

COURSE SYLLABUS
605.404.71 Object-Oriented Programming with C++
Summer 2007 Johns Hopkins University

Instructor: Mr. John E. Boon, Jr.

NOT AVAILABLE PERIODS SUMMER 2007

Monday	Tuesday	Wednesday	Thursday	Friday
	605.404 - 17:00-20:30		605.404 - 17:00-20:30	11:30-13:30

Mr. John E. Boon, Jr. is the course instructor for these class meetings. He is an Operations Research Analyst for RAND, and is a Lecturer in the Whiting School Engineering and Applied Science Programs for Professionals. Mr. Boon was previously an Associate Professor of Computer Science and Chairperson, Department of Mathematics and Computer Science, Hood College. Mr. Boon also continues to maintain a consulting practice in operations research and computer science.

You are encouraged at any time to phone my cell number. You may fax information to my home office. I encourage you to contact me if you have any questions at all. I may be able to clarify requirements as well as suggest additional resources and strategies for a given problem. Be sure to (1) state your name, (2) state what I may do for you, (3) state your phone number twice, (4) state the hours during which I may return your call if you leave a voice message.

My complete list of contact information follows:

My JHU email address: <mailto:jboonjr@apl.jhu.edu>

WWW Home Pages: <http://www.apl.jhu.edu/Notes/Boon/605404/>

Phone: **(301) 606-4115 (cell)** || (301) 371-5411 (voice and facsimile)

Course Description:

This course provides in-depth coverage of object-oriented programming principles and techniques using C++. Topics include classes, overloading, data abstraction, information hiding, encapsulation, inheritance, polymorphism, file processing, templates, exceptions, container classes, and low-level language features. The course briefly covers the mapping of UML design to C++ implementation and object-oriented considerations for software design and reuse. The course also relates C++ to GUI, databases, and real-time programming. Optional topics include the comparison of C++ with other OOP languages and techniques for interfacing C++ with Java.

Prerequisites:

Knowledge of C.

Computer Requirement:

Students must have an email address and must make that address known to the course instructor as soon as practical after the start of the course. Students may use their own computing resources (hardware/software tools) or may arrange to access the necessary tools through the student services entities of JHU. Class examples will be compiled and processed with GNU port of gcc for 32-bit Windows version (Cygwin). Students with JHU computer accounts may also access the GNU port of gcc on the Solaris system. Students must have access to the Internet, have a web browser

June 7 version

and the Acrobat Reader application.

Instructional Objectives:

The course will limit its implementations to ANSI Standard C++. This means that we will not be investigating the extensions to C++ provided in libraries like the Microsoft Foundation Classes (MFC) and we will not concentrate on Borland or Microsoft environment-specific extensions.

Students completing this course will demonstrate mastery of C++ syntax and semantics.

Students completing this course will demonstrate mastery of fundamental object-oriented programming techniques using C++ (data abstraction, information hiding, encapsulation, inheritance, polymorphism).

Students completing this course will demonstrate mastery of integrating STL components with their own C++ programs.

Required Texts:

Bjarne Stroustrup. *The C++ Programming Language: Special Edition (3rd edition)*. Addison-Wesley, 2000.

Nicolai M. Josuttis, *The C++ Standard Library: A Tutorial and Reference*. Addison-Wesley 1999.

Optional Resource Texts:

I will draw course materials from at least the following additional texts:

- Margaret A. Ellis and Bjarne Stroustrup. *The Annotated C++ Reference Manual*. Addison-Wesley, 1990.
- James O. Coplien, *Advanced C++ Programming Styles and Idioms*. Addison-Wesley, 1992.
- Brian W. Kernighan and Rob Pike, *The Practice of Programming*. Addison-Wesley, 1999.
- Texts by Scott Meyers, Addison-Wesley.

Internet Resources, Library Reserve Texts, Information, Articles, and Videos:

You are encouraged to consult resources in Dr. Dobb's Journal, <http://www.ddj.com> for topical articles on C++/STL.

It is your responsibility to check the class WWW pages no later than Noon each class day for updates and class information (<http://www.apl.jhu.edu/Notes/Boon/605404/>). Please note weekly changes to the class WWW page for posted links and special resources related to the course as it progresses. I tend to post extensive class information on my class pages (important news items about the upcoming class, items important to the upcoming class lecture, notes, links, problems, solutions, projects, programming resources). I promise to have the page updated by Noon the day of class for any information you may need to bring with you to class that day.

June 7 version

You are encouraged to obtain a JCARD for accessing the JHU library. You can learn how to access the JHU libraries from off-campus at <http://www.library.jhu.edu/services/computing/remotearchive.html>.

Reading Assignments:

You will be expected to read assigned material before the class at which it will be discussed. You do not serve yourself by being exposed to these ideas for the first time in lecture. Keep notes in the margins or in a notebook to remind you to ask them during lectures later. You will be expected to search out and read additional material on topics during this course.

Homework/Programming Assignments:

Homework assignments will involve reasoning, programming, and computation. Although the Internet provides more examples and potential homework solutions than you may ever wish to wade through, you are encouraged to think and assimilate, rather than regurgitate. The objective of homework this semester will be to gauge your understanding of C++/STL syntax and semantics. For this you need your mind, not just books and the Internet.

Please note that in all assignments, you must cite all references and sources you use (see links on course WWW page for proper form of references of WWW pages). I will not tolerate the use of quoted or paraphrased information, algorithms, or code segments from published sources or from others without appropriate citation.

You may write programs only in C++. I will use the GNU ports by Cygwin for class examples and during any re-compilation of your work should I need to suggest a change or improvement to your solution. We will not use any Borland, Microsoft, or other vendor-specific extensions during the duration of this class.

