Faculty of Information Technology
School of Information Management and Systems
Semester 2, 2005
UNIT OUTLINE

Unit: IMS1906, Business Software Fundamentals


Unit webpage: To access unit webpage, select: http://www.sims.monash.edu.au/subjects/ims1906/index.html

Staff:

<table>
<thead>
<tr>
<th>Lecturer</th>
<th>Angela Carbone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Room</td>
<td>S4.02 Level 4, Building S, - Caulfield Campus</td>
</tr>
<tr>
<td>Phone</td>
<td>9903 1911(o)    0407 886 791(m)</td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:Angela.carbone@infotech.monash.edu.au">Angela.carbone@infotech.monash.edu.au</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tutor</th>
<th>Shiu Ming Tsang</th>
</tr>
</thead>
<tbody>
<tr>
<td>Room</td>
<td>T1.03 Level 1, Building T, - Caulfield Campus</td>
</tr>
<tr>
<td>Phone</td>
<td>9903 2631</td>
</tr>
<tr>
<td>Email</td>
<td>TBA</td>
</tr>
</tbody>
</table>

Contacting staff: Outside the scheduled class contact hours, you can contact teaching staff by email, phone, during their consultation hours (available on unit webpage or at SIMS Frontdesk) or by making an appointment.

If you need a staff member urgently and are unable to contact them, please contact:
SIMS Frontdesk, Level 7 – Building S, Ph: 9903 2208
Aim: The unit aims to provide a general coverage of:

- Object-oriented and structured concepts and design techniques using a "hands-on" step-by-step programming approach utilising a commercially relevant programming language.
- Strategies for meeting user requirements and for designing solutions to programming problems will be presented.
- The fundamental programming concepts of the memory model, objects and classes, data types, declarations, expressions and statements, control structures, modules, parameters, files, input and output, and array data structures will be applied within the context of objects, attributes, event-handling and message-passing in a visual interactive development environment.
- Documentation, testing and debugging techniques will be applied throughout the subject. Compilers and interpreters will be discussed within context.

The emphasis throughout the unit will be on giving a broad overview of programming concepts, yet there will be a strong emphasis on practical demonstration of the concepts being studied, and extensive use will be made of practical exercises to highlight aspects of theory.

Objectives: At the completion of this unit the students will:

have knowledge and an understanding of:

- the nature of computer programs and programming languages
- the basic principles and processes of object oriented software design
- the basic principles and processes of structured software design
- the elements contributing to well-designed screens and user interfaces
- how to produce simple programs in a commercially relevant contemporary development environment

have the skills to:

- design object oriented solutions to simple programming problems
- design structured methods
- design and develop correct, well structured and well documented simple computer programs that solve users' needs
- apply sound testing and debugging techniques
- use appropriate techniques to communicate program data between modules
- use appropriate techniques to pass messages and information between objects
- perform event-handling in a graphical user interface
- manipulate data in one dimensional arrays
- display skills and creativity in problem-solving and algorithmic design;
- develop skills to produce simple well-designed screens and user interfaces;
- produce simple programs in a commercially used contemporary development environment.

have developed attitudes which enable them to:

- appreciate the role and importance of design in the development of computer programs;
- appreciate the qualities of a good program.

Prerequisite knowledge:

Nil
Texts and software:

**Recommended texts:**


**Other references:**

**Software:**

All necessary software used is provided in the tutorials.

**Computing and laboratory requirements:**

- One computer equipped laboratory per 20 students, for 2 hours per week for 12 weeks

**Study materials:**

We provide:

- list of required individual and group assessable deliverables
- Downloadable copies of weekly exercises supporting seminar topics and project objectives
- Downloadable copies of weekly seminar slides from web.
Unit structure and organisation:
Unit structure by lecture

<table>
<thead>
<tr>
<th>Week</th>
<th>Week beginning</th>
<th>Topics</th>
<th>Tutes/Labs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18 Jul</td>
<td>Introduction IMS1906 &amp; Program Design</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>25 Jul</td>
<td>Objects and Classes Introduction to VB.NET/IDE</td>
<td>Introduction to program design</td>
</tr>
<tr>
<td>3</td>
<td>1 Aug</td>
<td>Application Development</td>
<td>Designing a copyright screen</td>
</tr>
<tr>
<td>4</td>
<td>8 Aug</td>
<td>Variables and Data Types Input/Output</td>
<td>Creating a simple calculator</td>
</tr>
<tr>
<td>5</td>
<td>15 Aug</td>
<td>The Selection Structure I and II</td>
<td>Using Variable and Data types</td>
</tr>
<tr>
<td>6</td>
<td>22 Aug</td>
<td>Testing and Debugging + TEST 1</td>
<td>A custom made quote</td>
</tr>
<tr>
<td>7</td>
<td>29 Aug</td>
<td>Repetition Structure I and II</td>
<td>Calculating income tax</td>
</tr>
<tr>
<td>8</td>
<td>5 Sep</td>
<td>Functions and Subroutines</td>
<td>Applying Repetition Structures</td>
</tr>
<tr>
<td>9</td>
<td>12 Sep</td>
<td>File I/O + List Boxes</td>
<td>Writing Functions and Subroutines</td>
</tr>
<tr>
<td>10</td>
<td>19 Sep</td>
<td>Arrays I and Arrays II</td>
<td>File I/O and Debugging</td>
</tr>
<tr>
<td></td>
<td>26-30 Sep</td>
<td>Mid semester break (non teaching period)</td>
<td></td>
</tr>
</tbody>
</table>

