UNIVERSITY OF MALTA
Board of Studies for
Information Technology
B.Sc. I.T. (Hons.) Course

Department of Computer Information Systems
Department of Computer Science and Artificial Intelligence
Department of Communications and Computer Engineering

Course Catalogue
Academic Year 2005/2006

The Board of Studies reserves the right to withdraw, change or enhance units described in this catalogue.
1. Introduction .................................................................................................................. 3
2. The IT Disciplines ....................................................................................................... 3
   2.1 Informatics ............................................................................................................... 3
   2.2 Computer Science ................................................................................................. 4
   2.3 Communication & Computer Systems Engineering .............................................. 4
3. The Departments ......................................................................................................... 5
4. The Degree Programme ............................................................................................... 5
5. Summary of Credits ..................................................................................................... 7
   5.1 Year 1, Semesters 1 & 2 ....................................................................................... 7
   5.2 Year 2, Semesters 1 & 2 ....................................................................................... 9
   5.3 Year 3, Semesters 1 & 2 ....................................................................................... 11
   5.4 Year 4, Semesters 1 & 2 ....................................................................................... 13
6. Description of Study-Units .......................................................................................... 15
   6.1 Units in Informatics (Year I) .................................................................................. 15
   6.2 Units in Computer Science (Year I) ....................................................................... 19
   6.3 Units in Communication & Computer Systems Engineering (Year I) .............. 24
   6.4 Additional Units Advised by the Board (Year I) ................................................... 27
   6.5 Units in Informatics (Year II) ............................................................................... 29
   6.6 Units in Computer Science (Year II) .................................................................... 34
   6.7 Units in Communication & Computer Systems Engineering (Year II) ......... 41
   6.8 Units Advised by the Board ................................................................................ 47
   6.9 Units in Informatics (Year III) ............................................................................. 50
   6.10 Units in Computer Science (Year III) ................................................................. 53
   6.11 Units in Communication & Computer Systems Engineering (Year III) ....... 61
   6.12 Units in Informatics (Year IV) ........................................................................... 67
   6.13 Units in Computer Science (Year IV) ................................................................. 71
   6.14 Units in Communication & Computer Systems Engineering (Year IV) ....... 76
7. Student Guide ............................................................................................................... 81
   Course Regulations (1995) (Appendix A) ................................................................. 82
   Course Structure (1995) (Appendix B1) ................................................................. 84
   Course Bye-Laws October 2003 (Appendix B2) ................................................... 88
1. Introduction

Information Technology (IT) is the broad umbrella which encompasses the information processes which socio-economic activity creates, the information systems which govern the efficient use of information as a fundamental resource, as well as a wide spectrum of increasingly convergent and linked technologies which process and disseminate information.

There are two main forces driving the demand for IT expertise in Malta:

• the development, operation, and maintenance of the local IT infrastructure to increase the island’s competitiveness in all sectors, and

• the birth and growth of an outward-looking IT industry to prospect for niche markets and emerging technology with an eye to value-added applications.

The use of IT in every facet of modern life has made it necessary for all graduates to be proficient in those aspects of IT which are particularly relevant to their respective field of study. For example, prospective teachers ought to know how to use IT tools in the classroom, architects how to use IT tools for design, managers how to use IT tools to support decision making etc. Ideally, such proficiency courses ought to be catered for by specialised staff; the use of IT as an instructional technology is a pedagogical issue which ought to be taught by an educationalist with the appropriate IT expertise; the use of CAD tools in architecture ought to be taught by an architect who uses these tools actively in architectural applications; the meaningful application of decision support systems in management ought to be taught in the context of management theory and methodologies etc.

Besides the need for relevant IT proficiency amongst all graduates there is a dire need for IT graduates who have a sound mathematical and theoretical foundation in the technical facets of Information Technology. The main emphasis for IT professionals is the deep understanding of underlying principles, and the ability to apply these principles in the management, specification, design, validation, and implementation of complex systems: information, software, and hardware. There should be a clear distinction between the means in a university to educate all graduates in the application of IT in their field of study and the means used to generate IT graduates.

2. The IT Disciplines

There are at least three technical streams which a prospective IT graduate can follow in the context of Malta’s growth as an information economy, namely: Informatics, Computer Science, and Communication and Computer Systems Engineering Stream.

2.1 Informatics

Informatics as a field of academic study encompasses the concepts, principles, and processes for two broad areas of activity within organizations: (1) acquisition, deployment, and management of information technology resources and services (the information systems function) and (2) development, operation, and evolution of infrastructure and systems for use in organizational processes (system development, systems operation, and system maintenance). The systems that deliver information and communications services in an organization combine both technical components and human operators and users. They capture, store, process, and communicate data, information, and knowledge.

The information systems function in an organization has a broad responsibility to plan, develop or acquire, implement, and manage an infrastructure of information technology (computers and communications), data (both internal and external), and enterprise-wide information processing systems. It has the responsibility to track new information technology and assist in incorporating it into organization’s strategy, planning, and practices. The activity of developing or acquiring information technology applications for organizational and inter-organizational processes involves projects that define creative and productive use of information technology for transaction processing, data acquisition, communication, coordination, analysis, and decision support. Design, development and acquisition, and implementation techniques, technology and
methodologies are employed. Processes for creating and implementing information systems in organizations incorporate concepts of business process design, innovation, quality, human-machine systems, human-machine interfaces, e-business design, sociotechnical systems, and change management.

Informatics professionals work with information technology and must have sound technical knowledge of computers, communications, and software. Since they operate within organizations and with organizational systems, they must also understand organizations and the functions within organizations (accounting, finance, marketing, operations, human resources, and so forth). They must understand concepts and processes for achieving organizational goals with information technology.

2.2 Computer Science

Computer Science is the science of programming that is, creating programs that tell computers what to do. A central concern of programming is the definition and implementation of algorithms, step-by-step procedures that are guaranteed to carry out a specific task or achieve a particular result in a finite amount of time.

The teaching of Computer Science intersects with the study of different areas including Logic, Engineering, and Interaction. Logic is needed to describe programs and to reason about their properties in a precise way, whilst Engineering applied to software supplies principles and tools that are necessary to master ever more complex programming tasks. The investigation of Interaction involves the elaboration of formalised protocols, which define patterns of communication, whether these are between components belonging to one computer, or between different computers belonging to one network. Widespread compliance to such protocols can have enormous effects that extend beyond mere hardware to the community of human users: witness the evolving social impact of Internet since its inception.

Computer Science also studies the problem of programming intelligent behavior. This branch of the subject, known as Artificial Intelligence, is concerned with getting machines to behave humanly: to learn and discover on the basis of experience; to speak, to hear, to hold conversations in English or Maltese; to solve complex, ill-specified problems. In short, Artificial Intelligence studies the problem of building intelligent agents that demonstrate human-like competence in determined areas.

Proficient Computer Scientists must transcend the latest programming fashions; they must acquire a deep understanding of the principles by which the algorithmic “substance” of computation can be crafted to solve a wide range of practical problems with elegance and economy.

2.3 Communication and Computer Systems Engineering

Computer and Telecommunication Systems Engineering is concerned with the design, development, implementation and maintenance of three major systems in Information and Communication Technology: computational, communication and digital signal processing systems.

Communication and Computer Systems Engineering deals with the construction of physical machines that carry out computations in an automated fashion: it is the hardware perspective of Information Technology. It also deals with software components that access the hardware directly. This discipline is developing at a rapid rate as new electronic components emerge that allow one to customise and construct simple, low-cost or complex architectures that match the application in hand. The quest is to optimise the computing performance or the cost of the system as well as provide the low-level interface for software systems and practical application tools. The challenges tackled are related to the creation and interfacing of machinery capable of handling the storage, processing, and distribution of massive volumes of data. This discipline therefore provides the platform on which other systems, such as communications and signal processing systems can be realised.

Digital Signal Processing (DSP) is a discipline concerned with the design and implementation of complex algorithms that process sampled data to yield a second set of more useful data. Originally DSP was applied extensively in large industrial systems such as radar systems, geophysical exploration, space
exploration and later medical equipment. Today DSP is finding new applications in consumer electronic systems such as audio and video systems, and in manufacturing for process control.

The area of communications engineering deals with the design of systems, such as telephone and data networks, that enable information to be transmitted from one place to another. These systems are made up of a number of hardware and software components that can be mixed and matched to provide the system requirements. All this must be placed within the framework of emerging international standards for system implementation.

A graduate in Computer and Telecommunication Systems Engineering must primarily understand the concepts and parameters that govern the design of these systems, and the techniques involved in their design, development, implementation and maintenance.

3. The Departments

The Board of Studies for I.T. is made up of the following three departments:

- the Department of Computer Information Systems,
- the Department of Computer Science and Artificial Intelligence, and
- the Department of Communications and Computer Engineering.

This report outlines the range of units offered by the three departments, and how these units can be grouped to offer various degree programmes.

- The three IT departments will be offering a suite of units which are used in several degree programmes.

4. The Degree Programme

B.Sc. IT(Hons.) in Informatics, Computer Science, or Communication and Computer Systems Engineering

This degree programme aims to produce graduates fit to occupy responsible positions in the Information Technology industry. It will be regulated by the Board of Studies for Information Technology appointed by Council for this purpose. Entry requirements for the course are detailed in Appendix A. The regulations for the courses which started prior to October 2003 are included in Appendix B1 and for courses which started in October 2003 or later are included in Appendix B2.

In the first two years (Part One), a core suite of units will be offered, covering Informatics, Computer Science, and Communication and Computer Systems Engineering with some compulsory units in Mathematics, Statistics, Operations Research, and Management. The aim in the first year is to give a broad perspective of various concepts in Information Technology and how they interrelate. The second year refines the spirit of the first by exposing students to foundation topics in some detail. At the end of the second year, students should be in a position to decide on a specific stream: Informatics, Computer Science, or Communication and Computer Systems Engineering.

In the third and fourth years (Part Two), a student is expected to concentrate on the specific stream chosen. Candidates who wish to complete a three-year degree or who do not qualify for the fourth year in this programme may obtain an ordinary degree in the stream chosen.

To be registered as a regular student in this degree programme a candidate must, besides satisfying the General Entry Requirements specified in the Admissions Regulations of the University, for courses starting in October 1997 and later, be in possession of a pass with Credit or Distinction in Pure Mathematics and another subject taken at Advanced level as part of the Matriculation Certificate Examination from the following:
candidates are advised to have a background in programming (ideally in Pascal) prior to joining the course.
5. **SUMMARY OF CREDITS**

5.1 **YEAR I**

STUDENTS ARE REQUIRED TO REGISTER FOR ALL CREDITS IN SEMESTERS 1 & 2 AS WELL AS FOUR OPTIONAL CREDITS FROM THE RECOMMENDED LIST OR FROM ANY FACULTY/INSTITUTE OF THE UNIVERSITY TO A TOTAL OF 60 CREDITS

### Semester 1

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compulsory</td>
<td>CCE1201 (IT only)</td>
<td>Introduction to Signal Processing</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Buttigieg; Prof P Micallef</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CCE1003 (IT only)</td>
<td>Introduction to Computer Systems</td>
<td>21</td>
<td>4 labs + .5 tut</td>
<td>3</td>
<td>Dr C J Debono</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CCE1101 (IT only)</td>
<td>Introduction to Communication Systems</td>
<td>21</td>
<td>4 labs + 5 tut</td>
<td>3</td>
<td>Dr A F Muscat</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CIS1021</td>
<td>Fundamentals of Information Systems</td>
<td>26</td>
<td>4</td>
<td>4</td>
<td>Prof A L Ganado; Mr R Naudi; Mr T Spiteri Staines</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CIS1041</td>
<td>Introduction to Databases</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Mr J Vella</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSA1010</td>
<td>Principles of Structured Development</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Dr. Ernest Cachia</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSA1011</td>
<td>Algorithmics</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Dr. John Abela, Mr. Joseph Cordina</td>
</tr>
<tr>
<td>Compulsory</td>
<td>MAT 1091</td>
<td>Mathematical Methods I: Matrices &amp; Differential Equations</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Dr I Sciriha; Mr F Bezzina</td>
</tr>
</tbody>
</table>

### Semester 1 & 2

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compulsory</td>
<td>CSA1060</td>
<td>Mathematics of Discrete Structures</td>
<td>24</td>
<td>4</td>
<td>4</td>
<td>Dr. Gordon Pace, Mr. Kristian Guillamier</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSA1080</td>
<td>Declarative Programming</td>
<td>24</td>
<td>4</td>
<td>4</td>
<td>Dr. Matthew Montebello, Mr. Sandro Spina</td>
</tr>
</tbody>
</table>

### Semester 2

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compulsory</td>
<td>CCE1002</td>
<td>Computer Logic &amp; Organization I</td>
<td>28</td>
<td>12</td>
<td>4</td>
<td>Prof P Micallef</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CIS 1061</td>
<td>Structured Design and Applied Operating Systems</td>
<td>28</td>
<td>7</td>
<td>4</td>
<td>Mr C Meli; Mr T Spiteri Staines</td>
</tr>
<tr>
<td>Type of Unit</td>
<td>Unit Code</td>
<td>Title of Unit</td>
<td>Lect hrs</td>
<td>Lab/tut/hrs</td>
<td>ECTS Credits</td>
<td>Lecturer/s</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td>--------------------------------------------</td>
<td>----------</td>
<td>-------------</td>
<td>--------------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CIS1071</td>
<td>Further Programming</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Nezval, Mr C Meli, Mr T Spiteri Staines, Mr J Galea</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSA1012</td>
<td>Object Oriented Programming</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Dr. John Abela, Mr. Joseph Cordina</td>
</tr>
<tr>
<td>Compulsory</td>
<td>MAT 1803 (IT only)</td>
<td>Mathematical Transform Techniques</td>
<td>10</td>
<td>4</td>
<td>2</td>
<td>Mr J Borg</td>
</tr>
</tbody>
</table>

**Recommended Optional Credits – Semester 1**

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optional</td>
<td>CIS1032 (IT only)</td>
<td>Introduction to Programming in Pascal</td>
<td>14</td>
<td>8</td>
<td>2</td>
<td>Dr V Nezval &amp; others</td>
</tr>
</tbody>
</table>

**Recommended Optional Credits – Semester 1 & 2**

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optional</td>
<td>CSA1013</td>
<td>Historical and Scientific Perspectives in CS &amp; A.I.</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Various</td>
</tr>
</tbody>
</table>

**Recommended Optional Credits – Semester 2**

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optional</td>
<td>CIS1051</td>
<td>Introduction to Business Modelling and Office Automation Software</td>
<td>26</td>
<td>4</td>
<td>4</td>
<td>Prof A Leone Ganado, Mr C Meli, Mr J Galea</td>
</tr>
</tbody>
</table>
5.2 YEAR II

STUDENTS ARE REQUIRED TO REGISTER FOR 60 CREDITS: 12 CREDITS IN EACH AREA OF SPECIALISATION; 12 CREDITS FROM ANY AREA OF SPECIALISATION; AND 12 CREDITS AS DIRECTED FROM THE BOARD.

