



CS 7B - Section 3324 - Fall '17

Computer Science II

MW 2:00-5:05 in SOC 15



Instructor: Geoff Hagopian

ghagopian@collegeofthedesert.edu

geofhagopian.net

Office: Math Building RM 12 (NE corner)

Office Hours: MTWR 9:30-11

(760) 776-7223

This syllabus is the alpha version of an open source platform.

Course Description: This second course in computer science introduces more advanced topics in programming. Students will use modularity to develop solutions for larger-scale programming problems. Recursion, file processing, and object-oriented programming are implemented. This course will be taught using the C++ programming language.

Prerequisite: CS 007A with a C or better and MATH 012 (precalculus) with a C or better.

Note: A minimum grade of C is required in this course to progress to CS 9.

Credit Hours: 4

Texts: *Starting Out With C++, Early Objects*, by Gaddis, Walters, Muganda, ISBN13: 978-0-13-440024-2 and *Programming Principles and Practice Using C++* (2nd ed). ISBN: 978-0-321-99278-9, by Bjarne Stroustrup (Addison-Wesley, 2014) also recommended: *Theory and Problems of Fundamentals of Computing with C++* ISBN: 0-07-030868-3, by John R. Hubbard in the Schaum's Outlines Series.

“Learning Outcomes” Upon completion of this course, a student will be able to:

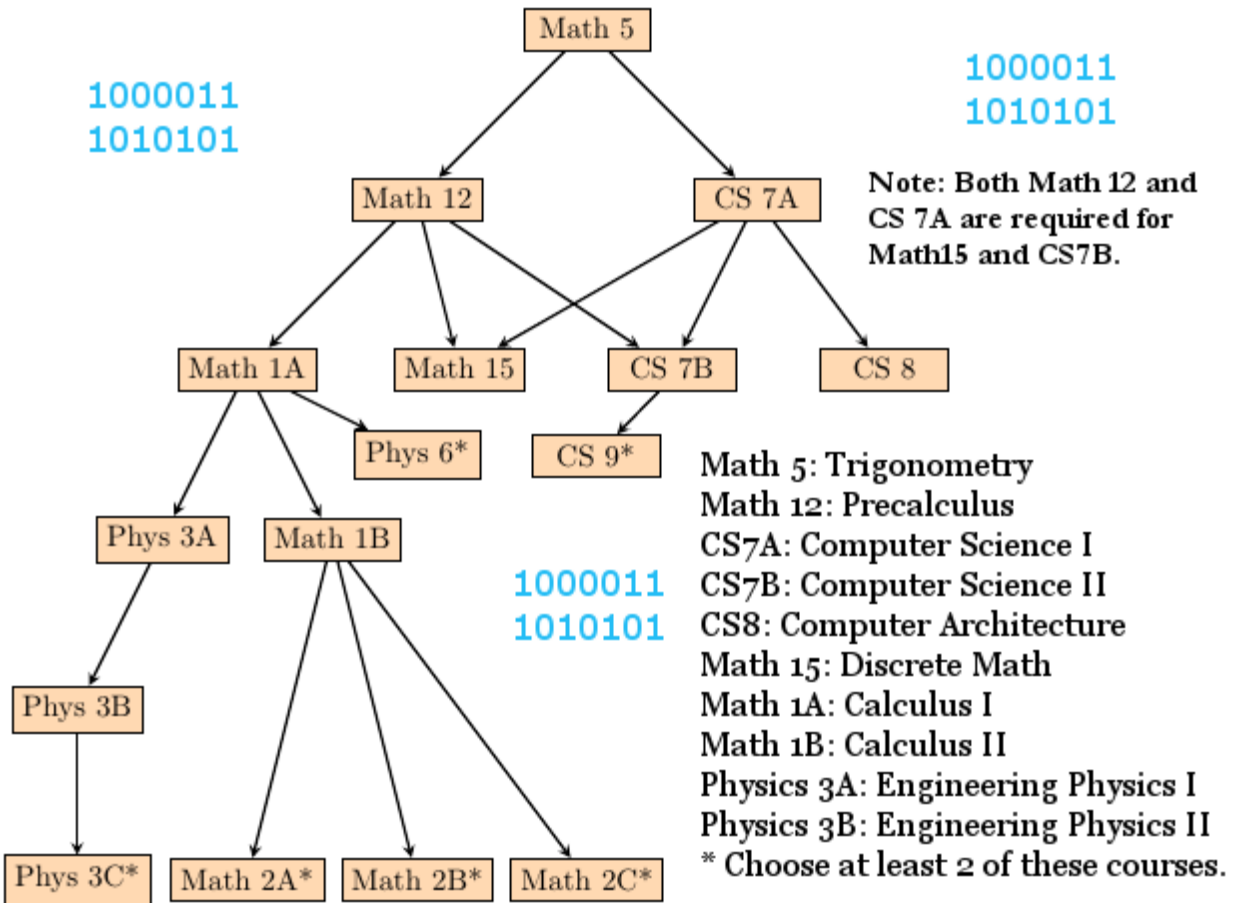
- Create programs which use standard C++ language features, including functions, arrays, arrays of arrays, pointers, pointer arithmetic, dynamic memory allocation, and structured data (structs).
- Design and implement an abstract data type using a class with member variables, member functions, constructors, and a destructor.
- Design and implement modular programs, created in appropriate .h and .cpp files, that use multiple classes with inheritance relationships, friend functions, friend classes, and operator overloading, using modern C++ language features and STL vectors and iterators where appropriate.

