MONASH University

# FIT2049
# Games programming using C++

## Unit guide

## Semester 2, 2008

*Last updated : 30 Jun 2008*

# Table of Contents

# FIT2049 Games programming using C++ - Semester 2 , 2008

## Unit leader :

Lindsay Smith

## Lecturer(s) :

### Caulfield

- Kieran Love

## Tutors(s) :

### Caulfield

- Kieran Love

# Introduction

Welcome to FIT2049 Games programming using C++ for semester 2, 2008. This 6 point unit is core in the Games Development major of the BITS degree. The unit has been designed to introduce students to the key programming language used in games development, C++. The unit will transition students programming skills from Java (where you have previously completed two units) into C++ and also introduce the basic building blocks of game programming in Microsoft Windows DirectX.

# Unit synopsis

ASCED Discipline Group classification: 020103 Programming.

This unit will further develop object-oriented programming skills, and introduce the C++ language to students. Streams, pointers & arrays, classes, inheritance & polymorphism, templates and the STL, along with the I/O class hierarchy will be discussed at length. Interactive programming techniques will be used to solve various programming exercises. This unit will build upon previous programming skills, and provide a strong grounding for further study in this area, especially related to games engine development. The unit will examine game creation using C++ and Microsoft Windows DirectX.

# Learning outcomes

Knowledge and Understanding

- An understanding of the history and concepts of the C++ language.
- An understanding of how C++ relates to other commercial languages, especially Java.
- An understanding of the features and capabilities of C++, comprising:
- Streams
- Pointers and arrays
- Classes, inheritance and polymorphism
- Templates and the STL
- The I/O class hierarchy
- An understanding of the possible solutions/approaches when using C++ for interactive programming.
- An understanding of Microsoft DirectX (2D and 3D) and the role of the Windows API in game development

Attitudes, Values and Beliefs

- Enthusiasm for interactive programming
- Motivation to develop further programming skills
- The confidence to understand and explain existing C++ code

Practical Skills

- Skills in adapting Java code to C++
- The ability to design, develop and debug software applications written in C++, with a focus on interaction
- Skills in manipulating music and sound effects within a game via C++
- Create a 3d interactive environment, using C++, that displays the techniques learnt during the unit

This is the first major offering of this unit for the Games Development major - we will be seeking your help and feedback as we proceed through the semester to ensure the unit is structured so that you gain the most value from your study. As a result there may be some variation in the scheduled weekly topics.

# Workload

For on campus students, the **weekly** workload commitments are:

- four hours of lectures / laboratory (requiring advance preparation), and
- eight hours of self directed study - this will include reading and computer based activities.

# Unit relationships

## Prerequisites

Before attempting this unit you must have satisfactorily completed:

FIT1007 and FIT2048, or equivalent.

## Relationships

FIT2049 is a core unit in the Games Development major of the BITS degree.

You may not study this unit and MMS2804 in your degree.

# Continuous improvement

Monash is committed to ‘Excellence in education' and strives for the highest possible quality in teaching and learning. To monitor how successful we are in providing quality teaching and learning Monash regularly seeks feedback from students, employers and staff. Two of the formal ways that you are invited to provide feedback are through Unit Evaluations and through Monquest Teaching Evaluations.

One of the key formal ways students have to provide feedback is through Unit Evaluation Surveys. It is Monash policy for every unit offered to be evaluated each year. Students are strongly encouraged to complete the surveys as they are an important avenue for students to "have their say". The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

# Student Evaluations

The Faculty of IT administers the Unit Evaluation surveys online through the my.monash portal, although for some smaller classes there may be alternative evaluations conducted in class.

If you wish to view how previous students rated this unit, please go to
http://www.monash.edu.au/unit-evaluation-reports/

Over the past few years the Faculty of Information Technology has made a number of improvements to its courses as a result of unit evaluation feedback. Some of these include systematic analysis and planning of unit improvements, and consistent assignment return guidelines.