I will provide you with a code header that you will complete for each program. You will be required to develop test cases for each program you submit. When you deliver your work, submit:

1. CD formatted for Window/Intel computer OR single .zip file transmitted to jboonjr@apl.jhu.edu containing:
 - a. program source code, including completed code header documentation (http://www.apl.jhu.edu/Notes/Boon/Common/CPP_Links.shtml)
 - b. data file(s) required by your program;
2. printed listing(s) of your program(s);
3. printed test case statements and printed test case results.

Failure to submit all required materials will result in a reduction of grade on the assignment. Homework will be assigned at each class meeting. Homework/programming assignments will be due only one time per week, though, on Tuesday.

Exams:

One exam is planned. The exam is intended to assess your mastery of the key concepts and relationships investigated during the course.

June 7 version

Graded Assignments:

Distribution of weights in grading:

- **Homework/Programs** **(65%)**
- **Exam** **(25%)**
- **Class Participation** **(10%)**

Homework and programming assignments turned after the due date will not be accepted. Up to one homework/programming assignments may be marked as excused for extraordinary situations; no points will be assigned and the assignment will not be used in computing your final grade. Each homework/programming assignment is equally important to your grade (that is, some homework assignments contain more problems than others but all are scored on a percent of possible basis resulting in a % earned out of 100% available score).

Grading Policy:

I will award partial credit for work done even if the result is incorrect, but this implies that you show all your intermediate work and clearly label your answer. I will deduct points for answers that do not make any sense at all; you should always check your work, even work done using the computer. The explanation of your work is as important as the work itself -- do not concentrate on the programs and ignore the importance of clear descriptions of what you did and what it means.

The following grading scale will be used 100-90 = A, 89-80=B, 79-70=C, 69-Below=F. Letter grades approximately imply:

- A - complete understanding of the topic plus additional insight, creative, or other indicators of advancement beyond complete understanding of the topic; no programming errors and program is correct (adheres to the specification), efficient, clear, and simple;
- B - mastery of the basic material; few programming errors and program is correct (adheres to the specification) but not necessarily efficient, clear, and simple;
- C - attempt to complete the basic material but significant gaps in understanding and mastery exist; many programming errors and program is not correct (adheres to the specification), efficient, clear, and simple;
- F - failure to complete basic material successfully.

I do not grade on a curve.

Attendance:

Class attendance is essential. Significant material is covered at each of our class meetings this semester. I may excuse absences if I am notified.

Academic Standards:

Students are reminded of sections in the current catalog: Academic Ethics and Violations of Academic Integrity¹.

All assignments this semester are individual effort assignments. It is a violation of the rules of academic conduct in this class for individuals to collaborate with other individuals, whether or

1 http://catalog.epp.jhu.edu/content.php?catoid=9&navoid=194#acad_ethi

June 7 version

not they are members of this class, on assignments, unless specifically directed that such collaboration is allowed for a specific assignment by the professor. You may ask one another about the objectives of assignments if they are not clear; you are encouraged to ask the instructor for definitive answers to assignment questions. You may tell other students what output was produced from your assignment, unless I otherwise direct you to keep this information private for a specific assignment. You may not ask to see another student's program or assignment solution. You must be careful not to plagiarize other students' work by asking questions of other students and building your solution piecemeal from their answers. If one or more students collaborate on an assignment solution beyond these guidelines, I will award a score of zero to each student's work. You may not reuse any code produced by students in previous semesters or from students in other classes. If you incorporate in any way code from a published source, you must fully cite that source and provide a copy of the source to me with your assignment. I am familiar with most of the magazines, journals, and texts that support this course. If I suspect that you committed plagiarism, I will treat this as a violation of academic conduct until proven otherwise.

During class, you are expected to concentrate on and contribute to class presentations, lecture, and group discussion. Small group discussions that distract from the ability of others in class to adequately concentrate upon class presentations, lecture, and group discussion will not be tolerated. Use of laptop computers during class will be restricted to note taking and computer-based activities as assigned by the instructor. Internet surfing, writing of other papers, or use of programs not specifically related to the lecture will not be tolerated.

June 7 version

TENTATIVE COURSE SCHEDULE
605.404.71 Object-Oriented Programming with C++, Summer 2007, JHU

Class meetings Tuesday/Thursday 6:00pm-8:25pm, A&R 213, Montgomery County Center.

Class	Date	Objectives	Text
1	06/07	Intro to semester; programming style and documentation conventions	S: Chapters 1-3, Sections 24.3.7.1-24.3.7.3 J: Chapters 1-3
2	06/12	I/O using Streams	S: 21 and J: Chapter 13
3	06/14	Types, Declarations, Pointers, Arrays, and Structures	S: Chaps 4-5
4	06/19	Types, Declarations, Pointers, Arrays, and Structures	S: Chaps 4-5
5	06/21	Expressions and Statements	S: Chaps 6
6	06/26	Functions	S: Chaps 7
7	06/28	Namespaces and Exceptions; Source files and Programs	S: Chaps 8-9
8	07/03	Classes and Operator Overloading	S: Chaps 10-11
9	07/05	Derived Classes and Templates	S: Chaps 12-13
10	07/10	Exception Handling	S: Chap 14
11	07/12	Class Hierarchies	S: Chap 15
12	07/17	STL General Concepts and Utilities and Containers	J: Chaps 4-6; S: Chaps 16-17
13	07/19	Iterators, Allocators, and Functions	J: Chaps 7-8; S: Chaps 18-19
14	07/24	Algorithms	J: Chaps 9-10; S: Chap 18
15	07/26	Strings and Numerics	J: Chaps 11-12; S: Chaps 20, 22

S – Stroustrup text

J – Josuttis text