CS 525: Object Oriented Design and Programming
Fall 2011 Course Syllabus

Contact Information

Instructor:	Dmitri Volper
Office Phone Number:	895-4452
Email address:	dvolper@digipen.edu
Home page (DigiPen):	`http://faculty.digipen.edu/~dvolper/2011/fall.cs525/index.html`
Office hours:	see my homepage `http://faculty.digipen.edu/~dvolper`

Course Description Course Description This course is a course in object-oriented design and programming using the C++ programming language. It is targeted at the graduate student that is already fluent in one or more programming languages. Among the language-specific topics included are pointers, pointer arithmetic, dynamic memory management, namespaces, scope, operator overloading, generic programming (templates), the Standard Template Library, and standards compliance. Object-oriented topics will cover analysis and design using OO paradigms, advanced meta-programming techniques, use of design patterns, RAII, SFINAE, and efficiency considerations. Prerequisites are programming fluency in another imperative language, preferably with some basic knowledge of C++. After successfully completing this course, students should have a much deeper understanding of the subtleties and complexities of using the object-oriented facilities of the C++ programming language, which is the standard programming language used in the game industry today.

Goals and Objectives Upon successful completing this course, students should

- have a deep understanding of C++ programming language design
- be able to create and compile a project from the command line using GNU, Borland, or Microsoft compiler
- be able to use various GNU tools to speed up debugging and error analysis
- know C/C++ bit manipulation techniques and be able to apply then to minimize memory usage and/or program runtime
- know C/C++ expression evaluation algorithm and be able to write his/her own complex declarations
- understand C/C++ conversion rules
- be able to use aggregation and inheritance to create new classes
- have a good understanding of function and class templates
- be able to use STL containers and generic algorithm to solve complex problems

Course	Day and Time	Room
CS525	T 4:30pm-5:50pm, Th 4:30pm-5:50pm	CURIE

Important days

Textbooks and References
Required

- none

Recommended

- *C++ Primer*, Fourth Edition, by Lippman, Lajoie, and Moo. Published by Addison-Wesley Publishing Company. Copyright ©2005 by Objectwrite Inc. (ISBN: 0-201-72148-1).

Additional references (Optional)

- *Pointers on C*, by Kenneth A. Reek. Copyright ©1998 by Addison Wesley Longman, Inc. (ISBN: 0-673-99986-6).
- *The C Programming Language*, Second Edition, by Kernighan and Ritchie. Published by Prentice Hall.Copyright ©1989 by Bell Telephone Laboratories, Incorporated. (ISBN: 0-13-110362-8).
- *The C++ Programming Language*, Third Edition, by Bjarne Stroustrup. Addison-Wesley . Copyright ©1997 by AT&T. (ISBN: 0-201-88954-4).

- *The C++ Standard Library: A Tutorial and Reference*, by Nicolai M. Josuttis. Addison-Wesley, Copyright ©1999. (ISBN: 0201379260). `http://www.josuttis.com/libbook/`
- *C++ Templates - The Complete Guide*, by David Vandevoorde and Nicolai M. Josuttis, Addison-Wesley, 2002. (ISBN 0-201-73484-2) `http://www.josuttis.com/tmplbook/`
- *The World Wide Web*. Quite possibly the greatest asset to learning since the teacher and the textbook.

Grading
Grades will be derived from homework assignments and exams. The detailed weightings and letter grades are as such:

Homework	35%
Midterm exam	25%
Quizzes	10%
Final exam	30%

You must receive an average score of 60% on both exams combined to pass this course, regardless of your homework scores. Homework scores will only be counted if the above condition is met.

x%	Grade
$x \geq 93$	A
$90 \leq x < 93$	A-
$87 \leq x < 90$	B+
$83 \leq x < 87$	B
$80 \leq x < 83$	B-
$77 \leq x < 80$	C+
$73 \leq x < 77$	C
$70 \leq x < 73$	C-
$60 \leq x < 70$	D
$x < 60$	F

Attendance is mandatory. There are no makeup exams or quizzes. Also, for every lecture that is missed, you will lose one point from your final grade (e.g. a 90 becomes an 89). The only exceptions are if you notify me prior to your absence with a valid reason. (Sleeping, studying for another class, working on your game, etc., are not valid reasons for an absence.) Class participation will boost your grade if you are on the border. (e. g. It is possible to get an A- with an overall average of 88.5%)

Classroom policies.

- No food.
- Drinks are allowed, unless prohibited by School policies.
- No loud noises.
- Laptops are allowed if used to display lecture material.
- No strong smells.

Tentative Schedule

Week	Topic	Reference Material
1	Intro to CS525, compilers, tools, Doxygen, File I/O, Text files vs. binary files assignment 1 due 3th week	Notes
2	Command line args, bit manipulation, mixing C and C++ code, calling conventions	Notes
3	Linked lists, Complex declarations, function pointers assignment 2 due 5th week	Notes
4	Exceptions	Textbook Ch. 17
5	Inheritance, interfaces, and polymorphism assignment 3 due 7th week	Textbook Ch. 15
6	Class templates, Function templates Midterm	Textbook Ch. 16
7	Using STL containers, iterators assignment 4 due 9th week	Textbook Ch. 9, 10
8	Generic algorithms, function objects	Textbook Ch. 11, Ch. 14.8
9	Generic algorithms, function objects	Textbook Ch. 11, Ch. 14.8
10	Implementing STL containers assignment 5 due 11th week	Notes
11	Overloaded operators, conversions, and efficiency	Textbook Ch. 14.1 – 14.7
12	new and delete operators, operator new, operator delete assignment 6 due 13th week	Textbook Ch. 18.1.3 – 18.1.6
13	Fall break	
14	Advanced templates	
15	Final Exam	