Monquest Teaching Evaluation surveys may be used by some of your academic staff this semester. They are administered by the Centre for Higher Education Quality (CHEQ) and may be completed in class with a facilitator or on-line through the my.monash portal. The data provided to lecturers is completely anonymous.
Monquest surveys provide academic staff with evidence of the effectiveness of their teaching and identify areas for improvement. Individual Monquest reports are confidential, however, you can see the summary results of Monquest evaluations for 2006 at http://www.adm.monash.edu.au/cheq/evaluations/monquest/profiles/index.html

# Unit staff - contact details

## Unit leader

**Mr Lindsay Smith**
Deputy Head of School
Phone +61 3 990 47201
Fax +61 3 990 47089

## Lecturer(s) :

**Kieran Love**

## Tutor(s) :

**Kieran Love**

## Teaching and learning method

The unit will be delivered via lectures and laboratories.

Lecture: During the lecture, your lecturer will introduce key theoretical concepts and demonstrate various approaches to games programming tasks.

Laboratory: The labs consist of a set of graded exercises which allow you to put the theory presented in the lecture to work in implementing a programming solution to a practical problem.

Before the lab you should carefully read through the lab activities. The teaching staff will presume that you have completed all the posted lab tasks each week and build subsequent activities on this assumption. For this reason it is very important that you complete all the posted tasks (please note you will not be able to complete them in the allocated time, these will be completed in your self study 8 hours). Given the cumulative nature of the learning, it is easy to fall behind if either you do not complete the required work or fail to understand key tasks/concepts. If you are having problems with lab exercises, please ensure you speak to your tutor and gain some assistance.

## Tutorial allocation

On-campus students should register for laboratories using Allocate+.

## Communication, participation and feedback

Monash aims to provide a learning environment in which students receive a range of ongoing feedback throughout their studies. You will receive feedback on your work and progress in this unit. This may take the form of group feedback, individual feedback, peer feedback, self-comparison, verbal and written feedback, discussions (on line and in class) as well as more formal feedback related to assignment marks and grades. You are encouraged to draw on a variety of feedback to enhance your learning.

It is essential that you take action immediately if you realise that you have a problem that is affecting your study. Semesters are short, so we can help you best if you let us know as soon as problems arise. Regardless of whether the problem is related directly to your progress in the unit, if it is likely to interfere with your progress you should discuss it with your lecturer or a Community Service counsellor as soon as possible.

## Unit Schedule

| Week | Topic | Key dates |
|---|---|---|
| 1 | Unit Introduction, Basic Syntax of C++ & Programming Constructs | |
| 2 | References, Pointers, Arrays and Functions | |
| 3 | Inheritance and Polymorphism | |
| 4 | Template classes and functions - Standard Template Library | |
| 5 | File Input and Output | |
| 6 | Windows Programming - Initialization and Main Loop for Games, Design Patterns | |
| 7 | Introduction to DirectX | Assignment 1 Due |
| 8 | Game Mathematics | |
| 9 | DirectX Graphics - lighting and objects, textures | |
| 10 | Game Events, Collisions and Physics | |
| 11 | Input Detection - DirectInput | |

| | | |
|---|---|---|
| Mid semester break | | |
| 12 | DirectSound and DirectMusic | Assignment 2 Due.<br>Lab Test |
| 13 | Revision | |

# Unit Resources

## Prescribed text(s) and readings

No text is prescribed for this unit, you will make use of a variety of resources as specified in the lecture notes.

## Recommended text(s) and readings

Grandon G.T., 'Introduction to Programming Using Visual C++ .Net', Wiley & Sons, 2005, ISBN: 0471-48724-4

Deitel H.M., Dietel P.J., 'Visual C++ .Net - How to Program', Pearson Education, 2004, ISBN: 0-13-437377-4

Thorn A., 'DirectX 9 Graphics – The definitive guide to Direct3D', Wordware, 2005, ISBN: 1-55622-229-7

Sherrod A., 'Ultimate Game Programming With DirectX', Charles River Media, 2006, ISBN: 1584504587

Internet based resources:

- http://directxworld.altervista.org/index.php?link=graphic_lessons
- http://www.cplusplus.com/doc/tutorial/

## Required software and/or hardware

You will need access to Visual Studio - if you wish to use this software at home the CDs can be borrowed from the technical services desk.

On-campus students may use this software which is installed in the computing labs. Information about computer use for students is available from the ITS Student Resource Guide in the Monash University Handbook.

## Equipment and consumables required or provided

On-campus students may use the facilities available in the computing labs.Information about computer use for students is available from the ITS Student Resource Guide in the Monash University Handbook.You will need to allocate up to **8** hours per week for use of a computer, including time for newsgroups/discussion groups.

