

Syllabus  
COSC 052 - Summer 2012 - Computer Science II  
3:15 pm - 5:15 pm Monday, Tuesday, Wednesday, Thursday  
ICC 205B

**Instructor:** Willis Addison Woods  
Adjunct Assistant Professor  
Department of Computer Science  
**Office:** Room 342C, St. Mary's Hall  
**Phone:** 703-608-3056  
**Email:** [addison@cs.georgetown.edu](mailto:addison@cs.georgetown.edu)  
**Web:** <http://cs.georgetown.edu/~addison/>

**Office Hours:** Monday, Tuesday, Wednesday 5:30 pm – 6:45 pm  
(or by appointment)

**TAs:** TBD

**Course Description:**

This course surveys advanced topics of C++ programming and introductory data structure concepts. It is intended for computer science majors, minors, and other students with a serious interest in learning C++ programming. The course covers the following topics: program organization, pointers, self-referential classes, dynamic object creation and destruction, linked lists, recursion, inheritance, abstract base classes, virtual functions, polymorphism, template classes, exception handling, C-style arrays, random file access, Big-Oh notation, abstract data types, stacks, queues, dequeues, lists, vectors, sequences, priority queues, binary trees, binary search trees, elementary graphs, searching, and sorting. This course satisfies the college science requirement.

**Prerequisites:**

COSC-051

**Required Text:**

Data Structures and Algorithms, 2<sup>nd</sup> Edition by Michael T. Goodrich,  
Robert Tamassia, and David Mount

**Recommended Text:**

C++ Programming: From Problem Analysis To Program Design, 5<sup>th</sup> Edition  
by D.S. Malik

**Grading:**

This course includes a total of 1000 points worth of tests, projects, participation, and other graded work.

Exams (45%): Three exams, 150 points each.

Programming Projects (40%): Four projects, 100 points each.

Homework/Quizzes/Class participation (15%): Regular quizzes and graded homework assignments throughout the course. There are 150 total points in this category. Point value of individual assignments will be determined when assigned.

**Grading Scale:**

<i>Grade</i>	<i>Range</i>
A	94 and up
A-	90-93
B+	87-89
B	83-86
B-	80-82
C+	77-79
C	74-76
C-	70-73
D	61-69
F	60 and below

**Submitting Assignments:** Assignments will be posted on Blackboard. All electronic submission requirements (source code, reports, conclusions, etc.) must be posted to Blackboard prior to the due date and time. Source code should be text files with the appropriate extension. Other file formats will be specified in the project description if applicable, naming conventions will be specified in the project description. For any other hard copy reports or documents, I prefer pdf files but will accept any Microsoft Office compatible file format.

Additionally, please note:

- All assignments are due before class begins on the due date
- 2.5% penalty per quarter-hour for late submission
- No make-up exams will be provided

**Programming Environment:** This class is about the use of computer programming to solve problems. You will do a lot of C++ programming. There are a wide variety of development environments that you can use to create C++ programs. I will provide links to some of these options separately. Installation and use of any such third party application is optional, is your responsibility, and will not be covered during class. All graded projects and homework assignments must compile on the computer science server specified for this class (cs-class.uis.georgetown.edu). Before submitting any programming assignment, your source code must be copied to the server and compiled using the GNU C++ compiler provided on that server. Again, there are many different development environments and computer tools that you may use to accomplish this. The simplest option is to create your programs directly on the server using a UNIX text editor. This is perfectly acceptable and eliminates the need to transfer files to the server prior to compiling your program.

**Attendance and Expectations:** Attendance is required. Not attending lectures will have an adverse effect on your class participation score. Further, you will be responsible for everything covered in class even if it is not in the textbook. Class meetings will include pop quizzes and if you miss one of those there will be no makeup. If you need to leave the classroom during a lecture feel free to do so as quietly as possible. Please turn off cell phones or set them to vibrate prior to the start of class.

**Academic Honesty:** 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 unless you need to ask **me** to clarify something regarding an exam question. In general exams will be closed book and closed notes unless otherwise specified.

All homework assignments and individual projects must be the result your own effort. You may use outside resources such as research papers and books from the library but any solution techniques taken from outside sources must be properly documented. You are permitted to have general discussions and interactions with other students concerning computing techniques. However, you are not permitted to discuss the specific details of your solution. You are strictly prohibited from viewing or copying someone else's work or source code; or allowing someone else to view or copy your source code. You may not email or otherwise provide to someone else the files associated with your programming project or other homework documents. You may not submit someone else's homework file or files as your own. Regardless of how much assistance you receive you must complete the assignment yourself and physically type your programs, reports, and all other assignment products on your own.

**Weekly Course Schedule:** Provided separately.

### **Schedule of Topics Covered (tentative)**

- inheritance
- composition
- pointers
- classes
- virtual functions, abstract classes
- lists
- overloading
- templates
- exception handling
- recursion
- linked lists
- stacks
- queues
- searching algorithms
- sorting algorithms
- algorithm analysis
- asymptotic notation
- binary trees
- graphs
- standard template library

*Course topics, administrative guidelines, and other specifics discussed in this syllabus are subject to change. Notice of any changes will be provided in class.*