CSC 100 Class Information and Syllabus

Instructor: Stephen R. Tate (Steve)
Lectures: Mon/Wed 10:00–10:50, Petty 223
Lab: Fri 10:00-11:50, Petty 222
Office: Petty 166
Office Hours: Mon/Wed 11:00–12:30, or by appointment
Phone: 336-256-1033
E-mail: `srtate@uncg.edu`

Class Web Page: `http://www.uncg.edu/cmp/faculty/srtate/100/`

Prerequisites: None.

Catalog Description: A broad-based introduction to key concepts and principles of computer science. Exploration of seven big ideas of computing: creativity, abstraction, data, algorithms, programming, the Internet, and impact of computing.

Who This Course is For: This course is for two types of students: non-majors who want to learn more about computing and computer science, and as an optional pre-major course for computer science majors who are just beginning and haven't taken the traditional first computer science class (CSC 130). The course is designed so that students build problem solving and critical thinking skills related to computing (i.e., "computational thinking" skills) that are becoming vital 21st century skills across a wide variety of interests and disciplines. For beginning computer science majors, the foundational skills built in this course are designed to improve success in later computer science courses.

Full Course Description: This course is part of a national initiative to create a new course covering computer science principles that can be offered to a broad audience and can provide a consistent experience for students at both the university level and the high school level through an AP class. The "CS Principles Project" (see http://www.csprinciples.org) is sponsored by the College Board and the National Science Foundation, with partners and advisors from universities and high schools. The CS Principles Project was initiated with the following description of their goals:

> *Computer Science: Principles* is designed to introduce students to the central ideas of computer science, to instill ideas and practices of computational thinking, and to have students engage in activities that show how computing changes the world.

> The course is rigorous and rich in computational content, includes computational and critical thinking skills, and engages students in the creative aspects of the field. Through both its content and pedagogy, this course aims to appeal to a broad audience.

This is not a "computer literacy" class or a "computer programming" class, although at the end students will have a better understanding of how computers work and will have gained some experience in computer programming. Rather, this class is about how to think about problems and solutions in a world that is increasingly data-rich and reliant on computing tools and techniques.

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

1. use programming and computing tools in creative expression, creating and analyzing artifacts (creativity);
2. devise and use multiple levels of abstraction in computation, including the use of models and simulations to raise and answer questions (abstraction);
3. use computing to examine large datasets and facilitate exploration in order to gain insight and knowledge (data and information);
4. devise, express, and evaluate algorithms for solving computational problems (algorithms);
5. create and evaluate a correct program to implement an algorithm, using abstraction to manage complexity (programming);
6. explain characteristics of the Internet and systems built on it, including issues of cyber-security and privacy (Internet);
7. analyze beneficial and harmful effects of computing, connecting computing to economic, social, and cultural contexts and innovations in other fields (impact).

Textbook and Readings: This class is organized around readings and lab exercises that are distributed through the class web page. We will regularly discuss readings from the following book, which is available online for free, or you can purchase a hardcopy if you prefer:

> Hal Abelson, Ken Ledeen, and Harry Lewis. *Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion.* Addison-Wesley, 2008 (online version available at http://www.bitsbook.com).

Teaching Methods and Assignments: Students are expected to participate responsibly in course activities, including class and lab sessions, assignments, and enrichment activities. These are described in more detail below.

Attendance/Participation: This class will meet for two lectures and one mixed lecture/lab per week, and students are expected to be in class on time. Students should prepare for class by completing assigned readings, and are expected to participate in class discussions. Students must also attend at least two enrichment events during the semester — these events are generally held on Fridays at 3:00, and can include visits from computing professionals, discussions with upper-level computer science students, and introductions to campus resources. A schedule of specific events can be accessed from the class web page. If it is impossible for you to attend these events due to scheduling conflicts, you must make arrangements with the instructor for alternative enrichment assignments well in advance (requests for alternative assignments can not be made after October 17).

Exams: The mid-term and final exam will cover topics discussed in lecture and emphasized through lab exercises and homeworks. Questions will cover all 7 of the defined student learning outcomes.

Readings/Discussion: Students will explore the impact of technology on society through a series of readings from the book *Blown to Bits*, and possibly other sources. Each major reading will take place during a two-week timeframe: in the first week, students are expected to read the material and answer "reading reflection" questions in Blackboard; during the second week students will participate in an on-line discussion of the reading with classmates and the instructor, where everyone is expected to contribute to the discussion. At the end of the on-line discussion period we will have a brief in-class wrap-up discussion.