## Study resources

Study resources we will provide for your study are:

- Weekly detailed lecture notes outlining the learning objectives, discussion of the content, required readings and exercises;
- Weekly tutorial or laboratory tasks and exercises with sample solutions provided one to two weeks later;
- Assignment specifications and sample solutions;
- A sample examination and suggested solution
- Discussion groups;
- This Unit Guide outlining the administrative information for the unit;
- The unit web site on MUSO, where resources outlined above will be made available.

## Library access

The Monash University Library site contains details about borrowing rights and catalogue searching. To learn more about the library and the various resources available, please go to http://www.lib.monash.edu.au. Be sure to obtain a copy of the Library Guide, and if necessary, the instructions for remote access from the library website.

## Monash University Studies Online (MUSO)

All unit and lecture materials are available through MUSO (Monash University Studies Online). Blackboard is the primary application used to deliver your unit resources. Some units will be piloted in Moodle. If your unit is piloted in Moodle, you will see a link from your Blackboard unit to Moodle (http://moodle.monash.edu.au) and can bookmark this link to access directly. In Moodle, from the Faculty of Information Technology category, click on the link for your unit.

You can access MUSO and Blackboard via the portal: http://my.monash.edu.au

Click on the Study and enrolment tab, then Blackboard under the MUSO learning systems.

In order for your Blackboard unit(s) to function correctly, your computer needs to be correctly configured.

For example:

- Blackboard supported browser
- Supported Java runtime environment

For more information, please visit: http://www.monash.edu.au/muso/support/students/downloadables-student.html

**You can contact the MUSO Support by: Phone: (+61 3) 9903 1268**

For further contact information including operational hours, please visit:
http://www.monash.edu.au/muso/support/students/contact.html

Further information can be obtained from the MUSO support site:
http://www.monash.edu.au/muso/support/index.html

# Assessment

## Unit assessment policy

To pass this unit, a student must obtain :

- 40% or more in the unit's examination (lab test and written paper) and
- 40% or more in the unit's non-examination assessment (two assignments) and
- an overall unit mark of 50% or more

If a student does not achieve 40% or more in the unit examination or the unit non-examination assessment then a mark of no greater than 44-N will be recorded for the unit.

## Assignment tasks

- ### Assignment Task

  **Title :** Text Based Game

  **Description :**

  Using C++ students will develop a text based game

  **Weighting :** 20%

  **Criteria for assessment :**

  These will be supplied as part of the assignment task.

  **Due date :** 29th August 2008
- ### Assignment Task

  **Title :** DirectX Game

  **Description :**

  Using C++ and DirectX students will develop a graphical game

  **Weighting :** 20%

  **Criteria for assessment :**

  These will be supplied as part of the assignment task.

  **Due date :** 10th October 2008

## Examinations

- **Examination**

  **Weighting :** 30%

  **Length :** 2 hours

  **Type ( open/closed book ) :** Open book

  **Remarks ( optional - leave blank for none ) :**

  Unit Lab Test run during week 12
- **Examination**

  **Weighting :** 30%

  **Length :** 2 hours

  **Type ( open/closed book ) :** Closed book

  **Remarks ( optional - leave blank for none ) :**

  Written Theory Examination

## Assignment submission

Assignments will be submitted by **electronic** submission to MUSO/Moodle.

Do not email submissions.

## Assignment coversheets

Coversheets will be made available electronically via the MUSO/Moodle assignment submission process.

# University and Faculty policy on assessment

## Due dates and extensions

The due dates for the submission of assignments are given in the previous section. Please make every effort to submit work by the due dates. It is your responsibility to structure your study program around assignment deadlines, family, work and other commitments. Factors such as normal work pressures, vacations, etc. are seldom regarded as appropriate reasons for granting extensions. Students are advised to NOT assume that granting of an extension is a matter of course.

Requests for extensions must be made to the unit lecturer at your campus at least two days before the due date. You will be asked to forward original medical certificates in cases of illness, and may be asked to provide other forms of documentation where necessary. A copy of the email or other written communication of an extension must be attached to the assignment submission.

## Late assignment

Assignments received after the due date will be subject to a penalty of 5% per day, including weekends. Assignments received later than one week (seven days) after the due date will not normally be accepted. In some cases, this period may be shorter if there is a need to release sample solutions.

This policy is strict because comments or guidance will be given on assignments as they are returned, and sample solutions may also be published and distributed, after assignment marking or with the returned assignment.

## Return dates

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Assessment for the unit as a whole is in accordance with the provisions of the Monash University Education Policy at http://www.policy.monash.edu/policy-bank/academic/education/assessment/

We will aim to have assignment results made available to you within two weeks after assignment receipt.

## Plagiarism, cheating and collusion

Plagiarism and cheating are regarded as very serious offences. In cases where cheating  has been confirmed, students have been severely penalised, from losing all marks for an assignment, to facing disciplinary action at the Faculty level. While we would wish that all our students adhere to sound ethical conduct and honesty, I will ask you to acquaint yourself with Student Rights and Responsibilities (http://www.infotech.monash.edu.au/about/committees-groups/facboard/policies/studrights.html) and the Faculty regulations that apply to students detected cheating as these will be applied in all detected cases.

In this University, cheating means seeking to obtain an unfair advantage in any examination or any other written or practical work to be submitted or completed by a student for assessment. It includes the use, or attempted use, of any means to gain an unfair advantage for any assessable work in the unit, where the means is contrary to the instructions for such work.

When you submit an individual assessment item, such as a program, a report, an essay, assignment or other piece of work, under your name you are understood to be stating that this is your own work. If a submission is identical with, or similar to, someone else's work, an assumption of cheating may arise. If you are planning on working with another student, it is acceptable to undertake research together, and discuss problems, but it is not acceptable to jointly develop or share solutions unless this is specified by your lecturer.

Intentionally providing students with your solutions to assignments is classified as "assisting to cheat" and students who do this may be subject to disciplinary action. You should take reasonable care that your solution is not accidentally or deliberately obtained by other students. For example, do not leave copies of your work in progress on the hard drives of shared computers, and do not show your work to other students. If you believe this may have happened, please be sure to contact your lecturer as soon as possible.

Cheating also includes taking into an examination any material contrary to the regulations, including any bilingual dictionary, whether or not with the intention of using it to obtain an advantage.

Plagiarism involves the false representation of another person's ideas, or findings, as your own by either copying material or paraphrasing without citing sources. It is both professional and ethical to reference clearly the ideas and information that you have used from another writer. If the source is not identified, then you have plagiarised work of the other author. Plagiarism is a form of dishonesty that is insulting to the reader and grossly unfair to your student colleagues.

## Register of counselling about plagiarism

The university requires faculties to keep a simple and confidential register to record counselling to students about plagiarism (e.g. warnings). The register is accessible to Associate Deans Teaching (or nominees) and, where requested, students concerned have access to their own details in the register. The register is to serve as a record of counselling about the nature of plagiarism, not as a record of allegations; and no provision of appeals in relation to the register is necessary or applicable.

## Non-discriminatory language

The Faculty of Information Technology is committed to the use of non-discriminatory language in all forms of communication. Discriminatory language is that which refers in abusive terms to gender, race, age, sexual orientation, citizenship or nationality, ethnic or language background, physical or mental ability, or political or religious views, or which stereotypes groups in an adverse manner. This is not meant to preclude or inhibit legitimate academic debate on any issue; however, the language used in such debate should be non-discriminatory and sensitive to these matters. It is important to avoid the use of discriminatory language in your communications and written work. The most common form of discriminatory language in academic work tends to be in the area of gender inclusiveness. You are, therefore, requested to check for this and to ensure your work and communications are non-discriminatory in all respects.

## Students with disabilities

Students with disabilities that may disadvantage them in assessment should seek advice from one of the following before completing assessment tasks and examinations:

- Faculty of Information Technology Student Service staff, and / or
- your Unit Coordinator, or
- Disabilities Liaison Unit

## Deferred assessment and special consideration

Deferred assessment (not to be confused with an extension for submission of an assignment) may be granted in cases of extenuating personal circumstances such as serious personal illness or bereavement. Information and forms for Special Consideration and deferred assessment applications are available at http://www.monash.edu.au/exams/special-consideration.html. Contact the Faculty's Student Services staff at your campus for further information and advice.