Syllabus
COSC 160-20 Data Structures (Summer 2021)

Instructor: Philip Sheridan Buffum, Ph.D.
Assistant Teaching Professor
Department of Computer Science

Email: Philip.Buffum@georgetown.edu

Office Hours: See Canvas

TAs: See Canvas

This course is designed as a second year course for majors and minors and covers basic data structures and algorithm analysis. Starting with the art and science of analyzing algorithms, the main goal of this course is to learn various techniques for organizing data so that computer programs can access, modify, and delete data efficiently. Topics covered include basic data structures (for example, lists, stacks and queues), trees, hashing, heaps, disjoint sets, and graphs, self-adjusting data structures; worst-case, average-case, and amortized analysis; and basic problem solving techniques. The topics are theoretical in nature but have dramatic impact in practice.

The assignments in this course are open-ended and require students to design software solutions using appropriate data structures. Do not under estimate time required to develop software.

Credits: 3

Prerequisites: COSC-052 and (COSC-030 or MATH-200)

References:

- *Data Structures and Algorithms in C++*, Second Edition (2011), Goodrich, Tamassia and Mount
- *Introduction to Algorithms*, 3rd Edition (2009), Cormen, Leiserson, Rivest and Stein

Course Grade: The letter grade assigned for the course is based on total points out of 100 and is as follows:

A	[94, 100]
A-	[90, 94)
B+	[87, 90)
B	[84, 87)
B-	[80, 84)
C+	[77, 80)
C	[71, 77)
C-	[67, 71)
D	[60, 67)
F	[0, 60)

Grading:

	Percent of Final Grade
1 Midterm Exam	20%
1 Final Exam	20%
Projects	40%
Small Assignments	10%
Lecture Quizzes/Participation	10%

Exams: There will be one midterm exam. The specific date for this midterm will be posted on Canvas. The final exam, which is cumulative, will be administered during the finals period; the time can be found on the registrar's website (https://registrar.georgetown.edu/schedules/finals).

Make sure to note the following exam policies for this course:

- No make-up exams will be provided.
- If you foresee having trouble in taking an exam, you must tell the instructor well in advance - two weeks before the exam.

Projects: There will be several projects throughout the semester, involving both developing programs and writing reports. Note that projects are an integral part of this course! It is assumed that you already have a proficient understanding of a programming language. Specifically, it is expected that students will complete projects using C++ (either C++14 or C++17 are acceptable). Students are responsible for learning and/or reviewing, as needed, the programming language. Following general good practice, you must thoroughly check for correctness each programming assignment before submitting it for a grade, running it on your test data ***on the course server***. Your project will be graded by running it on the course server, so if it does not work there, it cannot be graded. You will need to resubmit with a 10% penalty. You must also appropriately document your programming assignments. Please refer to the following coding style guides:

- https://tinyurl.com/y8vnpgez
- https://tinyurl.com/fvrvv

Small Assignments/Quizzes: For homework assignments, it is recommended that you type your answers (e.g., in a text editor). The teaching staff uses the following rule: "If the student's answer can't be read and understood then it must be wrong". In-class assignments and quizzes will often require a computer, so bring your laptop to each lecture.

Late Policy: All assignments (projects and homework) will be posted on Canvas, along with their due dates. All electronic submission requirements (source code, reports, conclusions, etc.) should be uploaded to Canvas prior to the due date and time. Late submissions incur a 10% penalty per day up to 2 days late; any submission more than 2 days late will earn a 0 grade. So for an assignment submitted between 1 minute late and 24:00 hours late, the pre-penalty grade will be multiplied by 0.9 ; for an

assignment submitted between 24:01 and 48:00 hours late, the pre-penalty grade will be multiplied by 0.8; assignments submitted 48:01 hours late or more will earn a 0 grade. You are responsible for thoroughly reading the project description and ensuring that your submission has the correct file formats and naming conventions, as specified in the project description.

Academic Honesty and Expectations: I am required to report any suspicion of academic dishonesty to the Honor Council.

- Exams: must be entirely your own work. During exams, you are not allowed to view any other students work, show any other student your work, or engage in any discussion. Exams will be closed book and closed notes unless otherwise specified.
- Project submissions: Discussion among students pertaining to project content and general methodology is allowed; however, students are NOT ALLOWED to share code or write code/solution for others. Students are NOT ALLOWED to copy code or use code of others without an explicit disclosure.
    - Note: Although the use of code that is not completely of your design is permitted (with appropriate disclosure), you will not receive (full) credit for the corresponding portion of your submission.
    - Note: A student may be asked to present and explain a project submission at any time, without notice. At the instructor's discretion, a student's project grade can be adjusted based on this presentation, demonstration, and/or explanation. If a student does not sufficiently understand or explain their submission, further action may be taken.
- ***Any violation of these policies can result in a grade of F for the course,*** in addition to the report to the Honor Council.

Other Notes: The specific course schedule will be provided on Canvas. Course topics, administrative guidelines, and other specifics discussed in this syllabus are subject to change. Notice of any changes will be provided in class or on Canvas. Feel free to ask questions in the class or during regular office hours of the instructor/TA. If you cannot meet during regular office hours, then set up an appointment.

Tentative Schedule of Topics:

- Linear ADTs and Data Structures
- Algorithm Analysis and Asymptotic Notation
- Trees, Binary Trees and Binary Search Trees
- Balanced Trees
- Multi-way Trees and B-Trees
- Heaps and Priority Queues
- Maps and Hash Tables
- Sorting Algorithms
- Greedy Algorithms
- Graphs