Submitting Homework Programming assignments will (obviously) use the C and C++ language. More specifically, all

programs must adhere to Standard C and C++, which is what this course is about. Assignments will be graded using Microsoft's C/C++ compiler version 9.0 (Visual Studio 2008), GNU's gcc/g++ compilers (version 4.3.2), and Borland's (version 5.82) compiler. All of these compilers are installed on DigiPen computers and available for free to all DigiPen students (see class website for instruction on downloading Cygwin (GCC) and Borland. MS VS2008 is available using your MSDN account (email me if you do not have one). Note that DigiPen computers have multiple versions of GCC and MS compilers, to check which one is the default type:
type `gcc --version` to check GCC version
you should see `gcc-4 (GCC) 4.3.2 .....`
type `cl` to check MS compiler version
VS2008 should contain `15.00.XXXXXX`

If versions are not as above, you should

- in Cygwin terminal (this is a permanent change)
  `ln -s /usr/bin/gcc-4 /usr/bin/gcc`
  `ln -s /usr/bin/g++-4 /usr/bin/g++`
- on DigiPen computers run `setvs9.bat` from `utils` folder.

Note that by default GCC is only available in Cygwin terminal and MS only in MS terminal. To make all 3 compilers available in the same terminal use instructions on the class website.

The source files must be submitted electronically through the course submission page - use your digipen login and student number to login.
`http://faculty.digipen.edu/~dvolper/cgi-bin/login.cgi`

Your source code should be archived in zip format. It is imperative that you begin thinking about your professional career as a game developer as soon as you learn to program. To motivate you towards this goal, at least 15% of the grade on a homework assignment is based on programming quality, clarity, and documentation. This means that even if you turn in a program that runs perfectly, you can expect a grade no higher than a C if you fail to adhere to good programming standards. (Documentation samples are posted on the course web page.) Partial credit will be awarded for incomplete assignments. The code documentation requirements will be discussed in class.

There will be 6 programming assignments during the semester, with the first one being assigned during the second week. You will be given between 7 and 14 days to complete each assignment. This gives you adequate time to manage your workload. The amount of time actually required to complete an assignment is much less than the time allotted and is generally between 6 and 8 hours. Depending on your grasp of the subject matter during the lectures, some of you will require more or less time to complete the assignments. In any event, you should plan on devoting 5 hours per week to this course (outside of the lectures).

Academic integrity Academic dishonesty, or cheating, occurs when a student represents someone else's work as their own, or assists another student in doing so. This can happen on exams, quizzes, homework, or projects. Academic dishonesty also may occur when a student uses any prohibited reference or equipment in the completion of a task. For example, the use of a calculator, notes, books or the internet when it is prohibited. Plagiarism is a common form of academic dishonesty. This can take the form of copying and pasting exerpts from the web, and representing them as original work. The type and severity of any occurence, as well as the legitimacy of any claim of academic dishonesty, will be judged by the instructor and the disciplinary committee. All students are asked to help in promoting a culture of academic integrity by discouraging cheating in all forms.

Submissions The following must be submitted electronically:

- source code for programming assignments
- multiple-choice answers to quizzes, midterm(s), and final exam. Note that hard-copy of quizzes, midterm(s), and final exam should be returned in class (use a separate sheet of paper to copy multiple-choice answers and test ID).

Programming Assignment Late Submissions For programming assignments late submission policy is 1 point every 2 hours, for a period of at least 2 days. After that no submission is accepted.

Multiple-choice Answers Late Submissions You'll be given 3 hours to submit your answers online, after that submission will be closed and I will submit your answers using the hardcopy (which will cost you 20% of the multiple-choice part of the grade).

Accommodations Students with physical, psychological or learning disabilities that affect their ablities to perform major life activities associated with this class may be eligible for reasonable accommodations under the Americans with Disabilities Act. If you have a documented disability please contact the Disabilities Support Servicies office to arrange for accomodations for this class.

Grades You can see all your current grades through my submission page. Login and choose "Standing" (button at the bottom), you'll see something like

```
test assign1 (weight 10): a1 out of 100
test assign2 (weight 10): a2 out of 100
test quiz1 (weight 10): q1 out of 72
test assign3 (weight 10): a3 out of 100
test quiz2 (weight 15): q2 out of 35
test quiz3 (weight 15): q3 out of 20
Total T out of 70 (P %)
```

Each assignment/exam/quiz has a weight which determines it's effect on your total/final grade, which is used as follows:

```
T = a1/100 * 10 + a2/100 * 10 + ... + q2/35 * 15 + ...
W = sum of all weights (70 in this example)
and
P = T / W * 100
```

Note that the total number of points (70 in the above example) doesn't have to be 100 in the end of a semester, but will be close.

P value is the easiest to use – just consult the letter-grade conversion table above.

Questions like "can I still pass this class?"
Answer: let's say you currently have `"T out of W=60 points"` and `P = T / W * 100 < 70%` , that is failing the course. Assuming the remaining number of points in the course is 35, then IF you get X out of them, then your final grade will be `"T+X out of W=60+35 points"`, which is `P = (T+X) / (60+35) * 100 %` . Now solve inequality `P = (T+X) / (60+35) * 100 >= 70` for X.