Lectures: 3:15pm–4:30pm (Section 1) and 12:25pm–1:40pm T/R (Section 2), PACCAR 107

Instructor: Shawn Bowers, bowers@gonzaga.edu

Office Hours: 11:00am–12:00pm T/W/R, 1:00pm–2:00pm W, or by appointment.

Course Webpage: www.cs.gonzaga.edu/bowers/courses/cpsc223

Course Description: This course covers topics related to data structures, algorithms, and C++ programming. Topics related to data structures include hash tables, binary search trees, balanced binary trees, and heaps. Topics related to algorithms include sorting and searching as well as basic ideas in complexity analysis. Finally, topics related to C++ include review of abstract classes, inheritance, templates, and operator overloading as appropriate. This course is a required course for many upper division classes.

Prerequisites: CPSC 122 and MATH 231 (can also be taken in the same semester as CPSC 223).

Access to Course Materials: There is no textbook for this course. Some homework assignments may have readings, which will be provided by the instructor. GitHub will be used for homework submission and homework feedback. Lecture notes, homework, and a weekly schedule will be made available on the course webpage. Blackboard (learn.gonzaga.edu) will be used to post grades. Piazza (piazza.com/gonzaga/fall2021/cpsc223) will be used for questions and discussions. Access to a computer capable of running the department virtual machine (see www.cs.gonzaga.edu/virtual) is required. The virtual machine is also available in the CS computer labs.

Grading:

- 10% – In-Class Participation
- 60% – Homework Assignments
- 5% – Weekly Quizzes
- 5% – Midterm Exam
- 20% – Final Report (7.5%) and Exam (12.5%)

There will be approximately 10 homework assignments, weekly quizzes and exercises, one written midterm exam, and a written final exam.

IMPORTANT: To pass this class you must average 60% or higher on homework assignments and average 60% or higher on exams. For example, if you average over 60% on homework assignments, but less than 60% on exams, you will not receive a passing class grade.


Course Policies:

Student Expectations: You are responsible for understanding and learning the course material. If you do not understand topics discussed in class, or instructions on tests or assignments, it is your responsibility to ask for help from the instructor. You can get help from the instructor during office hours, via email, or using the Piazza system set up for the course. Please start your assignments early to leave yourself enough time to ask questions and to complete the assignment once your questions are answered.

Assignment Grading: All work for the course must be turned in on or before the day the assignment is due for full credit. The late policy and resubmission policy for homework assignments is given below.

Homework Late Policy: If you are unable to submit an assignment by the due date, you can still turn your work in with a 25% late penalty up to 2 weeks after the due date. After 2 weeks, the late work will not be accepted. Extensions may be granted for exceptional situations (such as prolonged illness) by contacting the instructor. No late work will be accepted after the end of the Thanksgiving break.

Homework Resubmission Policy: If you score less than 75% on a homework assignment, you can fix and resubmit
your homework to improve your score. The maximum score that can be achieved through resubmission is 75%. You cannot resubmit your homework if it was turned in late. Only “legitimate” submissions can be resubmitted, which includes attempting a significant portion of the assignment, completed test cases, and code that compiles. No resubmissions will be accepted after the end of the Thanksgiving break.

**Exams and Quizzes:** All exams and quizzes are to be done individually. Clear cases of collaboration will result in a grade of 0 on the exam or quiz and/or an F in the course and possible suspension from the University. Students with testing accomodations must contact the instructor to arrange alternative testing times as needed. Quizzes must be taken when given and cannot be made up.

**Attendance:** It is important that you attend class and keep up with course content and assignments. If you become ill and/or cannot attend classes due to another reason beyond your control, contact the instructor as soon as possible to make arrangements. Attendance will be one factor used to determine your class participation grade. There is currently no plan to create or distribute video recordings of class lectures.

**Academic Honesty:** You are expected to follow the University’s policy on academic honesty. Please see the policy on the University’s webpage for more information, including procedures for violations. If you are unclear about the policy or how it applies to this class please ask the instructor.

**Office Hours:** You are strongly encouraged to take advantage of office hours or make an appointment to meet with the instructor if you have questions about the course material. I am more than happy to help you, and office hours are a great way to ask questions and get one-on-one help with the material. At this time, all office hours will be in-person unless otherwise specified.

**Grades of Incomplete:** University Policy states that incomplete grades can be “Given when a student with a legitimate reason as determined by the instructor, does not complete all the work of the course within the semester that he/she is registered for the course.” A grade of incomplete is given to students who find themselves in situations beyond their control and that make academic success near to impossible. The Center for Cura Personalis and Academic Advising & Assistance are available to help in such situations. Note that a grade of incomplete will not be granted for students due to a heavy course workload or because they have fallen behind in their coursework due to inadequate time management.

**ABET Specific Outcomes of Instruction:** Students completing the course will:
1. Design complex programs by the implementation of data abstraction.
2. Analyze the runtime of both iterative and recursive algorithms using O-notation.
3. Understand and use recursive programming constructs.
4. Understand and analyze \( n^2 \) and \( n \log n \) sorting algorithms.
5. Understand the construction and analyze the use of nonlinear data structures and balanced trees.
6. Implement, use, and analyze algorithms with hash tables.

**University Academic Policies & Procedures:** A full list of the academic policies and procedures can be found at: www.gonzaga.edu/academics/academic-calendar-resources/registrars-office/policies-procedures/academic-policies-procedures. Note that new policies are added and existing policies are modified frequently.