

## CSI 201 — Computer Science I

**Instructor:** Dr. Shaun D. Ramsey

**Email:** sramsey2@washcoll.edu

**Phone:** (410) 810-7485

**Office:** Goldstein 213

**Office Hours:** MW 10:30am–11:20am

Tu 11:30-12:45pm or by appointment

**Class meetings:** GLDS 201 MWF 09:30AM-10:20AM

**Text:** *Absolute C++, Second Edition*, by Walter Savitch

**Overview:** The purpose of this course is to teach you basic programming techniques and algorithmic problem solving methods. In particular, we will be focusing on the use of the C++ programming language. A core foundation is necessary to create complex, efficient and otherwise useful programs designed to solve specific needs and problems. Good algorithmic problem solving methods and programming skills are a valuable commodity and a marketable skill in most professional positions and disciplines. This course is a challenging course in the quantitative discipline. It requires good logic and problem solving skills for success.

**Topics:** The topics covered in this course include the basic tools for programming in C++. You will learn about variables, input/output, conditionals, iterators, functions and recursion. Other topics in the course include arrays, strings and classes.

**Advising:** CSI 201 has no prerequisites and is a required course for the major in computer science. This *is* a very difficult course for those with weak problem solving skills.

**Exams:** The exams will be held on February 22nd, March 31st, and April 28th. The final exam will be administered during its scheduled slot during final exam week. An absence on the day of an exam will result in a grade of a 0. Except in cases of extreme emergency, exams must be taken on the day the exam is administered. Before a make-up test is scheduled, documentation of the extreme emergency must be given. Make-up exams for tests missed due to an extreme emergency will be arranged for a time that is mutually convenient for the student and Dr. Ramsey.

**Grading:** Your grade consists of three exams, one final, homework, classwork, and oral presentations. You must pass each of these six sections to pass the course. Each exam is worth 15% of your final grade. The final exam is 20%. Homework contributes 15% of your final grade. Classwork comprises 10% of your grade in this course. Classwork will contain programs completed during class as well as weekly quizzes based on previous material and reading material. Oral presentations cover the last 10% of your grade. For programming assignments, credit is given only for programs which compile without errors, execute properly, and are not late. Late homework will not be accepted. Homework is due at the beginning of class on the assigned due date and in many cases is time stamped by the server we will be using.

**Attendance:** Attendance will be taken at the beginning of every class. After two weeks of missed absences you fail the course. You fail the course on your fifth absence in a TTh course and on your seventh absence in a MWF course. There is no distinction between excused and unexcused absences. I will likely email you if you miss a class, but it is ultimately your duty to keep track of your absences. Note that missing a class may also result in missed classwork or quizzes. There are no make-up quizzes. It is your responsibility to obtain assigned homework, announcements and class notes from that class. It is important that you attend every class. As a matter of courtesy, students are expected to inform Dr. Ramsey before class describing the reason for any absence.

**Academic Honesty:** You are always subject to the Honor Code of Washington College. You may discuss concepts with others, but work is to be done on your own (unless otherwise designated). If you are unsure if something is considered *cheating*, simply ask. As always, if you have questions, feel free to email or stop by my office.

**Accommodations:** If you have a special accommodation/need that has been reported to the college, please let me know discretely as soon as possible, so that I can work to meet your accommodation.

**Suggestions:** To become a good programmer and problem solver, you must work on many problems. If you need help, please see me, peers or the math center for assistance. There are many resources, so there is no excuse for not using them.