Course Objectives:

At the completion of this course, students will be able to:

1. Develop skills in the design and development of computer software continuing to utilize an object-oriented language, packages, modules and libraries.
2. Demonstrate an understanding of and use techniques of inheritance and polymorphism.
3. Select appropriate data structures from the Standard Template Library such as vectors, linked lists, stacks and queues in the design computer programs to solve complex problems from math and science.
4. Create classes which implement dynamic memory allocation techniques appropriate to the design of computer programs.
5. Implement advanced file reading and writing methods.

Computer Science Program Prerequisites Map



Grade Distribution:

Projects	50%
Midterms	25%
Final Exam	25%

Letter Grade Distribution:

≥ 90.00	A	70.00 - 79.99	C
80.00 - 89.99	B	60.00 - 69.99	D
.	.	≤ 59.99	F

Course Policies:

- **Homework**

- Homework will comprise of reading (and re-reading) course materials and working on your program projects.

- **Lecture vs. Lab and Computers**

- Please don't use computers during lecture, unless directed to do so.
- It is widely understood that learning is best accomplished by combining the processes of listening, looking and writing—each occupies a relatively independent neural network, the whole of which is greater than the sum. Therefore it is recommended that you take notes during lecture.
- It is essential that you have access to a computer with an Integrated Development Environment (IDE) which you have access to outside of class—you will need this to be able to modify code and test your changes in a timely manner. See the Resources

- **Grades**

- Grades in the **C** range represent performance that **meets minimal expectations**; Grades in the **B** range represent performance that is **substantially better** than the expectations; Grades in the **A** range represent work that is **excellent**.

- **Attendance and Absences**

- Attendance is expected and will be noted. If you're not there, you missed it.
- Students are responsible for all missed work, regardless of the reason for absence. It is also the absentee's responsibility to get all missing notes or materials.

Academic Honesty Policy

In addition to skills and knowledge, College of the Desert aims to teach students appropriate ethical and professional standards of conduct. The college catalog specifies that students are expected to "Integrate universally accepted values such as honesty, responsibility, respect, fairness, courage and compassion into judgments and decision-making." and that, "Students are expected to act in an honest and trustworthy manner. Work performed on examinations or other forms of evaluation must represent an individual's own work, knowledge and experience of the subject matter. Students are expected to follow the classroom rules established by each instructor." Any attempt to deceive a faculty member or to help another student to do so will be considered a violation of this standard.

Tentative Course Outline:

Week	Content
Week 1	<ul style="list-style-type: none">• Gaddis Chapter 10: <i>Pointers</i>• Stroustrup (PPP) Chapter 17: <i>Vectors and Free Store</i>
Week 2	<ul style="list-style-type: none">• Gaddis Chapter 11: <i>More Classes and OOP</i>• PPP Chapter 9: <i>Technicalities; Classes, etc.</i>
Week 3	<ul style="list-style-type: none">• Gaddis Chapter 12: <i>More on C-Strings and the String Class</i>• PPP Chapter 23 <i>Text Manipulation</i>
Week 4	<ul style="list-style-type: none">• Gaddis Chapter 13: <i>Advanced File and I/O Operations</i>• PPP Chapter 10: <i>Input and Output Streams</i>
Week 5	<ul style="list-style-type: none">• Gaddis Chapter 14: <i>Recursion</i>
Week 6	<ul style="list-style-type: none">• Chapter 15: <i>Polymorphism and Virtual Functions</i>•