Semester 1

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compulsory</td>
<td>CCE2001</td>
<td>Computer Logic &amp; Organisation II</td>
<td>28</td>
<td>7</td>
<td>4</td>
<td>Dr J Briffa</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CCE2112</td>
<td>Telecommunications I</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr A Muscat</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CIS2030</td>
<td>Systems Analysis &amp; Logical Design</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Prof L Ganado</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSA2070</td>
<td>Core Computer Science I</td>
<td>38</td>
<td>4</td>
<td>6</td>
<td>Dr. Ernest Cachia, Dr. Kevin Vella</td>
</tr>
<tr>
<td>Compulsory</td>
<td>SOR 1211</td>
<td>Probability</td>
<td>14</td>
<td>-</td>
<td>2</td>
<td>Mr L Camilleri</td>
</tr>
<tr>
<td>Compulsory</td>
<td>MGT2021</td>
<td>Business Management</td>
<td>28</td>
<td>-</td>
<td>4</td>
<td>Mr N Massa</td>
</tr>
<tr>
<td>Compulsory</td>
<td>MAT 2402</td>
<td>Networks</td>
<td>10</td>
<td>4</td>
<td>2</td>
<td>Prof J Lauri</td>
</tr>
<tr>
<td>Elective</td>
<td>CCE2101</td>
<td>Signal Analysis</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Buttigieg</td>
</tr>
<tr>
<td>Elective</td>
<td>CIS2131</td>
<td>Network &amp; Communication Issues in Information Systems</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Mr C Meli, Mr N Cutajar</td>
</tr>
<tr>
<td>Elective</td>
<td>CSA2010</td>
<td>Compiling Techniques</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Mr. Kristian Guillaumier</td>
</tr>
<tr>
<td>Elective</td>
<td>CSA2130</td>
<td>Object Oriented Modeling and Java</td>
<td>28</td>
<td></td>
<td>4</td>
<td>Mr. Patrick Abela, Dr. Ernest Cachia</td>
</tr>
</tbody>
</table>

Semester 1 & 2

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compulsory</td>
<td>CIS 2051</td>
<td>Physical Design &amp; Implementation of Object Oriented Applications</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Nezval</td>
</tr>
<tr>
<td>Elective</td>
<td>CIS2041</td>
<td>Information Systems Theory &amp; Practice</td>
<td>42</td>
<td>4</td>
<td>6</td>
<td>Prof A Leone Ganado, Mr T Spiteri Staines, Mr R Naudi, Mr S Caruana</td>
</tr>
<tr>
<td>Elective</td>
<td>CSA2090</td>
<td>Systems Programming</td>
<td>28</td>
<td></td>
<td>4</td>
<td>Dr. Christopher Staff, Mr. Joseph Cordina</td>
</tr>
</tbody>
</table>
### Semester 2

<table>
<thead>
<tr>
<th>Type of Unit</th>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect hrs</th>
<th>Lab/tut/hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compulsory</td>
<td>CCE2002</td>
<td>Microprocessor Systems</td>
<td>28</td>
<td>9</td>
<td>4</td>
<td>C J Debono</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CIS2090</td>
<td>Practical Design &amp; Implementation with DBMS</td>
<td>28</td>
<td>7</td>
<td>4</td>
<td>Mr J Vella</td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSA2080</td>
<td>Core Computer Science II</td>
<td>42</td>
<td></td>
<td>6</td>
<td>Dr. John Abela, Dr. Matthew Montebello, Dr. Gordon Pace</td>
</tr>
<tr>
<td>Compulsory</td>
<td>MAT 1401</td>
<td>Discrete Methods</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Prof J Lauri</td>
</tr>
<tr>
<td>Elective</td>
<td>CCE2122</td>
<td>Telecommunications II</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr A Muscat</td>
</tr>
<tr>
<td>Elective</td>
<td>CCE2301</td>
<td>Introduction to Matlab</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr J Briffa</td>
</tr>
<tr>
<td>Elective</td>
<td>CCE3012*</td>
<td>Introduction to Computer Networks</td>
<td>14</td>
<td>7</td>
<td>2</td>
<td>Prof P Micallef</td>
</tr>
<tr>
<td>Elective</td>
<td>CCE2105</td>
<td>Electromagnetic Theory</td>
<td>28</td>
<td>9</td>
<td>4</td>
<td>Dr A Muscat</td>
</tr>
<tr>
<td>Elective</td>
<td>CIS2150</td>
<td>Applications for e-business using Java</td>
<td>14</td>
<td>7</td>
<td>2</td>
<td>Dr V Nezval</td>
</tr>
<tr>
<td>Elective</td>
<td>CIS2120</td>
<td>Applied Graphic Design and User Application Standards</td>
<td>28</td>
<td>7</td>
<td>4</td>
<td>Mr C Meli</td>
</tr>
<tr>
<td>Elective</td>
<td>CSA2100</td>
<td>Techniques in Operating Systems</td>
<td>28</td>
<td></td>
<td>4</td>
<td>Dr. John Abela, Mr. Kristian Guillaumier, Mr. Joseph Cordina</td>
</tr>
<tr>
<td>Elective</td>
<td>CSA2050</td>
<td>Introduction to Computational Linguistics</td>
<td>14</td>
<td>4</td>
<td>2</td>
<td>Mr. Michael Rosner</td>
</tr>
<tr>
<td>Elective</td>
<td>PHY2150</td>
<td>Numerical Analysis</td>
<td>14</td>
<td>5</td>
<td>2</td>
<td>Prof A Buhagiar</td>
</tr>
</tbody>
</table>

* This unit is common with CCE 3002
5.3 **YEAR III**

**STUDENTS ARE REQUIRED TO REGISTER FOR 60 CREDITS: 30 CREDITS FOR 3 ASSIGNED PRACTICAL TASKS FROM THE CHOSEN STREAM OR TWO FROM THE CHOSEN STREAM AND ONE FROM ANY STREAM; ONE OF THE APTS HAS TO BE CARRIED OUT IN A GROUP - NORMALLY 3-5 STUDENTS; AND ANOTHER 30 CREDITS OF WHICH NOT LESS THAN 16 AND NOT MORE THAN 20 ARE TO BE TAKEN IN THE CHOSEN AREA OF SPECIALISATION.**

**ALL UNITS ARE ELECTIVE EXCEPT FOR THE ASSIGNED PRACTICAL TASKS WHICH ARE COMPULSORY**

### Semester 1

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCE3021</td>
<td>Real-Time Embedded Systems</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr C J Debono</td>
</tr>
<tr>
<td>CCE3201</td>
<td>Filters &amp; Digital Signal Processing</td>
<td>28</td>
<td>9</td>
<td>4</td>
<td>Dr V Buttigieg</td>
</tr>
<tr>
<td>CIS 3021</td>
<td>I.S. Strategy – Management and Practice</td>
<td>28</td>
<td>-</td>
<td>4</td>
<td>Prof A Leone Ganado, Mr R Naudi, Mr T Spiteri Staines, Mr S Caruana</td>
</tr>
<tr>
<td>CIS 3031</td>
<td>Scientific and Quantitative Aspects of I.S.</td>
<td>28</td>
<td>-</td>
<td>4</td>
<td>Prof A Leone Ganado, Mr J Galea</td>
</tr>
<tr>
<td>CIS 3051</td>
<td>Information Systems Engineering I</td>
<td>42</td>
<td>-</td>
<td>6</td>
<td>Mr R Naudi, Prof A Leone Ganado, Mr T Spiteri Staines</td>
</tr>
<tr>
<td>CIS 3090</td>
<td>Social &amp; Professional Issues for Computing</td>
<td>14</td>
<td>-</td>
<td>2</td>
<td>Prof A Leone Ganado, Dr C Staff et al</td>
</tr>
<tr>
<td>CIS 3081</td>
<td>Advanced Client/Server Computing &amp; Distributed Databases</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Nezval, Mr J Vella, Mr C Meli</td>
</tr>
<tr>
<td>CSA3030</td>
<td>Introduction to Computer Graphics</td>
<td>10</td>
<td>5</td>
<td>2</td>
<td>Mr. Sandro Spina</td>
</tr>
<tr>
<td>CSA3100</td>
<td>Computability and Complexity</td>
<td>12</td>
<td>2</td>
<td>2</td>
<td>Dr. Gordon Spina</td>
</tr>
<tr>
<td>CSA3150</td>
<td>Concurrent and Distributed Systems</td>
<td>38</td>
<td>4</td>
<td>6</td>
<td>Dr. Kevin Vella</td>
</tr>
<tr>
<td>CSA3160</td>
<td>Scientific Computing</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Mr. Sandro Spina, Mr. Joseph Cordina</td>
</tr>
<tr>
<td>CSA3170</td>
<td>Software Engineering</td>
<td>42</td>
<td>6</td>
<td>6</td>
<td>Dr. Ernest Cachia, Mr. Mark Micallef</td>
</tr>
<tr>
<td>CSA3180</td>
<td>Natural Language Processing</td>
<td>42</td>
<td>12</td>
<td>6</td>
<td>Mr. Michael Rosner</td>
</tr>
<tr>
<td>CSA3190</td>
<td>Internet Technologies and Java Server - Side Programming</td>
<td>24</td>
<td>8</td>
<td>4</td>
<td>Mr. Kristian Guillamier, Dr. John Abela, Mr. Patrick Abela</td>
</tr>
<tr>
<td>CSA3200</td>
<td>Adaptive Hypertext Systems</td>
<td>42</td>
<td>6</td>
<td>6</td>
<td>Dr. Christopher Staff</td>
</tr>
<tr>
<td>CSA3210</td>
<td>Agent Technology</td>
<td>42</td>
<td>6</td>
<td>6</td>
<td>Dr. Matthew Montebello, Mr. Charlie Abela</td>
</tr>
<tr>
<td>CSA3220</td>
<td>Machine Learning, Expert Systems and Fuzzy Logic</td>
<td>42</td>
<td>6</td>
<td>6</td>
<td>Dr. John Abela</td>
</tr>
<tr>
<td>ELE3103</td>
<td>Hardware Description Languages (HDL)</td>
<td>8</td>
<td>6</td>
<td>2</td>
<td>Dr I Grech</td>
</tr>
</tbody>
</table>
**Semester 2**

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCE3000-</td>
<td>Assigned Practical Tasks</td>
<td></td>
<td></td>
<td>10 EACH</td>
<td>Various</td>
</tr>
<tr>
<td>CCE3002</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCE 3022*</td>
<td>Network Modelling</td>
<td>14</td>
<td>7</td>
<td>2</td>
<td>Prof P Micallef</td>
</tr>
<tr>
<td>(IT only)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCE3102</td>
<td>Transmission Media for Communications</td>
<td>28</td>
<td>10</td>
<td>4</td>
<td>Dr Adrian Muscat</td>
</tr>
<tr>
<td></td>
<td>Systems</td>
<td></td>
<td></td>
<td></td>
<td>Mr P Debono</td>
</tr>
<tr>
<td>CCE3202</td>
<td>Advanced Digital Systems Design</td>
<td>14</td>
<td>14</td>
<td>4</td>
<td>Dr C J Debono</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Dr V Buttigieg</td>
</tr>
<tr>
<td>CCE3203</td>
<td>Digital Image Processing I</td>
<td></td>
<td></td>
<td>4</td>
<td>Dr K P Camilleri</td>
</tr>
<tr>
<td>CIS3000-</td>
<td>Assigned Practical Tasks</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>Various</td>
</tr>
<tr>
<td>CIS3002</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSA3000 -</td>
<td>Assigned Practical Tasks (APTs)</td>
<td></td>
<td></td>
<td>10</td>
<td>Various</td>
</tr>
<tr>
<td>CSA3002</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* This unit is common with CCE3002

**PROPOSED UNITS FOR YEAR IV UNDER NEW HARMONIZED REGULATIONS**

**Semester 1**

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSA4060</td>
<td>Formal Methods</td>
<td>24</td>
<td>4</td>
<td>4</td>
<td>Dr. Gordon Pace</td>
</tr>
</tbody>
</table>

**Semester 1 and 2**

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSA4000</td>
<td>Final Year Project (FYP)</td>
<td></td>
<td></td>
<td>20</td>
<td>Various</td>
</tr>
</tbody>
</table>
### 5.4 YEAR IV
(For courses which commenced in October 2002)

**Semester 1**

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCE 4001</td>
<td>Advanced Computer Architecture</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr C J Debono</td>
</tr>
<tr>
<td>CCE 4013</td>
<td>Real-Time Networks</td>
<td>28</td>
<td>4</td>
<td>4</td>
<td>Dr C J Debono</td>
</tr>
<tr>
<td>CCE 4004</td>
<td>Computer Networks II</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr S Zammit</td>
</tr>
<tr>
<td>CCE 4101</td>
<td>Communication Systems</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr A Muscat</td>
</tr>
<tr>
<td>CCE 4102</td>
<td>Coding for Communication Systems</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Buttigieg</td>
</tr>
<tr>
<td>CCE 4201</td>
<td>Digital Signal Processing</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Prof P Micallef</td>
</tr>
<tr>
<td>CIS 4020</td>
<td>Developing and Managing IS/IT Operations</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Prof A Leone Ganado</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Mr C Meli</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Mr T Spiteri Staines</td>
</tr>
<tr>
<td>CIS 4050</td>
<td>Information Systems Engineering</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Mr R Naudi</td>
</tr>
<tr>
<td>CSA4010</td>
<td>Concurrent and Distributed Systems II</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Dr. Kevin Vella</td>
</tr>
<tr>
<td>CSA4030</td>
<td>Scientific Computing</td>
<td>28</td>
<td></td>
<td>4</td>
<td>Mr. Sandro Spina, Mr. Joseph Cordina</td>
</tr>
<tr>
<td>CSA4040</td>
<td>Advanced Topics in Software Engineering</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Dr. Ernest Cachia</td>
</tr>
<tr>
<td>CSA4050</td>
<td>Advanced Topics in Natural Language Processing</td>
<td>20</td>
<td>8</td>
<td>4</td>
<td>Mr. Michael Rosner</td>
</tr>
<tr>
<td>CSA4060</td>
<td>Formal Methods</td>
<td>24</td>
<td>4</td>
<td>4</td>
<td>Dr. Gordon Pace</td>
</tr>
<tr>
<td>CSA4070</td>
<td>Machine Learning and Pattern Recognition</td>
<td>24</td>
<td>4</td>
<td>4</td>
<td>Dr. John Abela</td>
</tr>
<tr>
<td>CSA4080</td>
<td>Adaptive Hypertext Systems II</td>
<td>28</td>
<td></td>
<td>4</td>
<td>Dr. Christopher Staff</td>
</tr>
<tr>
<td>CSA4110</td>
<td>Intelligent Software Agents</td>
<td>28</td>
<td></td>
<td>4</td>
<td>Dr. Matthew Montebello, Mr. Charlie Abela</td>
</tr>
<tr>
<td>CIS 4080</td>
<td>Advanced Client/Server Computing: The Object Oriented Approach</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Dr V Nezval</td>
</tr>
<tr>
<td>SOR 1341</td>
<td>Dynamic Programming</td>
<td>14</td>
<td></td>
<td>2</td>
<td>Dr J Sklenar</td>
</tr>
<tr>
<td>SOR 1351</td>
<td>Programming in Simula</td>
<td>14</td>
<td></td>
<td>2</td>
<td>Dr J Sklenar</td>
</tr>
</tbody>
</table>

**Semester 1 and 2**

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIS 4000</td>
<td>Final Year Project</td>
<td>20</td>
<td></td>
<td></td>
<td>Various</td>
</tr>
<tr>
<td>CSA4000</td>
<td>Final Year Project</td>
<td>20</td>
<td></td>
<td></td>
<td>Various</td>
</tr>
<tr>
<td>CCE 4000</td>
<td>Final Year Project</td>
<td>20</td>
<td></td>
<td></td>
<td>Various</td>
</tr>
<tr>
<td>CIS 4010</td>
<td>Advanced Databases: Data Models and Database Languages</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Mr J Vella</td>
</tr>
<tr>
<td>CIS 4040</td>
<td>I.S. Strategic Analysis and Business Implications</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Prof A Leone Ganado</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Mr R Naudi</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Mr T Spiteri Staines</td>
</tr>
</tbody>
</table>
Semester 2

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Title of Unit</th>
<th>Lect/ Hrs</th>
<th>Lab/tut/ hrs</th>
<th>ECTS Credits</th>
<th>Lecturer/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELE 4607</td>
<td>Introduction to Digital VLSI</td>
<td>28</td>
<td>14</td>
<td>4</td>
<td>Prof J Micallef</td>
</tr>
</tbody>
</table>

The Board reserves the right to withdraw, change or enhance options from the above list in any academic year depending upon the availability of teaching resources and number of students. Each department has a policy of continuing revision and updating of courses, so they may not be in exactly the form specified.
6. DESCRIPTION OF STUDY-UNITS

YEAR I

6.1 UNITS IN INFORMATICS

► CIS1032 Introduction to Programming in Pascal

Lecturers: Dr V. Nezval, Mr. C. Meli, Mr. T. Spiteri Staines, Mr. J. Galea, Ms. A. Giuliano
Prerequisites: ‘A’ Level Grade C (Computing or I.T.) or following a computing course
Leads to: CIS1071, CIS1061, CIS2120
Credits: 2
Lectures: 14
Tutorials/Practical Labs: 8
Semester: 1st

This unit is an intensive introductory course in programming using the procedural paradigm. The accent is given to programming techniques understanding, key algorithms and writing programs in Pascal. High importance is given to practical problem solving.