Lab Exercises: Lab exercises involve programming in various languages/systems and utilizing a variety of computational tools. Student work products include programs that they create, or modifications of programs provided by the instructor. Each lab exercise contains some pre-lab reading (which students should read *before* coming to lab), some self-test questions to check understanding of the pre-lab reading, the actual activities to perform in the lab, and an explanation of how the lab exercise and concepts fit into the bigger picture of computer science. Lab activities will utilize *pair programming*, where pairs of students work together on a single computer, taking turns as the *driver* and the *navigator*. At the end of the lab, students will individually take an on-line quiz to test how well they have mastered the concepts in that lab. Exercises are concrete demonstrations of topics being discussed, and each lab session will start with a mini-lecture on relevant topics and an opportunity for students to ask questions about the pre-lab reading. The final lab sessions will be dedicated to guided work on student projects. Lab exercises are available on the class web site, both under "Lab Exercises" and linked from the class schedule.

Homeworks: Homeworks consist of independent student work that is less guided than lab work. Homeworks may involve programming, independent research of some topic, or written

exercises and analysis. Homeworks often include open-ended aspects in which students can demonstrate their own particular creativity. Students will typically have two weeks to complete homework. No homeworks will be given in the last month of the semester, when students will be working on their projects.

Project: Students will work in teams of 2 or 3 people to produce some creative software product (an animation, game, tool, etc.), and will work on this project during the final month of the semester. At the end of the semester there will be a "project showcase" in which students present and demonstrate their projects for the class. Students will vote on the projects, and the winning project will receive a "Student's Choice Award."

Evaluation and Grading: Each student work product will be graded, and the student's final grade will be determined by assigning each category of work a weighted score according to the following distribution:

Lab Exercises (lowest dropped)	20%
Homeworks and Readings	20%
Project	20%
Mid-term Exam	15%
Final Exam	20%
Participation	5%

Professionalism and Work Quality: While the objectives of this class deal with technical issues, students should at all times produce high quality, neat, and grammatically correct work. Spelling and grammar reflect substantially on your seriousness and professionalism, and sloppy or grammatically incorrect work will result in up to 10% being taken off the grade, even if all technical information is correct.

The final weighted average will be assigned a letter grade according to the following table (exact borderline cases will be given the higher grade – for example, a student with a weighted average of 90 will earn an A-).

97 – 100: A+	93 – 97: A	90 – 93: A-
87 – 90: B+	83 – 87: B	80 – 83: B-
77 – 80: C+	73 – 77: C	70 – 73: C-
67 – 70: D+	63 – 67: D	60 – 63: D-
Below 60: F		

Topic Outline and Calendar: For the current and planned list of topics and dates, please see the "Schedule" area of the class web site.

Academic Integrity: Each student is required to abide by the UNCG Academic Integrity Policy on all work in the course. Refer to http://academicintegrity.uncg.edu or the UNCG

Undergraduate Bulletin for more information. Some assignments and exercises may involve teamwork, pair programming, or group discussion. When portions of these assignments or exercises are to be done individually, they will be clearly marked as such, and work turned in should be entirely your own. On all work that you turn in, work or ideas that are not your own should be clearly marked and a citation should be given (a list of team members or collaborators, external references, etc.).

Attendance Policy: Attendance will be taken, and students may be dropped from the course if they have more than three unexcused absences. The university allows for a limited number of excused absences for religious observances – students who plan to take such an absence should notify the instructor at least two weeks in advance so that accommodations can be made (also see below for "late work and makeup exams"). It is the student's responsibility to obtain notes from another student if they miss class.

Late Policy and Makeup Exams: Assigned work is due at the beginning of class on the due date, and will not be accepted late. Exams, labs, and in-class work must be done on the scheduled day, in class. In cases of serious medical situations (which must be documented with a doctor's excuse), some exceptions can be made. Students with planned absences, whether for university events, religious observance, or other reason, are expected to make arrangements with the instructor to turn in assignments or take exams before the scheduled date of the assignment or test.

Final Exam: The final exam will be held according to the official UNCG Final Exam Schedule, which lists the time for this exam as Wednesday, December 3, 12:00 – 3:00.

ADA Statement: UNCG seeks to comply fully with the Americans with Disabilities Act (ADA). Students requesting accommodations based on a disability must be registered with the Office of Accessibility Resources and Services located in 215 Elliott University Center: (336) 334–5440 (or `http://ods.uncg.edu`).

University Closings: If university facilities are closed due to flu outbreak or other emergencies, it does not mean that classes are canceled. In such an event, please check the class web page and Blackboard site for information about if and how the class will proceed.

Commercial note-taking services: Selling class notes for commercial gain or purchasing such class notes in this or any other course at UNCG is a violation of the University's Copyright Policy and of the Student Code of Conduct. Sharing notes for studying purposes, or borrowing notes to make up for absences, without commercial gain, are not violations.