NB. This information is subject to change

Workload:
This is a six point unit which, according to University guidelines, requires you to spend 12 hours per week (a total of at least 156 hours per semester).
The anticipated workload is:
- 2 hours per week lecture
- 2 hour per week tutorials
- 7 hours per week preparation and assignment
- 2 hours per week reading

Assessment:
The assignments (40% weighting) and a (two) hour examination plus 2 (45 minutes unit tests (60% weighting) will be used to assess whether you have achieved the objectives of this subject.
Two assignments, total assessment value 40%
Assignment 1, Value 10%, Due Tutorial Week 5
Assignment 2, Value 15%, Due Tutorial Week 10
Assignment 3, Value 15%, Due Tutorial Week 13

Two unit tests, total assessment value 20%
  Test 1, Value 10%, Lecture Week 6, (testing material weeks 1-5)
  Test 2, value 10%, Lecture Week 11, (testing material weeks 1-10)

A closed book examination, 2 hours, assessment value 40% will be used to test your ability to meet the unit objectives

**Note** Formal supervised assessment
  (e.g. The formal supervised assessment for this unit will be an exam scheduled in the formal examination period following the last week of semester. You are required to be available for the exam and any necessary supplementary assessment procedures until the end of the assessment period. Alternative times for exams will not be approved without a medical certificate for a significant illness, or equivalent evidence.)

**Note:**
- Assignments in this unit are no less important than those of other units. Your inability to manage your time or computing resources will not be accepted as a valid excuse. (Several assignments falling due at the same time is often unavoidable.)
- Backup copies are required to be made of all assignments and retained for 12 months, in case of loss.
- Hardware failures are not normally recognised as a valid reason for obtaining an extension or handing in a late assignment.
Assessment Notes

1 Acknowledgment of sources

Each time you complete any assessment, please refer to and make yourself familiar with the most current information regarding acknowledgement of sources, plagiarism and academic conduct contained in the SIMS Policy website.

http://www.sims.monash.edu.au/policies

2. Assignments

2.1 Standards for presentation

All printed assignment work must be word processed and meet the standards set out in the assignment. Refer also to the School of Information Management and Systems guidelines for writing assignments for additional information on presentation standards:


2.2 All assignments must include an appropriate signed SIMS assignment cover page. See the SIMS web site for downloadable (PDF) copies of SIMS assignment cover pages


2.3 Extensions

If you believe that your assignment will be delayed because of circumstances beyond your control such as illness, you should apply for an extension prior to the due date. All applications for extensions must be made in writing to your lecturer. Medical certificates or other supporting documentation will be required.

Late assignments submitted without an approved extension may be accepted up to one week late at the discretion of your lecturer, but will be penalised at the rate of 10% of total assignment marks per day (including weekends).

Example:
Total marks available for the assignment = 100 marks
Marks received for the assignment = 70 marks
Marks deducted for 2 days late submission (20% of 100) = 20 marks
Final mark received for assignment = 50 marks

2.4 Submission of assignments

All assignments are to be submitted to your tutor during your allocated tutorial.

2.5 Return of assignments

Assignments will either be returned in specified tutorials during semester or via the SIMS Frontdesk collection system outside semester.

In general, assignments will be returned within two to three weeks of the due date.
3  **Student Academic Grievance Procedure**

If you have a concern or issue about aspects of your assessment or other academic matters, you are encouraged to follow the SIMS Student Academic Grievance Procedure: [http://www.sims.monash.edu.au/policies](http://www.sims.monash.edu.au/policies).

4. **Pass requirements**

The 40% rule applies to units and determines the final result for a student where the student's performance in either the examination or assignment component of the unit is unsatisfactory. Students need to be aware of the 40% rule which is:

In order to pass a unit, a student must gain all of the following:

- at least 40% of the marks available for the examination component: i.e. the final examination and any tests performed under exam conditions, taken as a whole
- at least 40% of the marks available for the assignment component: i.e. the assignments and any other assessment tasks (such as presentations) taken as a whole
- at least 50% of the total marks for the unit

Where a student gains less than 40% for either the examination or assignment component, the final result for the unit will be no greater than '44-N'.

5. **Grades**

The grades awarded by the Faculty of Information Technology are:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Code</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Distinction</td>
<td>HD</td>
<td>80-100</td>
</tr>
<tr>
<td>Distinction</td>
<td>D</td>
<td>70-79</td>
</tr>
<tr>
<td>Credit</td>
<td>C</td>
<td>60-69</td>
</tr>
<tr>
<td>Pass</td>
<td>P</td>
<td>50-59</td>
</tr>
<tr>
<td>Fail</td>
<td>N</td>
<td>0-49</td>
</tr>
<tr>
<td>Near Pass</td>
<td>NP</td>
<td>45-49 (may be awarded by Board of Examiners only)</td>
</tr>
<tr>
<td>Deferred</td>
<td>DEF</td>
<td>-</td>
</tr>
<tr>
<td>Withheld</td>
<td>WH</td>
<td>-</td>
</tr>
</tbody>
</table>