The first part concentrates to Pascal programming with the following topics:
Basic programming algorithms, basic language constructs (if/then, case, loops), basic data structures (including arrays, records, variant records and free unions) principles of modular programming (functions, procedures) and text file handling.

In the second part the emphasis is given to important, more advanced algorithms: Random number generation, Sorting (Insertion, Bubble, Selection Sort, Quicksort and Mergesort), Searching (linear, hashing), higher structures based on arrays (lists and stacks).

Method of Assessment:
Test (80%) Project (20%)

Textbooks:
- P Grogogno Programming in Pascal Addison-Wesley
- CIS1032 website at http://www.cis.um.edu.mt/staff/cmeli/other/cis1032
- comp.lang.pascal.borland and malta.comp USENET newsgroups
- Timo Salmi’s Pascal FAQs at http://www.uwasa.fi/~ts

► CIS1021 Fundamentals of Information Systems

Lecturer: Prof. A. Leone Ganado, Mr. R. Naudi, Mr T. Spiteri Staines
Prerequisites: Following a computing course
ECTS Credits: 4
Lectures: 26
Tutorials/Practical Labs: 4
Semester: 1st

The unit provides an introduction to systems theory concepts, information technology, and application software. It focuses on the way information is utilised as a critical resource in organisations. Issues such as information flows, quality and security of information and and its logistic availability will be discussed.
Introduction to the basic concepts of systems theory. Components of an information system. Characteristics of Data and Information. Cost/value and quality of information. Types of information systems. Business management and reporting functions. Transaction processing systems, Office Information systems including groupware, Management information systems.

Information Administration Issues, Role and Nature of IS managers and management.

Functional Information Systems including financial, marketing, manufacturing and human resources systems.

The structure and components of decision support and executive information systems. Their role within organizational decision making.


Basic issues on the strategic use of IT. Competitive advantage, distributed processing, business process re-engineering and the benefits and challenges offered by E-Business.

The role and scope of communication and networking services as a means to increase information flow and business productivity.

Introduction to professionalism and ethical issues in IS.

Real world examples and case analysis application of the above will be included.

**Method of Assessment:**
Test (85%) Group Assignment (15%)

**Textbooks:**
- Effy OZ, *Management Information Systems* Thomson

► **CIS1041 Introduction to Databases**

Lecturer: Mr. J. Vella
Prerequisites: ‘A’ Level Grade C (Computing or I.T.) or following a computing course
Leads to: CIS2090
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 8
Semester: 1st

The aim is to introduce students to the concepts of good database design and to teach basic skills and techniques in querying databases.

Introduction to Relational Databases: The relational model, domains and relations, data integrity.

Introduction to more sophisticated modeling techniques: integrity constraints and triggers. Complementing SQL with procedural constructs. The cursor data structure.

Database and Database Management Systems topics: introduction to query processing and concurrency issues, and data dictionary.

Basic introduction for building an application program with a Fourth Generation Language (4GL).

Use of a relational database management systems such as Oracle is made though a number of practical / presentation. The lecturer provides unit material.

**Method of Assessment:** Test (85%) Practical (15%)

**Textbooks**

- R Earp & S Bagui, *Learning SQL*, Addison-Wesley
- Oracle’s systems manuals
- A manual for a 4GL will be decided in class.

**CIS1051 Business Modelling and Office Automation Software**

Lecturer: Prof A. Leone Ganado, Mr C. Meli, Mr J. Galea
Leads to: CIS3070
ECTS Credits: 4
Lectures: 26
Tutorials/Practical Labs: 4
Semester: 2nd

The aim of the unit is to develop numeracy and practical computing skills and apply them to business problems. The importance of associating quantitative measures whether of a deterministic or probabilistic nature will be emphasised.

Students will also be encouraged to utilise office and modelling software packages to solve problems after they have been suitably formulated and to interpret correctly the output provided by such packages.

Course Contents:

- Decision trees and Decision models.
- Financial and Investment models. Linear programming techniques as a business tool
- Network analysis and PERT techniques in IT and project management.
- Simple forecasting techniques.
- Introduction to simulation in IT.
- Complexity of algorithms and use of parallel processes to solve these models will also be discussed
- Sharing, integrating and communicating data among various applications, with special reference to OLE, Active X, Com Objects, .Net, Mono, ODBC. Groupware.
- The necessity of promoting proper housekeeping practices in terms of data management and security.
- The role of a help and information desk.
- Selecting the appropriate tools and operating platform. Endusers and workflow.
- The electronic office: email, teleconferencing, videoconferencing, personal organizers for enterprise-wide communication mobile computing.
- Information retrieval systems: teletext, videotext, online databases, Internet, Intranets.

Case studies are presented to discuss issues raised above.

**Method of Assessment:**
Test (85%) Project (15%)
Textbooks:
- T Lucey *Quantitative techniques* DP Publications
- V Lofti C C Pegels *Decision Support Systems for Management Science* IRWIN
- Chapell D, *Understanding Active X and OLE*, Microsoft Press
- Relevant software manuals.
- Lecture Handouts

▶ **CIS1061 Structured Design and Applied Operating Systems**

Lecturer: Mr. C. Meli, Mr T. Spiteri Staines  
Prerequisites: CIS1031  
Leads to: CIS2010, CIS2030  
ECTS Credits: 4  
Lectures: 28  
Tutorials/Practical Labs: 7  
Semester: 2nd

The aim is to teach students the basic skills required for good programming and to relate program design in the broader framework of information systems development and practical skills in using and understanding facilities provided by modern operating systems. The main theory and principles involved will be analysed.

The first part on program design will include:  
Aims of Modularity. Program Complexity Measures e.g. McCabe, Coupling and Cohesion and Bottom-up Design and Shared Modules. Functional Decomposition and JSP (Jackson Structured Programming or Michael Jackson program design method). Introduction to Object-Oriented Design using the Class-Responsibility-Collaborator (CRC) design technique.

The second part on OS will include:  
Microsoft Windows (W9X, NT, 2000, XP), and more recent ones & UNIX. Evolution of operating systems, basic concepts of operating systems as a shell between the user and the hardware (concurrency issues, way of running executable programs), pipes, redirection, writing simple shell scripts, file systems and file manipulation, and basic & advanced system utilities (for example editors, text handling).

**Method of Assessment:**  
Test (85%) Practicals (15%)

Textbooks:  
- King and Pardoe *Program design using JSP* Macmillan. ISBN 0470202319.  
- Either  
- H Hahn *A Student’s Guide to UNIX*, McGraw-Hill Book or  
- UNIX Wizards *UNIX Unleashed*, SAMS Publ.
CIS1071 Further Programming

Lecturers: Dr. V. Nezval, Mr. C. Meli, Mr. T. Spiteri Staines, Mr. J. Galea
Prerequisites: CIS1031 or CIS1061
Leads to: CIS2010, CIS2050, CIS3080
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 2nd

The aim of this unit is to write well structured, modular, programs in Pascal using appropriate data structures and programming techniques. It will also show how some concepts taught in CIS1061 can be implemented. In the second part of the unit the pointers (and their applications in dynamic data structures) are included as necessary structures for new programming techniques, mainly object oriented programming.

Contents of Unit (in brief):
Procedures and Functions (revised and extended), program modularity (units and libraries), subrange and enumerated data types, sets, variant records and free unions, recursive technique, sorting and searching techniques.
Pointers and dynamically linked data structures such as lists, queues and trees. Implementation and solution of problems involving the above data structures and their applications. Data abstraction and generic subroutines.

Three types of Data files and their basic handling methods with orientation to business problems, typed files (editing, indexing and hashing techniques, handling huge files), untyped files (structure, basic features and use).

Method of Assessment:
Test (80%) Project (20%)

Textbooks:
- W Feibel Turbo Pascal 7 Handbook McGraw Hill
- P McBride Turbo Pascal Programming Made Simple Whitaker ISBN 0750632429
- B Chandra Advanced Turbo Pascal with Graphics and Object Oriented Programming Cloth. Narosa Publishing House (Whitaker UK)
- R Foster Programming Techniques 1 & 2 (Pascal V.7) Eastern House (OEC of TAFE) ISBN 1 875794 70 0 (Thorpe AUS)

6.2 UNITS IN COMPUTER SCIENCE

CSA1010 - Principles of Structured Development

Lecturer: Dr. Ernest Cachia
Semester: 1st
Leads to: CSA2070
ECTS Credits: 4
Tutorials / Practicals: 4 Hrs
Lectures: 28 Hrs

It should be stressed that this unit is not in any way a “programming” course. It could, however, expose students to some very limited practical programming in the form of examples to consolidate theoretical discussion. This unit introduces the basic concepts of imperative programming languages from the
viewpoint of algorithmic structures and computation. Such notions as state, variable, and transition through Coursework and input; the notion of syntax and semantics, data declaration and usage, representation of control flow, basic programming constructs (namely sequence, selection, and repetition), as well as sub-routine control and referencing environments. Structure theorems and their application will also be introduced and discussed. As its second part, this unit will introduce a rigorous structured approach to program specification and design using simple universal concepts, and guidelines applied in an ever-increasing spectrum of development environments. The unit will introduce the student to such basic concepts as functional connectivity through data flow, modularity, module structure and relationships, through basic development paradigms such as stepwise refinement and levels of abstraction. Throughout this unit, the student will be exposed to various standard analysis and design paradigms and notations. All the principles discussed in this unit are accompanied by practical examples whenever possible.

Method of Assessment: Test: (80%) Coursework: (20%)

Textbooks:

Additional Textbooks:

CSA1060 – Mathematics of Discrete Structures

Lecturers: Dr. Gordon Pace, Mr. Kristian Guillaumier
Semester: 1st & 2nd
Leads to: CSA2080
ECTS Credits: 4
Tutorials / Practicals: 4 Hrs
Lectures: 24 Hrs

The course is primarily aimed to introduce the basic mathematical tools that are required for the formal and rigorous treatment of the various aspects of computing. The importance of formal reasoning is emphasised in the course, concentrating on syntax, and formal proofs. The course also explains various mathematical notions and structures that will be used in later courses.

Syllabus:
- Propositional Calculus
- Predicate Calculus
- Set theory
- Relations and Functions
- Natural Numbers and cardinality
- Group theory
- Graph theory

This unit is also intended to introduce the concept of logic as a tool for studying the validity of arguments.

Topics include an introduction to:
- Predicate and propositional logic.
- Logical equivalence and satisfiability.
- The syntax of First Order Logic.
- Axioms and inference rules.
- Proof systems and techniques.
- Set theory.

**Method of Assessment:**

<table>
<thead>
<tr>
<th>Test Semester 1</th>
<th>Test Semester 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(30%)</td>
<td>(70%)</td>
</tr>
</tbody>
</table>

**Textbooks:**

- Andrew Simpson, Discrete Mathematics by example, McGraw-Hill.

**CSA1080 – Declarative Programming**

Lecturers: Dr. Matthew Montebello, Mr. Sandro Spina

Semester: 1st & 2nd

Leads to: CSA2080

ECTS Credits: 4

Tutorials / Practicals: 4 Hrs

Lectures: 24 Hrs

The first part of the course pursues the theme of logic programming by offering a comprehensive introduction to the syntax, semantics and features of Prolog, a well-known logic programming language that has been used extensively in a wide variety of AI application areas. The approach will be organized around a series of carefully chosen laboratory exercises.

The second part of the course gives an introduction to the functional programming paradigm, using Haskell. The course starts by explaining notions such as currying, strong typing, pattern matching, higher-order functions and advanced functional datatypes using practical examples. Evaluation order is discussed to explain the difference between call-by-value and call-by-name strategies, concentrating on lazy evaluation, the strategy used in Haskell. The practical part of the course finishes with an overview of type classes. This is then followed by a short overview of how we can prove properties of functional programs, from basic rewrite proofs to proofs requiring list and structural induction. The course will be complemented by a number of practical sessions.

**Syllabus:**

- Introduction: Notation, Currying, Basic Types, Pattern Matching, Referential Transparancy.
- Lists and Tuples
- Functional IO
- Higher-Order Functions
- Evaluation Order and Lazyness
- fold, foldr, scanl, ...
- Advanced Types: Enumerated types, parametrised types, data vs type.
- Type Classes
- Embedded Languages
- Proving properties about functions
- Proving properties of list functions
- Proving properties of parametrised types
Method of Assessment: Test: (80%) Coursework: (10% - SS, 10% - MM)

Textbooks:
- Website: http://staff.um.edu.mt/mmon1/lectures/csa1080/

CSA1011 – Algorithmics

Lecturers: Dr. John Abela, Mr. Joseph Cordina
Semester: 1st
Leads to: CSA1012
ECTS Credits: 4
Tutorials / Practicals: None
Lectures: 28 Hrs

This unit introduces and elaborates on the theory and use of algorithms. This course is offered as a two-part unit, with each part being taught concurrently to the other. The 1st part of the course is an introduction to computer science in general. It emphasizes concepts making up a programming language including Turing Machines and Turing powerful languages, programming constructs, and the theory of algorithms. It also introduces several paradigms such as imperative, object-oriented and declarative programming. A pseudo-code approach is used to exemplify these paradigms. This course concludes with an overview of what makes up Computer Science as a field on its own.

The 2nd part of the unit introduces the concepts of algorithms and data structures, highlighting the relation that exits between the two. The concepts are introduced in a gradual fashion, proceeding from abstract principles to concrete examples. Correctness and efficiency will be emphasized as the main properties of algorithms. A number of algorithms will be discussed, with emphasis on sorting, searching, graph and tree processing, and hashing techniques. Abstract data types (ADTs) will be formally defined and illustrated with case studies for list, stack, queue, priority queues, and heaps.

Method of Assessment: Test: (90%) Coursework: (10%)

Textbooks:

Optional Textbooks:
- Data Structures and Algorithm Analysis, Mark Allen Weiss, Benjamin Cummings, ISBN 080539057X.
CSA1012 – Object Oriented Programming

Lecturers: Dr. John Abela, Mr. Joseph Cordina
Semester: 2\textsuperscript{nd}
Prerequisites: CSA1011
Leads to: CSA2090, CSA2130
ECTS Credits: 4
Tutorials / Practicals: None
Lectures: 28 Hrs

The aim of this unit is to cover and explain the concepts of OOP using Java or C# as an example programming language and show the power of OOP through the use of advanced algorithms. This course is made up of two parts (that will not be taught following each other in time), with the first part of the course focusing first on the Java or C# programming language whereby the use of objects will be emphasized. The use of pre-defined classes will also be discussed allowing the use of libraries, file I/O, etc. Secondly emphasis is placed on the object-oriented properties of Java or C# to construct classes, and why and how inheritance, polymorphism, abstract classes and interfaces are used.

The second part deals mainly with advanced algorithms. The ‘Big O’ notation will be introduced as a formal framework for describing the resource use (i.e. time and space) of an algorithm. Other topics covered include: graphs and their associated searching and traversal algorithms, hashing techniques, encryption and cryptographic algorithms, AVL trees, 2-3 trees, and B-trees. A good knowledge of programming and problem-solving is required for this unit. Code examples will be presented in a modern object-oriented programming language such as Java or C#.

\textbf{Method of Assessment:} \quad \text{Test: (75\%)} \quad \text{Coursework: (15\% \text{ Part 1} \& 10\% \text{ Part 2})}

\textbf{Textbooks:}

- C# How to Program - Deitel H. & Deitel P. (Prentice Hall) ISBN 0130622214

\textbf{Optional Textbooks:}

- Data Structures and Algorithm Analysis, Mark Allen Weiss, Benjamin Cummings, ISBN 080539057X.

CSA1013 – Historical and Scientific Perspectives on CS & AI
(Recommended Optional Credit to B.Sc. I.T. Students)

Lecturers: CSAI Lecturers
Semester: 1\textsuperscript{st} \& 2\textsuperscript{nd}
Leads to: None
ECTS Credits: 4
Tutorials / Practicals: 8 Hrs
Lectures: 20 Hrs
This study unit has two aims: (a) to give the student a sense of perspective regarding currently accepted concepts of Computer Science and Artificial Intelligence and (b) to develop the student's ability to research and report in these fields. During the one part of the course, a series of lectures given by different members of staff will trace developments in key areas including logic, reasoning, machine learning, operating systems, human language understanding, internet, computer architecture, information retrieval. The other part of the course will concentrate on developing the student's understanding of the methods and practices used for carrying out scientific investigations, writing up and disseminating results. Students will be assessed by coursework which will involve a literature review and presentation in an area chosen from the first part of the course.

Method of Assessment: Coursework: (75%) Presentation: (25%)

Textbooks:

- Website: http://www.cs.um.edu.mt/~gpac1/Teaching/History

6.3 UNITS IN COMMUNICATION & COMPUTER SYSTEMS ENGINEERING

► CCE 1201 Introduction to Signal Processing
Lecturer: Dr. V. Buttigieg / Prof. P. Micallef
Prerequisites: None
Leads to: CCE 2101
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14 hrs
Semester: 1

Objective
This course provides an introduction to applied signal processing. The student is introduced to various real-life signals, ways of detecting and sampling these signals, simple calculations of signal parameters and simple ways of processing and manipulating these signals.

Signals and their parameters
What is a signal? Different types of signals; Electrical signals, Acoustic signals, Electromagnetic signals, vibrations and biological signals.
Signals in Frequency Domain (FD).
Frequency and time period.
Signals as a summation of sinusoidal waves.
Hand sketching of signals in TD and FD; square waves, saw-tooth.
Examples of signals in computer systems (data on buses) and in Communication systems (Ethernet signals or mobile phone signals).
Transducers to convert signals into electrical signals:
Acoustic transducers (Microphone and Speaker), Position, Speed, Temperature and Pressure sensors.
Power of signals, Energy in signals.
Simple power and energy calculation in various practical signals.

Signals Processing components
Amplifiers: gain and bandwidth.
Filters: phase and amplitude.
Practical examples in acoustics.

**Lab sessions in Signal processing**  
Measurements on oscilloscope and spectrum analysers.  
Use of simple LF filters and amplifiers.  
Fourier transform on Simulink  
Addition and multiplication of simple signals on Simulink.

**Method of Assessment:**  
Examination (80%) Coursework (20%)

**Textbook:**

**CCE 1003 – Introduction to Computer Systems**

*Credits: 3 ECTS  
Lectures: 1.5 hrs/wk.  
Tutorials: 0.5 hrs/wk.  
Lab: 4 hrs.  
Prerequisite: None  
Leads to: CCE 1002, CCE 2001, CCE 2002  
Lecturer: Dr Ing. Carl James Debono*

**Semester: 1**

**Electrical Technology Concepts**
- DC voltages and currents
- Resistance, Capacitance and Inductance
- AC voltages and Currents
- Power
- Phasor Diagrams

**Electronic Devices**
- Diode Characteristics
- Zener Diode characteristics
- Transistor characteristics
  - BJT
  - MOSFET

**Electronic Circuits Concepts**
- The power supply
- Transistor operation as a switch
- Switching considerations
- Construction of Logic gates

**Computer Systems**
- Structure and Functions
- CPU
- Memory
- Peripherals
- Interconnection requirements
- Programming requirements
- Practical examples

**Assessment**
Coursework – 20%  
Exam – 80%

**Further Reading:**
CCE 1101 – Introduction to Communication Systems

Credits: 3 ECTS
Lectures: 1.5 hrs/wk.
Tutorials: 0.5 hrs/wk.
Lab: 4 hrs.
Prerequisite: None
Leads to: CCE2112
Lecturer: Dr Ing. Adrian F. Muscat
Semester: 1

What are telecommunication systems?
   Classification of communication systems
   What is Information
   The need for protocols

Voice, Video and Data signals
   Electrical Signals and Transducers
   Band-Limited Signals and Noise
   Digital Signals

Some simple communication systems
   Baseband Analogue transmission
   Baseband Digital transmission
   A note on digital systems in noise
   Analogue Amplitude Modulation
   The Communication System Model

More complex communication systems
   Frequency Division Multiplexing

Networks
   Circuit Switching - the PSTN
   Packet Switching – Data network
   Medium Sharing via a contention based protocol
   Network Topologies

Case studies
   Cable television and Cable modems
   Wireless telephony
   ADSL link

Instrumentation
   Voltage measurements
   Time domain measurements
   Oscilloscope controls
   Some more examples

Assessment
Coursework – 20%  Exam – 80%

Course Notes will be provided.
CCE1002 Computer Logic & Organisation I

Lecturer: Prof. P. Micallef
Prerequisites: None
Leads to: CCE2001
Credits: 4
Lectures: 28
Tutorials/labs: 12 hours
Semester: 2

- Computer Logic: This unit introduces boolean logic and functions for use in combinational logic circuits. Tradeoffs in implementation using NOR-logic or NAND-logic are discussed. Design and implementation using various minimisation techniques including Karnaugh Maps and tabular methods.
- Computer Arithmetic: Computer arithmetic for addition and subtraction including number representation, fixed and floating point standards, valid range and error checking.
- Computer Organisation: Basic computer and CPU organisation. Introduction to the components making up a processor - internal registers, ALU, internal bus, multiplexers and decoders. Relations to external system units - memory and input/output.
- The basis of sequential computing: Instructions, fetch, and execute cycle. Addressing Modes, Special Registers, Memory Stack. The need of control signals to sequence ALU operations.
- Computer arithmetic structures: serial and parallel full-adder. The Arithmetic Logic Unit design using a parallel full adder.

Method of Assessment:
Examination (90%) Coursework (10%)

Textbooks:
- A set of notes will be used.

Reading List:

6.4– ADDITIONAL UNITS AS ADVISED BY THE BOARD

MAT1091 Mathematical Methods I: Matrices and Differential Equations

Lecturers: Dr I. Sciriha/Mr Frank Bezzina
Follows from: A-Level
Credits: 4
Lectures: 20
Tutorials: 8
Semester: 1

- Matrices
- Determinants
- Eigenvalues
- Ordinary differential equations of the first order;
- Ordinary differential equations of the second order with constant coefficients;
- Partial differentiation and exact differential equations.
Textbooks:

Method of Assessment: Exam (100%)

MAT1803 Mathematical Transform Techniques

Lecturer: Mr J. Borg
Prerequisites: MAT1001, MAT1002
Semester: 2
Credits: 2
Lectures: 10
Tutorials: 4

- Laplace Transforms
- Fourier Series
- Fourier Transforms

Textbooks

Method of Assessment: Exam (100%)
YEAR II

6.5 UNITS IN INFORMATICS

CIS2030 Systems Analysis and Logical Design

Lecturer: Prof. A. Leone Ganado
Prerequisites: CIS1061 or CIS1071
Leads to: CIS2041, CIS2051, CIS2090
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 4
Semester: 1st

Communication, presentation, and interviewing skills. Structured walk-throughs. JAD sessions

The Software Development life-cycle including the business, socio-technical and organisational aspects.


System Development methodologies: The SSADM, DSDM, SSM and object-oriented development frameworks including the UML method.

Method of Assessment:
Test (100%)

Textbooks
- P Weaver Practical SSADM4 Pitman
- Jennifer Stapleton, DSDM Addison-Wesley
- David Brown, Object-Oriented Analysis John Wiley
- Joseph Schmiller UML in 24hrs Sams Publishing

► CIS2041 Information Systems Theory and Practice

Lecturer: Mr. R. Naudi, Prof A. Leone Ganado, Mr S Caruana, Mr T. Spiteri Staines
Prerequisites: CIS2030
Leads to: CIS3020 CIS3050
ECTS Credits: 6
Lectures: 40
Tutorials/Practical Labs: 4
Semester: 1st and 2nd

Estimation Techniques including the Function Point method and COCOMO. Project costing techniques. Basic Quality Management concepts.

Using MS project 2003 for project management


Project management of e-commerce development

**Method of Assessment:**
Test (85%) MS Project Assignment (15%)

**Textbooks:**
- K. Schwalbe *I.T. Project Management* Thomson Learning
- D Yeates, *Project Management for Information Systems* Pitman
- Duhig Berry Ltd. Series on *Project control, Project Initiation, Quality Management, People Management, Configuration Management* Blackwell

► **CIS2051 Physical Design and Implementation of Object Oriented Applications**

Lecturer: Dr. V. Nezval  
Prerequisites: CIS1071 or Java or C  
Leads to: CIS 3081, CIS3180  
Lectures: 28  
Tutorials/Practical Labs: 14  
Semester: 1st & 2nd  
Credits: 4

Physical design of applications using languages supporting both Procedural and Object Oriented design as well as development of applications using productive tools available.


Two languages, in particular Object Pascal and C#, will be used for implementation of the same Object Oriented Applications.


Available visual tools will be use for implementation of these Applications

**Method of Assessment:**
Test (80%) Practical Assignment (20%)

**Textbooks:**
- G Booch *Object Oriented Analysis and Design, 2nd edition* Benjamin Cummings 1993  
- C. Calvert *Delphi 4 Unleashed* SAMS 1998 0-672-31285-9  
CIS2090 Practical design and implementation with DBMS

Lecturer: Mr. J. Vella
Prerequisites: CIS2010 CIS1040
Leads to: CIS3010, CIS3030
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 7
Semester: 2nd

The objective of this unit is to allow students to demonstrate their acquired knowledge on logical design and database models to construct a sound physical implementation.

The course material covers some important pragmatic aspects of DBMS. These aspects include file structures; database engines; query processing optimisation and tuning; and connectivity (issues and standards).

Design methods, specifically Entity-Relationship Models and normalisation, are revisited and emphasised with intuitive integrity constraints on schemas designed for the relational data model. Database triggers are also introduced to maintain the operational characteristics of an application. Data security and DBMS mechanisms to support it discussed at some length.

The operational roles of a database administrator (for example data back up, data replication decisions, database design reviews, monitoring of access and resources) are enumerated.

Students are assigned a description of an application on which the practicals are done.

Method of Assessment:
Test (85%) Practicals (15%)

Textbooks
- K T Owens; Building Intelligent Databases with Oracle PL/SQL, PH Publ.
- Reference: Systems Manuals available on request

CIS2131 Network and Communication Issues in Information Systems

Lecturer: Mr. C. Meli, Mr. N. Cutajar
Prerequisites: CIS1021 or CIS1061
Leads to: CIS3081
ECTS Credits: 4
Lectures: 28
Pract. Labs/Tutorials: 4
Semester: 1st

This unit is a basic course in the technology and software which supports both intra and inter-organisational communication and on the importance of security and controls in both the development stage and the operational stage of software management.

Topics to be covered include:

- Adding value through communication. Business network configuration and network management with the importance of computer security
- The role of Groupware and workgroup computing; Electronic Mail; Information retrieval from news and database services; USENET Hierarchies and RFD/CFV processes
- USENET Control Message Formats and solutions to network abuse; Mail2News Gateway programming
Denial of Service and DDOS attacks; Threats of malicious software e.g. viruses, worms, spyware and their prevention/removal

The aim is to introduce students to information systems functioning over a network. The course starts with the basics of communication protocols and the OSI reference model. Logical design issues of each of the seven layers. Issues such as routing and connection management, data compression principles and techniques are studied. Operating system communication interfaces are presented (e.g. Unix and Windows sockets). Comparison and contrasts between the OSI and TCP/IP model. Also discusses the various Network topologies and importance of Storage Area Networks in today’s computing environments.

Method of Assessment:
Test

Textbooks:
- M. Hills *Intranet as Groupware* John Wiley and Sons ISBN 0471163732
- C.P. Fleeger *Security in Computing* Prentice-Hall.
- Lance Hoffman *Rogue Programs*
- FAQs on USENET at http://www.faqs.org

**CIS2120 Applied Graphic Design and user application standards**

Lecturer: Mr C. Meli
Prerequisites: CIS1071
Leads to: CIS3030
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 7
Semester: 2nd

This unit introduces the concepts of computer graphics with a practical orientation, towards ray-tracing and animation applications.

The course covers:
- Applications of computer graphics
- Professional web development through HTML 4.01 Transitional and XHTML W3C standards and validation, including graphics optimisation for the web, Meta-Tag definitions and Maltese language ISO web standard encodings
- Graphics file formats and “Safe” Palettes
- Vertex-Based Models
- Edge-based and Explicit-based Models with practical implementation
- Polygon-nets, Euler-Poincare’ Formulas
- Modelling, Scanners and Triangulation
- Sweeps, Binary Spaces and Solid Modelling
- Implicit Surfaces
- Using Raytracers to Create Scenes
- CSG including Clipping and UV-bounding of objects
- Particle Systems and Movement
- Animation Techniques, Lerpning (Linear Interpolation)
- Creation of a Ray-Traced Animation
- L-Grammars and L-Systems
- User interface guidelines, GUI standards, SDI and MDI implemented under Windows, Linux (KDE, Gnome)
- Chrome and Chrome Providers
- Platform-neutral Scripting applications using html interfaces
- Website and Application Usability
- Codecs and Containers eg. Matroska, OGM, AVI and Graphic and Data Visualization Tools
• Standardisation Forms
• VRML and CAD

**Method of Assessment:**
Test (85%) Practical Project (15%)

**Textbooks**
• Foley, Van Dam et al *Computer Graphics, Principles and Practice*
• Polyray, Raggier documentation
• CIS2120 Lecture Handouts
• [http://www.w3c.org](http://www.w3c.org)
• comp.lang.graphics and malta.comp USENET newsgroups
• comp.lang.graphics FAQ
• A List Apart website at [http://www.alistapart.com](http://www.alistapart.com)
• CIS2120 website at [http://www.cis.um.edu.mt/staff/cmeli/other/graphics](http://www.cis.um.edu.mt/staff/cmeli/other/graphics)

► **CIS2150 Applications for e-business using Java**

Lecturer: Dr. V. Nezval  
Prerequisites: CIS 1032 or CIS1071 or ANSI C programming course  
Leads to: CIS3081, CIS3180  
Lectures: 14  
Tutorials/Practical Labs: 7  
Semester: 2nd  
ECTS Credits: 2

The aim of this unit is to apply programming skills in Java language to relevant topics in e-Business oriented applications.

Contents (in brief):
Applets, advanced topics: Applet user interface components. Threads in applets, Communicating with other programs  
Servlets and JSP’s: Servlet containers and Web servers. Life cycle of servlets and JSP’s and their access from applications and applets, servlet and JSP connectivity to databases using ODBC

Hands on Visual tools used for creation of Java Component based programs, JBuilder as a representative tool, Writing Java program using JBuilder

For B.Sc. I.T. Students Only.

**Method of Assessment:** Test

**Textbooks :**
6.6 UNITS IN COMPUTER SCIENCE

CSA2010 - Compiling Techniques

Lecturer: Mr. Kristian Guillaumier  
Semester: 1st  
Prerequisites: None  
Leads to: None  
ECTS Credits: 4  
Tutorials / Practicals: 8 Hrs  
Lectures: 20 Hrs

This unit discusses the basic concepts of compilers and compiling techniques. It introduces the theory of formal languages and presents the concepts behind:

- Lexical Analysis: revisiting finite state machines, automata, and regular expressions to proceed to token specification and lexical analyser generators,
- Parsing and Syntax Analysis: revisiting context-free grammars to proceed to LL / LR parsing, top-down parsing, bottom-up parsing, operator precedence parsing, and parser generators,
- Code Generation: describing the notions of intermediate languages and register machines and proceeds to cover control flow, and symbol-table handling.

Method of Assessment: Test: (75%) Coursework: (25%)

Textbooks:

- Bennet. Introduction to Compiling Techniques.  
- Trembley, Sorensen. Theory and Practice of Compiler Writing.

CSA2050 - Introduction to Computational Linguistics

Lecturer: Mr. Michael Rosner  
Semester: 2nd  
Prerequisites: None  
Leads to: CSA3180  
ECTS Credits: 2  
Tutorials / Practicals: 4 Hrs  
Lectures: 14 Hrs

Computational Linguistics applies the concepts of Computer Science to the discipline of Linguistics, and is the basis for a wide range of technologies related to the processing of human language and speech. These include question answering; spoken dialogue systems; text mining, information extraction etc.

The main aim of the course is to provide an introduction to the fundamentals of the subject with a mix of theory and practice. The following areas will be looked at in particular:
Grammatical categories and automatic part of speech tagging.
Morphology (word structure) and the computational lexicon.
Grammar formalisms and computation.

No prior knowledge of Linguistics is presumed, the necessary background being provided as an integral part of the course. Similarly, no special knowledge of Computer Science is assumed, but experience with the use of computers and some appreciation of the role of programming would be desirable. A keen interest in language and applications of language technology is more or less mandatory.

**Method of Assessment:**

Test: (80%)
Coursework: (20%)

**Textbooks:**


**CSA2070 – Core Computer Science I**

Semester: 1<sup>st</sup>
ECTS Credits: 6
Prerequisites: CSA1010
 Leads to: CSA2100

This credit consists of two modules:

**Module 1: Operating Systems**

Lecturer: Dr. Kevin Vella
Leads to: CSA3150
Tutorials / Practicals: 4 Hrs
Lectures: 24 Hrs

The aim of this unit is to discuss the issues involved in designing a general purpose multiprogramming operating system, and to enable students to interact with the internals of a real operating system. The course covers topics such as the role of an operating system in providing a process abstraction, CPU scheduling, interprocess synchronisation and communication, memory management, virtual memory, file system facilities, I/O device handling, as well as adequate security and protection.

**Method of Assessment:**

Test: (66%)

**Textbooks:**

Module 2: Software Engineering

Lecturer: Dr. Ernest Cachia  
Leads to: CSA3170  
Tutorials / Practicals: None  
Lectures: 14 Hrs

This unit introduces the student to the attributes associated with a rigorous approach to the construction of reliable software systems. It will take the student from the basic heuristic as well as more formal principles outlining software engineering, through a systematic insight into software quality aspects and their indicative nature. This unit will also offer students insight into system representation forms other than those discussed in unit CSA1010, further analyse the principles behind the definition of system specifications which lie at the basis of correct system development. This unit will also briefly introduce the student to the Unified Modeling Language (UML) and its application in modern systems development. Software reusability techniques will also be discussed, compared and examples provided.

Method of Assessment: Test: (33% of which 80% is Exam and 20% is Coursework)

Textbooks:

-or-

Additional Textbooks

CSA2080 – Core Computer Science II

Semester: 2nd  
ECTS Credits: 6  
Prerequisites: CSA1060, CSA1080

Method of Assessment:  
Test: (85%)  
Coursework: (15%)

This credit consists of two modules:

Module 1: Formal Languages & Automata

Lecturer: Dr. Gordon Pace  
Leads to: CSA3100  
Tutorials / Practicals: None  
Lectures: 21 Hrs
This module deals with the formal treatment of languages and automata (or machines) to recognise languages. The aims are not only at instilling the basic notions of languages, grammars and automata using formal mathematical notation but also provides a practical perspective, by applying the mathematical results to design parsers.

Syllabus:

- Formal languages and grammars.
- Regular languages: regular grammars, finite-state automata, regular expressions.
- Context-free languages: context-free grammars, pushdown automata.
- Closure properties of regular and context-free languages.
- Normal forms for grammars.
- Recognition algorithms for grammars.

Textbooks:


Module 2: Techniques in Artificial Intelligence

Lecturers: Dr. John Abela, Dr. Matthew Montebello
Leads to: CSA3210, CSA3220
Tutorials / Practicals: None
Lectures: 21 Hrs

This module aims to familiarize students with the nature of AI problems and related practical solution techniques. The module is divided into two parts, each of which includes practical coursework.

Part I is concerned with fundamental issues and serves as an introduction to the subject, its goals, and the use of symbolic programming techniques. Topics include:

- Problem Solving
- Searching – Graphs
- Agenda/Means-End Analysis/Constraint Satisfaction
- Game Playing
- Knowledge Representation
- Expert Systems
- Planning – Blocks World

Part II is oriented towards the following topics:

- Search and Optimization problems
- Function Approximation problems (i.e. Function Learning and Synthesis)
- ANNs (theory, architecture, design, and implementation)
- Genetic and Evolutionary Algorithms (background, design, and implementation)
- Montecarlo Techniques applied to Search and Optimization problems.
- Artificial Life.
- WISARD neural network for scene analysis
- Feature extraction and the Hough Transform
The emphasis will be on the design and implementation of programs that solve various search, optimization, and function approximation problems using Neural Networks, Genetic Algorithms, and the Montecarlo random search technique.

Problems discussed include: OCR, machine vision, speech recognition, scheduling, pattern recognition, and some NP Hard problems

Textbooks (Part 1):
- Website: http://staff.um.edu.mt/mmon1/lectures/csa2120

Textbooks (Part 2):
- Course notes and handouts will also be provided by the lecturer for part II

CSA2090 – Systems Programming

Semester: 1st & 2nd
ECTS Credits: 4
Prerequisites: CSA1011
Leads to: CSA3030

This credit consists of two modules:

Module 1: Programming in C

Lecturer: Dr. Christopher Staff
Tutorials / Practicals: None
Lectures: 10 Hrs

This course assumes that students have an aptitude for computer programming in an imperative programming language, and that students are familiar with programming concepts such as variables, functions, flow control, and linked lists. The teaching environment is UNIX-based, and so students are expected to know either UNIX System V or Berkeley UNIX. By the end of the course, students will be proficient C programmers who will also appreciate the differences in how DOS and UNIX manage memory.

Following a rapid introduction to the C language syntax, which will take place in one or two two-hour sessions, the focus of the course will be on pointers, dynamic memory allocation, and data structures, as well as standard library routines for standard and file I/O and string handling. Program structuring using multiple files and header files will be encouraged.

Textbooks:
Module 2: Systems Programming

Lecturer: Mr. Joseph Cordina
Tutorials / Practicals: None
Lectures: 18 Hrs

This module reinforces the concepts introduced in Operating Systems module of Core Computer Science with a hands-on approach. The UNIX system and the C language will be used to experiment with operating system facilities in avenues such as low-level file input/output, process manipulation, advanced signal handling and interprocess communication (such as pipes, fifo files, semaphores, shared memory and message queues).

Method of Assessment: Test: (60%) Coursework: (40%)

Textbooks:


CSA2100 – Techniques in Operating Systems

Semester: 2nd
ECTS Credits: 4
Prerequisites: CSA2070
Leads to: CSA3170

Method of Assessment: Test: (65%) Coursework: (35%)

This credit consists of two modules:

Module 1: Sockets and TCP / IP

Lecturer: Mr. Joseph Cordina
Tutorials / Practicals: None
Lectures: 14 Hrs

In this module socket programming (TCP and Unix sockets) will be investigated. A generalised API will be used that can be applied to any programming language. In addition, TCP/IP concepts will be discussed and introduced, familiarising the student with the underlying techniques used to connect computers on the internet.

Textbooks:
Module 2: Windows Programming

Lecturers: Dr. John Abela / Mr. Kristian Guillaumier
Tutorials / Practicals: None
Lectures: 14 Hrs

Microsoft Windows is the world’s most popular operating system and it is easy to forget the sea of change that Windows brought to office and home desktop computing. The aim of this two-credit module is to introduce the student to the core concepts and technologies that are used in programs and applications that run under this very important operating system. Topics covered are:

- The history of Microsoft Windows
- Windows Architecture
- Windows, Events, Messages, and Callback functions
- The Win32 API
- Windows Multi-tasking and Multi-threading
- Building Windows Applications
- Building Dynamic Link Libraries (DLLs)
- Window Font Technology and GDI
- Winsock and TCP/IP programming
- ActiveX / COM
- The .NET framework (including CLR technology)

Prerequisites: Programming in a high-level language.

Main textbooks:

- Charles Petzold, Programming Windows with C# (Core Reference), Microsoft Press, ISBN: 0735613702.

Additional textbooks


CSA2130 – Object Oriented Modeling and Java

Semester: 1st
ECTS Credits: 4
Prerequisites: CSA1012
Leads to: CSA3190

This credit consists of two modules:

Module 1: Java Programming

Lecturer: Mr. Patrick Abela
Tutorials / Practicals: None

Lectures: 25 Hrs

This course focuses on the Java programming language and covers a number of standard API’s such as Swing, Input/Output, Java Beans, XML, serialization, threads and Remote Method Invocation (RMI). We shall also cover some techniques (such as profiling and the use of garbage collection interfaces), which are used to optimize Java’s virtual machine performance. Each lecture is followed up by a tutorial session.

Method of Assessment: Test: (60%) Coursework: (40%)

Textbooks:
- H Deitel, P Deitel. Java: How to Program. 2nd Ed. Prentice Hall.

Module 2: UML

Lecturer: Dr. Ernest Cachia
Tutorials / Practicals: None
Lectures: 3 Hrs

This part of the course will briefly introduce the student to the basic concepts and accompanying diagramming techniques inherent in the modern development environment of UML.

Main textbooks:

6.7 Units in Communication and Computer Systems Engineering

> CCE2001 Computer Logic and Organisation II

Lecturer: Dr J Briffa
Prerequisites: CCE1002
Leads to: CCE2002,
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 7
Semester: 1
- Sequential logic. The ideas of state governed by clocks, state tables, state minimisation techniques are included. These ideas are then applied to registers and counters.
- The control unit. Hardwired and Microprogram Control. Design of a computer control unit.
- The memory unit: Design of various types of memories using standard memory IC’s. Memory interleaving. Cache Memory.
Method of Assessment:
Test (90%) Assignment (10%)

Textbook:
  McGraw Hill ISE

► CCE2101 Signal Analysis

Lecturer: Dr V. Buttigieg
Prerequisites: MAT1803
Leads to: CCE3201
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 1

• Linear time invariant systems
• Discrete and continuous time systems
• Impulse response of systems
• The convolution sum and the convolution integral.
• Fourier Series Analysis
• Trigonometric and exponential forms of the Fourier series for periodic signals.
• Properties of the Fourier series.
• The RMS of a complex waveform.
• Fourier Transform Analysis
• Continuous time Fourier transform for non-periodic and periodic signals.
• Properties of the Fourier transform.
• Analyses of linear circuits using the Fourier transform
• Power and Energy Spectral Density functions
• Correlation
• Discrete-time Fourier Transform and the Discrete Fourier Transform
• Sampling
• The sampling theorem
• Aliasing
• Reconstruction

Method of Assessment:
Examination (90%) Coursework (10%)

Textbook:

► CCE2105 Electromagnetic Theory

Lecturer: Dr A Muscat
Credits: 4
Prerequisites: CCE 2101
Lecture/Tutorials: 28 Hours
Labs: 9 Hours
Leads to: CCE 3201
Semester: 2
Objectives:
The role of Transmission Line and Electromagnetic Theories in the design of electro-magnetic components. The theories are supported with engineering applications.

Syllabus:
- Laws of Electrostatics and Magnetostatics:
  Link to circuit theory:
  Design of bio-medical devices, pace-makers and de-fibrillators.
  Electrodynamics: Cathode-ray-tube, electro-mechanical actuators, electro-acoustic transducers, Liquid Crystal Displays.
  Numerical Solution of Static Fields.
- Transmission Line Theory:
  The Smith Chart:
  Design of High Speed Digital Systems.
  Design of High-Frequency Filters and Amplifiers.
  Quasi-Static approximation of printed circuits.
  Coupling and cross-talk in PCB design.
- Maxwell's Equations:
  Derivation of Circuit and Transmission-Line Theories from Electro-magnetic theory.
  Resonators for oscillators and filters.
  Transmission of waves in space.
  Design of Radar Systems.

Laboratory Work
- Computer simulation of electro-magnetic fields.
- Electro-magnetic field measurements.

Method of Assessment:
Coursework 10% Test 90%

Textbook:

CCE2002 Microprocessor Systems

Lecturer: Dr. C. J. Debono
Prerequisites: CCE2001
Leads to: CCE3001
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 7
Semester: 2

Syllabus:
- Microprocessor organisation: The internal set-up of a standard 16-bit microprocessor such as the 8086. The need of special registers.
- CPU cycles: Clock, machine and instruction cycles.
- Microprocessor Assembly Language: The assembly language of the 8086. The use of assembler and debugging tools. Design and implementation of assembly language programs.
- Types of interrupts and their use in sequencing events.
- The connection of a microprocessor such as the 8086 to special peripheral chips including a parallel peripheral interface, a serial peripheral interface, a priority interrupt controller, and interval counter and timer, a DMA IC.
- Programming of the peripheral chips
- Relation of a microprocessor system to a PC.

**Method of Assessment:**
Test (80%) Assignment (20%)

**Textbook:**
- Uffenbeck J. The 80X86 Family Prentice Hall ISBN 13-571241-6

▶ **CCE2112 Telecommunications I**

- Lecturer: Dr. A. Muscat
- Prerequisites:
- Leads to: CCE2122
- Credits: 4
- Lectures: 22
- Tutorials/Practical Labs: 6
- Semester: 1

**Analogue systems**
- Voice and video analogue signals
- Signal bandwidth
- Baseband transmission
- Amplitude and Frequency Modulation
- Simplex and Duplex systems
- Transceiver architecture

**Digital systems**
- Digital Voice and video analogue signals
- Digital Signal bandwidth
- Baseband transmission
- Amplitude and Phase Modulation
- Simplex and Duplex systems
- Transceiver architecture

**Multiplexing Methods**
- Frequency Division Multiplexing
- Time Division Multiplexing

**Information Theory**
- Self Information and Entropy
- Sources with and without Memory
- Source Coding
- Channel Capacity
- Channel Coding

**Method of Assessment:**
Test (100%) Assignment (0%)
CCE2122 Telecommunications II

Lecturer: Dr A Muscat
Prerequisites: CCE 2112
Leads to:
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 2

Sampling and Quantisation
Impulse, flat-topped and natural sampling
Sampling in TD and FD
Quantisation Noise
Companding

Pulse Systems
PAM in TD and FD
PAM Transceiver
PPM and PWM systems
PPM and PWM transceivers

Performance of Systems in Noise
AWGN representation and modelling
Narrow-Band representation of noise
DSB-SC performance in AWGN
NRZ code performance in AWGN
ASK and PSK performance in AWGN
Energy per Bit and BER plots
Inter-symbol Interference
Pulse systems in Noise

Method of Assessment:
Test (80%) Assignment (20%)

CCE3012 Introduction to Computer Networks [CCE3002]

Lecturer: Prof. Paul Micallef
Prerequisites: ELE1601, ELE1602
Leads to: CCE3022
Credits: 2
Lectures: 14
Tutorials/Practical Labs: 7
Semester: 2

- Type of systems: terminal, mainframe, client-server.
- Computer Network Topology: bus, ring, tree, star, distributed.
- Computer Communication Techniques: polling, multi-access, bandwidth considerations.
- Multiaccess local networks - low speed - CSMA/CD; Token Ring High speed - FDDI, Fast Ethernet
- Bridges in LAN’s.

Method of Assessment:
Examination 90% Coursework 10%

Textbooks:
- Halsall F. *Data Communications, Computer Networks and Open Systems* (Fourth Edition)
CCE 2301 – INTRODUCTION TO MATLAB

Lecturer: Dr J Briffa
Prerequisite: CCE1001
Credits: 4
Lectures: 2 hrs/wk
Semester: 2

Objectives
MATLAB has become a standard package to solve engineering mathematical problems, for signal processing, visualization of data and system modeling. The objectives of this unit are to give a good foundation in using MATLAB and to develop the necessary skills to effectively use this tool to solve engineering problems.

Syllabus
MATLAB Basics
The MATLAB Desktop;
Using MATLAB as a calculator;
Performing vector/matrix calculations;
Arrays and Cells;
Using the MATLAB help;
Creating and using script files.
Basic Graphics
2-D and 3-D plotting;
Interpolation and curve fitting of experimental data;
MATLAB Programming
Flow control and array operations;
Exception handling;
Object-oriented programming;
Debugging and Profiling;
Optimising MATLAB code.
File I/O
Data import and export;
Engineering Maths with MATLAB
MATRIX Algebra;
Data analysis and interpolation;
Polynomials;
Fourier Analysis;
Optimisation;
Integration and differentiation;
Differential equations.
Graphical User Interfaces
Developing GUI applications;
Callback programming.
Interfacing MATLAB with C.

Laboratory Work
Examples of topics covered in theory.

Assessment
  Coursework - 20%
  Exam – 80% (Practical)

Textbook

6.8 ADDITIONAL UNITS AS ADVISED BY THE BOARD

►MAT1401 Discrete Methods

Lecturer: Prof. J. Lauri
Prerequisites: A-Level
Leads to: MAT2402
Credits: 4
Lectures: 20
Tutorials: 8
Semester: 2

• Basic counting methods:
• Permutations,
• Combinations,
• Partitions of a set,
• The exclusion principle,
• Recurrence relations:
• Generating functions,
• Partitions of a positive integer.

Method of Assessment: 100% Exam

Textbook:
or
MAT2402 Networks

Lecturer: Prof. J. Lauri
Prerequisites: MAT1401
Credits: 2
Lectures: 10
Tutorials: 4
Semester: 1

• Basic graph theoretical concepts;
• Shorexam paths and minimum spanning tree;
• Flow in networks, max-flow-min-cut theorem;
• Matching in bipartite graphs;
• Scheduling and sequencing problems;
• Travelling salesman problem.

Method of Assessment: Exam (100%)

Textbook:

PHY2150 Numerical Analysis

Lecturer: Prof. A. Buhagiar
Credits: 2
Lectures: 14
Tutorials/Practical Labs: 5
Semester: 2

Roots of Equations: Bisection method, Newton’s method, fixed point iterations, convergence rates.


Interpolation: Finite difference operators, interpolating formulas, Lagrange Polynomials, cubic splines.

Ordinary Differential Equations: Euler method, modified Euler method, Runge Kutta, finite differences for Boundary value problems.

Method of Assessment:
Exam (80%) Assignment (20%)

Textbooks:
• Gerald Curtis F., Wheatley P.O. Applied Numerical Analysis, Addison Wesley
• Isaacson E. I, Keller H., Analysis of Numerical Methods, Wiley

MGT2021 Business Management

Lecturer: Mr N. Massa, Department of Management
Credits: 4
Lectures: 28
Tutorials/Practical Labs: None
Semester: 1
Rationale & Learning Objectives:

Building on MGT1011 Introduction to Management, this course provides an active and applied view of managerial competencies and the execution of key management functions engaged towards achieving organisational goals. Emphasis is made on the application of course material to current management issues and students are encouraged to interact and participate.

Indicative Syllabus & Course Outline

The course introduces students to the complex ethical and social responsibilities facing organisations, managers and employees. The fundamental managerial function of planning and strategy formulation is covered, followed by the components of leadership and motivation, important in any organisational context. Main aspects of the other managerial functions are also treated as are other aspects of business administration.

Method of Assessment:
Exam or assignment

Recommended Text
- Griffon Management (Last edition); Selected Bibliography
- Mintzber The Nature of Managerial Work Harper and Row 1973
- Drucker Managing for results Harper and Row 1964
- Peters and Waterman In Search of Excellence

► SOR1211 Probability

Lecturer: Mr L. Camilleri.
Prerequisites: Pure or Applied Mathematics at the level requested for entry to engineering and technology courses
Credits: 2
Lectures: 1 hr per wk for 14 wks including tutorials and collective computer lab sessions
Semester: 1

- Elementary Probability
- Combinatorial Probability
- Probability Spaces
- Conditional Probability, Independence, Bayes' Theorem
- Random Variables
- Distributions: discrete and continuous
- Uniform, Binomial, Poisson, Geometric Distributions
- Uniform continuous, Exponential and Normal Distributions

- Expectations and Variances
- Definition of Expectation
- Expectations of Standard Discrete and Continuous Distributions
- Definition of Variance
- Variances of Standard Discrete and Continuous Distributions

Some time will be dedicated to the instruction and use of statistical modules in some widely used package like a commercial spreadsheet package. The extent and amount of material and time dedicated to this will naturally depend on the availability of human and computing resources.

Method of Assessment: Exam

Recommended Texts:
- Freund J.E., Miller I. Probability and Statistics for Engineers
YEAR THREE

The following credits will be offered in years 3 & 4:

6.9 UNITS IN INFORMATICS

► CIS3021 I.S. Strategy Management and Practice

Lecturer: Prof. A. Leone Ganado, Mr. T. Spiteri Staines, Mr. R. Naudi, Mr. S. Caruana
Prerequisites: CIS2131, CIS2041
Leads to:
ECTS Credits: 4
Lectures: 28
Semester: 1st

- Introduction to Strategic management and IT including IT strategies for the web.
- Analytical tools for strategic management. Strategic choices.
- Information System planning: Formulation and Toolkit.
- Information Resource management.
- Requirements Engineering
- Management issues in IT
- Problem Management
- Help Desk Models
- Software Control and Distribution
- Service Level Management
- Organisational Structures and Strategies for managing IS resources – outsourcing
- Configuration Human Resource Planning in IS.
- IT performance & capacity planning, measurement and evaluation

Method of Assessment:
Test (85%) Assignment (15%)

Textbooks:
- AEleen Frisch Essential System Administration O'Reilly & Associates, Inc.
- A. Rae, Software evaluation for certification ISBN 007709042X Addison-Wesley
- N. Bruton, Effective User Support - How to manage a Help desk ISBN 0077079531 Addison-Wesley

► CIS3031 Scientific and Quantitative Aspects of I.S.

Lecturer: Prof. A. Leone Ganado, Mr. J. Galea
Prerequisites: Some knowledge of mathematics and probability is expected
Leads to:
ECTS Credits: 4
Lectures: 28
Semester: 1st

Scientific applications pose different problems from those of traditional business-oriented applications. Although issues such as flexibility, scalability, and efficient query processing, are raised in general conventional information systems, the focus in scientific information systems is clearly different. Scientific information systems involve analysing public and proprietary data across diverse sources. One of the
main challenges of such analyses is coping with large amounts of data scattered across poorly correlated data sources, and the absence of consistent methods for representing, storing, querying, and interpreting data, thus interoperability is dealt with in further detail. High performance computing and visualisation capabilities also essential features of scientific environments. This unit emphasise the need for collaboration between data engineers and domain specialists.

The components and procedures of information systems, namely data capture, data storage, database systems, multidimensional access methods, data mining, object-oriented techniques, data analysis and decision support, monitoring, simulation models, dissemination of information, management information systems, metadata management, and visualisation are revisited in the context of a scientific information systems.

A case study will be derived from scientific applications such as bioinformatics, laboratory information management systems, geographical and / or environmental information systems. High-performance computing and the absolute necessity to couple disparate systems are common features of scientific information systems will be discussed. Techniques for storing and accessing scientific data will be discussed.

Concept of building models to support decision taking. Mathematical, probabilistic and heuristic models to support decisions.

Applying programming techniques and algorithms to map such models into computer based algorithms. Computer architectures to solve decision support problems.

Decision support models, network models, game theory models and dynamic programming problems.

Iteration and approximation techniques. Alpha and Beta pruning. Simulation techniques.

Method of Assessment:
Test (85%) Assignment (15%)

Textbooks:
- V Lofti C C Pegels Decision Support Systems for Management Science IRWIN
- Jean-Louis Lauviere' Problem Solving and Artificial Intelligence, Prentice – Hall
- Eugene Isaacson, Herbert Bishop Keller, Analysis of Numerical Methods John Wiley and Sons
- Textbooks / journals on specific domain: bioinformatics, GIS, Environmental Information Systems as suggested
- J.P. Bigus, Data Mining with Neural Networks, 1996 McGraw Hill.
- Research papers from journals

► CIS3051 Information Systems Engineering I

Lecturer: Mr. R. Naudi, Prof. A. Leone Ganado, Mr. T. Spiteri Staines
Prerequisites: CIS2030
Leads to: CIS4050
ECTS Credits: 6
Lectures: 42
Semester: 1st


Information and meaning. Logical and physical systems. Concept mapping.
The software development process including Web development, Underlying processes, methods, techniques and tools. Methodology framework, data modelling, Process Modelling, Behavioural modelling.


Issues of quality and risk analysis when developing and deploying systems

**Method of Assessment:**
Test (70%) Assignment (30%)

**Textbooks:**
- Ghezzi, Jayazeri, Mandrioli *Fundamentals of Software Engineering* Prentice-Hall
- David Card, Robert Glass *Measuring Software Design Quality* Prentice Hall
- *Software Engineering Journal (SEJ)* IEEE & BCS
- Various papers and articles

**CIS3081 Advanced Client/Server Computing and Distributed Databases**

Lecturer: Dr. V. Nezval, Mr. Joseph Vella, Mr. C. Meli
Prerequisites: CIS 2051, CIS2090
Leads to: CIS 3180, CIS3110
ECTS Credits: 4
Lectures: 28 Tutorials/Practical Labs: 14
Semester: 1st

The overall aim to this unit is to reinforce and introduce to students principles, techniques and methodologies for the deployment and movement of the computational artifacts (eg. data and programs) across state of art computer infrastructures. This unit should provide an essential background for further Client/Server model applications which deploy objects and use an object oriented approach for interoperability among different platforms.

One accent of this unit is given to deep understanding of Client/Server model mainly regarding connection and communication issues between the Client and Server. Three tier architecture is also applied namely to the Server (middle tier) handling of a database (third tier).


Another accent of this unit is the study of distributed data and functions, across computer network. The theory, design, specification, implementation, and performance of distributed database systems are pursued. Also the inherently different set-ups for multidatabase are emphasized.

Topics include principles of naming and location, atomicity, query processing and optimisation, resource sharing, concurrency control and other synchronization, recovery procedures, deadlock detection and avoidance, security, distributed access and control, distributed systems design, consistency and fault tolerance.
Method of Assessment:
Test

Textbooks:
• M.T. Ozu, P.V. Valduriez, *Principles of Distributed Databases*, PH Publishers

► CIS3090 Social and Professional Issues for Computing

Lecturer: Prof. A. Leone Ganado, Dr C. Staff et al
Prerequisites: For Students who intend following an IT career only.
ECTS Credits: 2
Lectures: 14
Semester: 1st

What is Ethics? What is Computer Ethics? Why have Ethics? Difference between ethics, morals, and laws.

Ethical frameworks: Just because something appears to be “for the common good” and it can be done, does it make it right? (e.g. adaptive marketing promises to bring you adverts for products that “you want” – but how have your preferences / interests / needs been discovered?) Does the need justify the means? Ethical Relativism, Utilitarianism, Deontological Theory.


Method of Assessment:
Test

Textbooks:

6.10 UNITS IN COMPUTER SCIENCE

CSA3030 – Introduction to Computer Graphics

Lecturer: Mr. Sandro Spina
Semester: 1st
Prerequisites: CSA2090
ECTS Credits: 2
Tutorials / Practicals: 5 Hrs
Lectures: 10 Hrs

This unit serves as an introduction to the field of Computer Graphics. This course will focus particularly on digital image synthesis, which entails making use of computers to generate images. Digital images synthesis involves the design of models for representing, rendering, viewing and storing images.
Topics to be covered in the course include raster and video displays, colour models, affine transformations, the 2D and 3D viewing pipeline, 3D viewing systems, three dimensional object representations, clipping, visible surface detection, local illumination models, shading, texture mapping and a gentle introduction to global illumination using ray tracing. Furthermore, a graphics API will be introduced as a method of expressing the algorithms outlined throughout the course.

Method of Assessment: Test: (66%) Coursework: (34%)

Textbooks:

CSA3100 – Computability and Complexity

Lecturer: Dr. Gordon Pace
Semester: 1st
Prerequisites: CSA2080
ECTS Credits: 2
Tutorials / Practicals: 2 Hrs
Lectures: 12 Hrs

This course starts with an appraisal of Chomsky's language classes, aimed primarily at showing how different computation models (finite state automata, pushdown automata) have limits on what they can compute. This insight is extended to Turing Machines, leading to an overview of Church's Thesis, which states that all reasonable computation models are equivalent. The limits of computation are discussed, leading to the Halting Problem, and other non-computable functions. But not all computable functions are tractable. The course finishes with an overview of NP-completeness and an appreciation of the open problem of whether P=NP.

Syllabus:
- Chomsky's language hierarchy
- Pumping lemmata to prove limits of each of Chomsky's language classes.
- Turing Machines and the limits of computing.
- Intractability, and NP-completeness

Method of Assessment: Test: (100%)

Textbooks:

CSA3150 - Concurrent and Distributed Systems

Lecturer: Dr. Kevin Vella
This first part of this course aims to introduce parallel computing as a host of emerging technologies which offers a departure for mainstream computing from the physical limits imposed by traditional Von Neumann computing. An alternative view of parallel computing as an enabling technology for supercomputing is also explored. Recent developments in multiprocessor systems (architecture and system software), parallel models of computation, programming parallel systems, and parallel algorithms will be briefly discussed. Students will use a multithreaded programming environment to implement a selection of concurrent algorithms. The skills that the student is expected to acquire through this course are relevant in the current computing scene, where multithreaded programming is emerging in the mainstream and application developers are expected to take on threads and concurrency in their stride.

The second part of this course investigates distributed systems from a real-world perspective by taking on a large case study in the form of the Internet. Fundamental Internet technologies such as the TCP/IP protocol stack, network addressing, the IP routing mechanism, subnetting, ICMP, the TCP protocol including sliding windows and congestion avoidance, DNS, dynamic routing protocols such as RIP and OSPF, security (firewalls, packet filters and proxies) and various application level protocols (FTP, Telnet, HTTP, STMP, etc.), will be dissected. The design rationale and implementation of these protocols and systems will be studied in depth.

Other topics in the general area of system software, concurrency and distribution will also be covered.

**Method of Assessment:**
- Test: (85%)  
- Coursework: (15%)

**Textbooks:**
- P.H. Welch. Parallel Computing Course Notes. Parallel Computing Research Group, University of Kent, UK (To Be Supplied During Lectures)

**CSA3160 - Scientific Computing**

Lecturers: Mr. Joseph Cordina, Mr. Sandro Spina  
Semester: 1st  
Prerequisites: PHY2380, SOR0211  
ECTS Credits: 4  
Tutorials / Practicals: None  
Lectures: 28 Hrs

The aim of this course is to introduce some mathematical methods and computational techniques for the simulation of real-life situations.

The first part of this course will cover the theoretical and mathematical foundations applicable to scientific computing. Most of the emphasis will be placed on Queuing theory, Markov models and an introduction to Chaos theory.
The second part of the course will cover problems with implementing the above models on computers. The limitations of traditional computer architectures will be discussed in terms of floating-point arithmetic and random number generation and their relationships to errors, stability and convergence. This course will be taught using several sample problems showing how the techniques introduced can be applied to real-life scenarios.

**Method of Assessment:** Test: (100%)

**Textbooks:**

TBA

**CSA3170 - Software Engineering**

Lecturers: Dr. Ernest Cachia, Mr. Mark Micallef  
Semester: 1st  
Prerequisites: CSA2070, CSA2100  
ECTS Credits: 6  
Tutorials / Practicals: None  
Lectures: 42 Hrs

The aim of this unit is to apply previously gained knowledge, with some additions introduced in this unit, to illustrate various techniques used in the creation of specific and effective software development environments – both technical and human. This unit will acquaint the student with basic software engineering approaches towards effective method, notation, tool, and team development – all the ingredients necessary for modern team-oriented software development. This unit will also expose students to traditional problematic issues (and possible solutions) encountered in software development. Fundamental concepts, such as data attributes/integrity and notation properties, lying at the heart of certain individual diagramming techniques as well as inter-tool communication, will be discussed. Another aim of this unit is to introduce students to the more specialised topics of software engineering, which include reliability, availability, fault/testing metrics, and real-time and concurrent system specification and scheduling theory and to introduce, and further explain to, the candidate the intricacies of software metric extraction and application (metrication). Topics such as Function and Object Point Analysis, Reliability and Availability functions, Stochastic state analysis, system testing criteria calculation, and similar other topics will be discussed. Depending on time and progress, hands-on experience of a cutting-edge commercial RAD development tool will be supplied. Some very basics aspects of model checking will also be discussed. Whenever possible, theoretical material will be supplemented with practical examples.

**Method of Assessment:** Test: (70%)  
Coursework: (30%)

**Textbooks:**

  ISBN: 0-077-09677-0


**Additional Textbooks:**
CSA3180 – Natural Language Processing

Lecturer: Mr. Michael Rosner
Semester: 1st
Prerequisites: CSA2050
ECTS Credits: 6
Tutorials / Practicals: 12 Hrs
Lectures: 42 Hrs

This course is about Natural Language Processing (NLP), a cover term which deals with different applications of Computational Linguistics to concrete problem areas that involve, in the first instance, the processing of text. The aim of the course is to provide some insight into the different techniques that are used in these areas. The course is divided into two parts.

Part one supplies the algorithmic basis for many of the background concepts introduced in CSA2050 (Introduction to Computational Linguistics), so some familiarity with these concepts will be presumed. This is not, however, a formal prerequisite. Key algorithms for handling words, sentences and texts will be presented including:

- **Words**
  - Tokenisation
  - Stemming
  - Morphological Processing

- **Sentence Analysis**
  - Parsing
  - Generation

- **Texts**
  - Chunking
  - Tagging

The second part of the course will offer an in-depth examination of the techniques used in a selection of application areas. These will be chosen from amongst the following:

- Computational Semantics
- Computational Morphology
- Discourse and Dialogue
- Information Extraction.
- Statistical NLP.
- Machine translation.

Coursework will involve the development of an NLP programs in one or more of the above areas.
Method of Assessment: Test: (75%) Coursework: (25%)

Textbooks:
- Website: http://www.cs.um.edu.mt/~mros/csa3180

CSA3190 – Internet Technologies and Java Server – Side Programming

Lecturers: Mr. Kristian Guillaumier, Dr. John Abela, Mr. Patrick Abela
Semester: 1st
Prerequisites: CSA2130
ECTS Credits: 4
Tutorials / Practicals: 8 Hrs
Lectures: 24 Hrs

The Internet and the World Wide Web have revolutionised software development with multi-media intensive, platform independent, code for conventional Internet-, Intranet-, and Extranet-based applications. The aim of this course is to introduce the student to the core technologies and concepts in use in this very exciting and important area. Topics include TCP/IP, HTTP, SSL, Web Server Technology, Browser Technology, Client-side scripting Javascript and VBScript, Multi-tiered applications, Databases and the Web, Credit-card transaction processing, and Security issues - symmetric vs Public Key encryption.

During the Java part of the course it is mostly focused on how the J2EE platform, addresses the needs of robust and scalable web systems. J2EE is the natural evolution of J2SE into a fully-fledged web-development platform. Various Java technologies which are used in four-tiered web applications in particular JDBC, JSP, Servlets and EJB's will be considered. In particular how the use of EJB's provides the necessary tools to separate the business logic from the presentation logic as well as implement transactional systems will be focused upon. Also the security mechanisms implemented in the Java platform will be revisited.

Method of Assessment: Test: (65%) Coursework: (15% - KG, 20% - PA)

Textbooks:
- Deitel, Deitel, and Nieto: Internet and World Wide Web: How to Program, Prentice Hall.
- Richard Monson-Haefel Enterprise JavaBeans

CSA3200 – Adaptive Hypertext Systems

Lecturer: Dr. Christopher Staff
ECTS Credits: 6
Semester: 1st
Prerequisites: None
Tutorials / Practicals: None
Lectures: 42 Hrs
The next-generation World Wide Web, the Semantic Web, will contain representations of meaning so that software agents can find meaningful information on the Web and act upon it in accordance with their users’ wishes. The generation beyond that, the Adaptive Web, will learn from its users to organise itself efficiently and effectively, and to provide its users with personalised and individualised spaces for searching, browsing, and using information. Adaptive Hypertext Systems must be able to discover, represent, and manipulate user interests and requirements as users navigate and search through a hyperspace, and then adapt the organisation of and the presentation of information accordingly. This study-unit explores essential components of Adaptive Hypertext Systems: user-adaptive systems, hypertext system, user modelling, knowledge representation, information retrieval, intelligent tutoring systems, and adaptation techniques.

Method of Assessment: Test: (100%)

Main textbooks (recommended):

CSA3210 - Agent Technology

Lecturers: Dr. Matthew Montebello, Mr. Charlie Abela
Semester: 1st
Prerequisites: CSA2080
ECTS Credits: 6
Tutorials / Practicals: None
Lectures: 42 Hrs

The first part of this course gives an overview of the state of the art in agent research and technologies with reference to applications in a variety of domains including: Internet-based information systems, adaptive (customizable) software systems, autonomous mobile and immobile robots, data mining and knowledge discovery, smart systems (smart homes, smart automobiles, etc.), decision support systems, and intelligent design and manufacturing systems. The second part will concentrate on employment of such software agents to practical and intelligent applications. It will build on issues covered in the first part with particular interest in areas of agent application like electronic commerce, recommendation systems, auctions, information retrieval over the WWW, and other commercial and cutting-edge scenarios. Some of the topics covered are: basics (history, subject matter), software architecture, properties and models of agents, agent inter connectors and agent systems, aspect models, mobility, co-ordination and security, architecture types for agent-based application systems, commercial agent application, standardization efforts, web services, ontologies, mark-up languages, semantic web and future directions.

Method of Assessment: Test: (75%) Coursework: (25%)

Textbooks:
Expert Systems (5 hrs)

Expert Systems are considered by many to be the most important contribution of A.I. to the wider world of computing. Hundreds of expert systems have been successfully implemented worldwide. This part of the course introduces the students to the history, principles, design, and implementation of modern production-rule expert systems.

Fuzzy Logic (9 hrs)

Since it was invented in the 70s to the present day, fuzzy logic has slowly gained in popularity. Fuzzy logic arose from the need for a mathematical formalism to characterize the concept of uncertainty (or fuzziness). Today we find fuzzy logic controllers in automobile transmissions, home appliances, cameras, VCRs, industrial machinery, trains, and many other devices. We introduce the student to the basic principles of fuzzy logic and fuzzy sets and then proceed to study the design and implementation of fuzzy logic controllers as used in many devices. The course Coursework will involve the implementation of a Fuzzy Logic Controller.

PART TWO – Machine Learning (28 hrs)

Both pattern recognition and machine learning belong to the most advanced areas of A.I. Numerical methods combined with A.I. techniques have been especially successful in pattern recognition. Research in Machine Learning is now recognised as one of the most important areas of A.I. as well as having application to knowledge acquisition in A.I. systems and contributing to the understanding of human cognition.

Part (I): Machine Learning

Topics discussed:

- Principles of learning machines, Gold’s Theorem
- Concepts and Categories in Cognitive Science
- Computational learning theory (COLT)
- PAC-learning
- Grammatical inference
- Concept learning
- Find-S, Candidate Elimination, and the ID-3 learning algorithms
Part (II): Pattern Recognition

Topics discussed:

- Clustering techniques
- Linear discriminant analysis
- Pattern Feature Extraction
- Pattern Understanding
- Advanced Neural Networks (Hopfield, Kohonen networks)
- Support Vector Machines

Method of Assessment: Test: (80%) Coursework: (20%)

Textbooks:

- Course notes and WWW links

6.11 UNITS IN COMMUNICATION AND COMPUTER SYSTEMS ENGINEERING

CCE 3021 – Real-Time Embedded Systems

Lecturer: Dr C J Debono
Credits: 4
Lectures/Tut.: 28 hrs
Labs: 14 hrs
Prerequisite: CCE 2002
Leads to: APTs and Projects
Semester: 1

Syllabus

Micro-Controller: The 8051 family of microcontrollers with particular emphasis on the 8032 using the Flite32 development board. Programming techniques in assembly language (A51) and C using the Keil system development software package.

Real-Time Operating Systems: An overview of different operating systems for the 8051, the Simple Embedded Operating System, the Round-Robin OS and the in-house developed PaulOS RTOS. Reference to other commercial RTOS products.

Embedded Systems: Interfacing, proto-typing, EPROM programming, finally developing a self-contained embedded system as part of the laboratory sessions and course assignment.

Assessment
Coursework - 20%
Exam - 80%

Textbooks:
Notes provided by lecturer
Reference books:


►CCE3022 Network Modelling [CCE3002]

Lecturer: Prof. Paul Micallef  
Prerequisites: CCE3012  
Leads to: CCE4004  
Credits: 2  
Lectures: 14  
Tutorials/Practical Labs: 7  
Semester: 2

- Traffic Modeling for Telecommunications Networks.  
- Queuing Theory: Poisson and other distributions, various types of queues including M/M/1 and M/M/n.  
- TDM and Statistical Multiplexing.  
- Simple error recovery: ARQ protocols; throughput and efficiency.

Method of Assessment:  
Examination (90%) Coursework (10%)

Textbooks:  
- Papers from IEEE Magazines and Notes will be provided

►CCE3201 Filters & Digital Signal Processing

Lecturer: Dr V. Buttigieg
Prerequisites: CCE2101
Leads to: CCE4201
Credits: 4
Lectures/Tut: 28
Labs: 9 hrs
Semester: 1

- The z-transform  
- Properties of the z-transform  
- Analysis of linear time-invariant systems in the z-domain  
- Causality and stability  
- Discrete Fourier Transform  
- Frequency domain sampling  
- Properties of the DFT  
- Circular convolution  
- Fast Fourier Transform  
- DFT for long data sequences  
- Analogue Filters  
- Modern Filter Theory: Characteristics of lowpass, highpass, bandpass, bandstop and comb filters.  
- Filter Section Approximations: Butterworth, Chebyshev and Bessel Filters  
- Frequency Transformations  
- Circuit Synthesis: Poles, zeros and impulse response.  
- Use of filter design software and filter tables.  
- Design of digital filters
Various Designs for linear phase FIR filters
- Design of IIR filters from analogue filters
- Direct design of digital IIR filters
- Decimation and Interpolation
- Realisation of discrete time systems
- Quantisation effects in digital signal processing.

Laboratory Work

Method of Assessment
Examination 10% Coursework 90%

Text Books

**ELE 3103: Hardware Description Language (HDL)**

Lecturer: Dr. I. Grech
Credits: 2
Lectures/Tut: 8 hrs
Labs: 6 hrs
Prerequisite: none
Leads to: ELE 3102, CCE 3202
Semester: 1

**Objectives**
The module introduces hardware description languages (mainly VHDL) as a tool for describing and synthesizing logic circuits in the process of digital design automation. During the lab sessions, the students will familiarise themselves with the process of digital integrated circuit synthesis, together with place and route, starting from HDL code to silicon level.

**Detailed Syllabus**
Introduction to the design methodology using hardware description languages.

Structural, concurrent and sequential VHDL descriptions, subprograms, VHDL operators, signals and variables, variable types, WAIT statements and sensitivity lists.

Simulation of VHDL models, exam benches, debugging – outputting to the console.

Design organisation and parameterisation: subprograms, packages, default, fixed and generic parameters, design configuration, general purpose exam benches.

Delay modeling and timing issues.

Dataflow description in VHDL: multiplexers, Moore and Mealy FSMs, open collector gates.

Brief introduction to Verilog HDL, comparison with VHDL.

**Laboratory Work**
Design of a digital circuit starting from a HDL description using Xilinx XC4000X FPGA chips. Also familiarisation with Cadence simulation, synthesis and place and route tools for digital chip design.

**Assessment**
Examination 75% Coursework 25%
Text Book
Zainalabedin Na

Reading List

►CCE3102: Transmission Media for Communications Theory

Lecturer: Mr P. Debono; Dr A Muscat
Credits: 4
Lectures/Tut: 28 hrs
Labs: 10 hrs
Prerequisite: CCE2105
Leads to: CCE4101
Semester: 2

Detailed Syllabus
- Transmission of Electromagnetic Waves in Space
- Transmitting and Receiving Antennas
- Design of wire, waveguide and microstrip antennas
- Design of reflector and multi-element antenna array systems
- Antennas for mobile terminals

Guided Electromagnetic Waves
- Characterisation of:
  - Copper Transmission lines
  - High Frequency Waveguides
  - Optical Waveguides

Propagation of Electromagnetic Waves in Space
- Characterisation of the mobile radio environment
- Radio direction finding and GPS
- Communications Satellites
- Weather Satellites

Method of Assessment:
Examination 80% Coursework 20%

Text books:
• Connor F. R. Antennas Publishers E. Arnold

►CCE3202: Advanced Digital Systems Design

Lecturer: Dr. Carl J. Debono/Dr. Victor Buttigieg
Credits: 4
Lectures/Tut: 14 hrs
Labs: 14 hrs
Prerequisite: CCE 3103, CCE 3201
Leads to: CCE 4201, CCE 4003
Semester: 2
Objectives
Two important components in modern digital system design are the Digital Signal Processor and the Filed Programmable Gate Array. These are also important design tools in rapid prototyping. The objective of this unit is to give a solid foundation how to design systems using these devices through a hands-on approach.

Detailed Syllabus
- DSPs in digital system design
- Architectural differences between Microprocessor, Microcontrollers and Digital Signal Processors
- The TMS320C3X and TMS320C6X Architectures
- Fixed and floating point number representation and its use in DSP
- Real-time issues when using DSPs – Interrupts, DMA, Pipelining
- Software Optimisation Techniques – Assembly Optimisation, Software Pipelining, C and Assembly Language Interactions
- Software Development Tools
- Programmable Logic Devices in Digital System Design
- Programming Logic Devices
- Field Programmable Gate Arrays
- The Xilinx XC4000X series FPGA chips
- Design and Exam Methodology using VHDL
- Rapid Prototyping
- Design Examples

Laboratory Work
Design of signal processing systems using DSPs and FPGAs.

Method of Assessment:
Examination  60%   Coursework  40%

Text books:
Stefan Sjoholm and Lennart Lindh, VHDL for Designers, ISBN: 0134734149
CCE 3203 - Digital Image Processing I

Lecturer: Dr K P Camilleri

Credits: 4
Lectures/Tut.: 28 hrs
Labs: 9 hrs
Prerequisite: CCE 2101, CCE 3201
Leads to: None

Semester: 2

Detailed Syllabus

- Preliminaries
  Images, pixels, resolution, operators, point spread function.
- Image Transforms
  Unitary Transforms, Singular value decomposition, 2D-DFT, Karhunen-Loeve Transform.
- Image Enhancement
  Histogram equalisation, principal component analysis, image noise, image filtering.
- Image Segmentation
  Edge detection, histogram-based thresholding, minimum error thresholding, region-based segmentation.

Assessment
Coursework - 20%
Exam - 80%

Textbooks

- Lecturer's Notes

YEAR IV

PROPOSED UNITS UNDER NEW HARMONIZED REGULATIONS

CSA4060 – Formal Methods

Lecturer: Dr. Gordon Pace
Semester: 1st
Prerequisites: CSA3100
ECTS Credits: 4
Tutorials / Practicals: 4 Hrs
Lectures: 24 Hrs

When designing safety-critical systems, it is not sufficient to test the software / hardware written, but it may be necessary to mathematically verify that the system works correctly. This course explores different topics in this research area, including mathematical tools (such as logics, and property languages), formal modeling of computer languages and systems, and techniques used for automatic model checking of such systems.
Syllabus:

- Temporal Logic, process calculi.
- Synchronous programming, circuit semantics.
- Model Checking of software and hardware.

Method of Assessment: Coursework/Presentation: (100%)

Students will each be presenting a different key-paper in the area of formal methods and will have to hand in a detailed review of their paper. They will also have to present a report on how 5 other papers presented by other students relate to their paper.

Textbooks:

- The course will be based around a number of key-papers in the area.

YEAR IV

6.12 UNITS IN INFORMATICS (Old Regulations)

CIS4010 Advanced Databases: Data Models and Database Languages

Lecturer: Mr. J. Vella
Prerequisites: CIS3010
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 1st & 2nd

This is a unit for students that want to consider doing a study on current research topics in databases and data modelling.

The relational data model and its database languages (for example SQL) go some way to satisfy the needs of computerised systems. On the other hand, sophisticated computer assisted applications (for example CASE, CAX and GIS) apply shearing pressure on a relational DBMS. One of their unsatisfied needs is powerful modelling capabilities which have really been long advocated for in the database community.

Two advanced data modelling approaches, of the many proposals, are the object-oriented and the temporal. What are these data models and how do these two models satisfy the needs of CAX is the main motivation of this unit. For each data model the discussion also addresses the status of the respective standards (for the definition, access and control language) and their realisation as a DBMS’s data model.

Ultimately the class has to address the possibility and gains of having to design an Information System’s repository with a number of different data models and database languages.

The latter part of the Advanced Database Systems double credit is concerned with the integration of databases with deduction and knowledge representation schemes - deductive databases or knowledge-based systems are the generic names given to these systems. Specifically deductive databases infer new information from the database’s rules and facts; this data is modelled and maintained by the database.
Important operational aspects of a deductive database management tool are that its “data” (specifically rules and facts) is resident on secondary storage systems and there are many more facts than there are rules in an application domain.

Of special interest during the study of deductive databases are their:
- data definition language for access of the deductive database system;
- query processing mechanisms to utilise the knowledge of the database when responding to queries;
- integrity constraints and constraint checking mechanisms;
- aggregate constraints and their enforcement mechanisms;
- characterisation of a transaction model.

During this unit an example is used to present a deductive database as an ideal back-end to an Expert System and / or Decision Support System. Also a research prototype is used for practical hands on experience.

Direct student input is expected.

**Method of Assessment:**
Exam (85%) Assignment (15%)

**Textbooks:** Research papers from journals

►CIS4020 Developing and Managing IS/IT operations

Lecturer: Prof. A. Leone Ganado, Mr. C. Meli, Mr. T. Spiteri Staines
Prerequisites: CIS3020
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 1st

- Development Methodologies
- System Administration including audit and security issues
- Simulation of Decision environments in IT management
- Project Management methodologies
- Downsizing and rightsizing issues
- Ethical and professional issues in managing the IT function

**Method of Assessment:** Exam (80%) Assignment (20%)

**Textbooks:**
- Duling Beray Ltd, *Series on Project Control*

►CIS4040 IS strategic analysis and business implications

Lecturer: Prof. A. Leone Ganado, Mr. R. Naudi, Mr. T. Spiteri Staines
Prerequisites: CIS3040
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14  
Semester: 1st and 2nd

- Capability Maturity Model
- A strategic Analysis Toolkit
- Risk Assessment Analysis
- Competitive Advantage
- SIS planning methodologies
- Business Process Re-engineering & TQM
- Use of Distributed Systems, Data warehousing, and Resource Management as a strategic tool
- Simulation of Decision environments.

**Method of Assessment:**  
Exam (80%) Assignment (20%)

**Textbooks:**
- Wendy Robson, *Strategic Management and Information Systems*, Pitman  
- D S J Remenyi *Strategic Information Systems Planning (SISP)* ISBN 0850128145 Blackwell  

**CIS4050 Information Systems Engineering**

Lecturer: Mr. R. Naudi  
Prerequisites: CIS3050  
ECTS Credits: 4  
Lectures: 28  
Tutorials/Practical Labs: 14  
Semester: 1st

The use of diagrammatic notation to capture and model various aspects of information systems. Induced subjectivity and re-engineering issues. Standard Data Flow Diagrams as a basis to define a complementary set of techniques and notations designed to be more rigorous and, at the same time, intuitive to master.

Two views of an information system are considered – the static model, to visualise the time-independent interactions between entities, and the dynamic model, capturing the system's path of execution. A descriptive set of notations to capture the meaning behind symbols used and the path flows between events is presented. Their relation with executable specifications, examining and dynamic screen interfacing is examined.

An in-depth case study will be used throughout the course to illustrate the various issues discussed throughout the lectures.

Reverse and Re-engineering. General considerations of re-use within different life cycles.

**Method of Assessment:** Exam (60%) Assignment (40%)

**SOR1341 Dynamic Programming**

Lecturer: Dr. J. Sklenar  
Credits: 2 ECTS
Prerequisites: Pure or Applied Mathematics at the level requested for entry to engineering and technology courses
Lectures: 14 hours, tutorials
Semester: 1

- Formulation of the general problem
- Multistage decision model
- Separability and Optimality conditions
- Selected applications
- Stochastic Dynamic Programming

Method of Assessment: Test

Suggested Texts:


**► SOR1351 Programming in Simula**

Lecturer: Dr. J. Sklenar
Credits: 2 (ECTS)
Prerequisites: The unit is based on previous knowledge on Computer Simulation and the basic ideas of Object Oriented Programming
Lectures: Introductory lectures and tutorials will be given, but most of the work will be carried on individually.
Semester: 1

Method of Assessment: Project

Suggested Texts:


**► CIS4080 Advanced Client/Server Computing: the object-oriented approach**

Lecturer: Dr. V. Nezval
Prerequisites: CIS3080
ECTS Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 1st

Services of Client/Server model - objects (data exchange, sharing, retrieval, update) and control of server applications.
Distributed object technology and object interoperability using industry standards such as Common Object Model (COM) and OM6’s CORBA.

Methods of object sharing / exchange:
- OLE - linking and embedding of objects shared,
- COM, OCX and ActiveX servers
- DCOM model and technology

CORBA – ORB architecture and features. Automation services, static method binding, dynamic method invocation, methods of object repository access and updating. CORBA middleware services.

Tools for Client/Server Computing - Delphi, JAVA SDKs’. Use for practical implementation of various client/server implementations in windowing environment.

**Method of Assessment:**
Exam (80%) Assignment (20%)

**Textbooks:**

### 6.13 UNITS IN COMPUTER SCIENCE (Old Regulations)

**CSA4010 - Distributed and Concurrent Systems II**

Lecturer: Dr. Kevin Vella  
Semester: 1st  
Prerequisites: CSA3010  
ECTS Credits: 4  
Tutorials / Practicals: 8 Hrs  
Lectures: 20 Hrs

This unit investigates distributed systems from a real-world perspective by taking on a large case study in the form of the Internet. Fundamental Internet technologies such as the TCP/IP protocol stack, network addressing, the IP routing mechanism, subnetting, ICMP, the TCP protocol including sliding windows and congestion avoidance, DNS, dynamic routing protocols such as RIP and OSPF, security (firewalls, packet filters and proxies) and various application level protocols (FTP, Telnet, HTTP, STMP, etc.), will be dissected. The design rationale and implementation of these protocols and systems will be studied in depth.

Other topics in the general area of system software, concurrency and distribution will also be covered.

**Method of Assessment:**  
Test: (70%) Coursework: (30%)

**Textbooks:**
CSA4030 - Scientific Computing

Lecturers: Mr. Joseph Cordina, Mr. Sandro Spina
Semester: 1st
Prerequisites: PHY2380, SOR0211, CSA2020 (This credit was offered during Academic Year 2003 – 2004), CSA3010
ECTS Credits: 4
Tutorials / Practicals: None
Lectures: 28 Hrs

The aim of this course is to introduce some mathematical methods and computational techniques for the simulation of real-life situations.

The first part of this course will cover the theoretical and mathematical foundations applicable to scientific computing. Most of the emphasis will be placed on Queuing theory, Markov models and an introduction to Chaos theory.

The second part of the course will cover problems with implementing the above models on computers. The limitations of traditional computer architectures will be discussed in terms of floating-point arithmetic and random number generation and their relationships to errors, stability and convergence. This course will be taught using several sample problems showing how the techniques introduced can be applied to real-life scenarios.

Method of Assessment: Test: (100%)

Textbooks: TBA

CSA4040 - Advanced Topics in Software Engineering

Lecturer: Dr. Ernest Cachia
Semester: 1st
Prerequisites: CSA3040
ECTS Credits: 4
Tutorials / Practicals: 8 Hrs
Lectures: 20 Hrs

The aim of this unit is to introduce students to the more specialised topics of software engineering, which include reliability, availability, fault/testing metrics, and real-time and concurrent system specification and scheduling theory and to introduce, and further explain to, the candidate the intricacies of software metric extraction and application (metrication). Topics such as Function and Object Point Analysis, Reliability and Availability functions, Stochastic state analysis, system testing criteria calculation, and similar other topics
will be discussed. A more in-depth look into UML will also be included. Depending on time and progress, hands-on experience of cutting-edge RAD development will be supplied. Some basic aspects of model checking will also be discussed. Whenever possible theoretical material will be supplemented with practical examples.

Method of Assessment:  
- Test: (80%)  
- Coursework: (20%)  
  [If hands-on RAD is not included in the coursework]
- Test: (70%)  
- Coursework: (30%)  
  [If hands-on RAD is included in the coursework]

Textbooks:

Additional Textbooks:

CSA4050 - Advanced Topics in Natural Language Processing

Lecturer: Mr. Michael Rosner
Semester: 1st
Prerequisites: CSA3050
ECTS Credits: 4
Tutorials / Practicals: 8 Hrs
Lectures: 20 Hrs

This course has two main aims: to deepen the student's knowledge of a number of specific NLP topics, and to make use of the algorithms introduced in CSA3050. Selections from amongst the following application areas will be studied:

- Computational semantics.
- Discourse.
- Information Extraction. Named entity extraction. Anaphora and co-reference resolution.
- Machine translation.

Method of Assessment:  
- Test: (95%)  
- Coursework: (5%)

Textbooks:
- Website:  
  http://www.cs.um.edu.mt/~mros/csa4050
When designing safety-critical systems, it is not sufficient to test the software / hardware written, but it may be necessary to mathematically verify that the system works correctly. This course explores different topics in this research area, including mathematical tools (such as logics, and property languages), formal modeling of computer languages and systems, and techniques used for automatic model checking of such systems.

Syllabus:
- Temporal Logic, process calculi.
- Synchronous programming, circuit semantics.
- Model Checking of software and hardware.

Method of Assessment: Coursework/Presentation: (100%)

Students will each be presenting a different key-paper in the area of formal methods and will have to hand in a detailed review of their paper. They will also have to present a report on how 5 other papers presented by other students relate to their paper.

Textbooks:
- The course will be based around a number of key-papers in the area.

CSA4070 - Machine Learning and Pattern Recognition

Both pattern recognition and machine learning belong to the most advanced areas of A.I. Numerical methods combined with A.I. techniques have been especially successful in pattern recognition. Research in Machine Learning is now recognised as one of the most important areas of A.I. as well as having application to knowledge acquisition in A.I. systems and contributing to the understanding of human cognition. This two-credit course is divided into two parts are follows:

Part (I): Machine Learning

Topics discussed:
- Principles of learning machines, Gold’s Theorem
• Concepts and Categories in Cognitive Science
• Computational learning theory (COLT)
• PAC-learning
• Grammatical inference
• Concept learning
• Find-S, Candidate Elimination, and the ID-3 learning algorithms

Part (II): Pattern Recognition

Topics discussed:
• Clustering techniques
• Linear discriminant analysis
• Pattern Feature Extraction
• Pattern Understanding
• Advanced Neural Networks (Hopfield, Kohonen networks)
• Support Vector Machines

Method of Assessment: Test: (100%)

Textbooks:
• Course notes and WWW links

CSA4080 – Adaptive Hypertext Systems II

Lecturer: Dr. Christopher Staff
Semester: 1st
Prerequisites: CSA3080
ECTS Credits: 4
Tutorials / Practicals: None
Lectures: 28 Hrs

This course builds upon CSA3080 (Adaptive Hypertext Systems I). Adaptive Hypertext Systems discover, represent, and manipulate user interests and requirements as users navigate and search through a hyperspace, and then adapt the organisation and the presentation of information accordingly. We take an in-depth look at fields of User and Domain Modelling, Information Retrieval, Intelligent Tutoring Systems, and Hypertext, which all contribute to Adaptive Hypertext Systems.

Method of Assessment: Test: (100%)

Textbooks (Recommended):
• Website:
CSA4110 - Intelligent Software Agents

Lecturers: Dr. Matthew Montebello, Mr. Charlie Abela
Semester: 1st
Prerequisites: CSA3110
ECTS Credits: 4
Tutorials / Practicals: None
Lectures: 28 Hrs

The main topic of this course is the employment of software agents to practical and intelligent applications. We will look into technologies for implementing intelligent agents. Agents differ from conventional (object-oriented) software technology in one or more of the following four dimensions: autonomy, pro-activeness, embeddedness and distributedness. The course builds on existing knowledge in AI and agent technology in general and focuses on areas of agent application like electronic commerce, recommendation systems, auctions, information retrieval over the WWW, and other commercial and cutting-edge scenarios.

Method of Assessment: Test: (80%) Coursework: (20%)

Textbooks:
- Website: http://staff.um.edu.mt/mmon1/lectures/csa4110/

6.14 UNITS IN COMMUNICATION AND COMPUTER SYSTEMS ENGINEERING (Old Regulations)

CCE 4001 – