develop a set of skills related to model building and implementation. Assessment is through grading of projects in these various classes.

Source: FSU Institutional Effectiveness Portal, 2018-19.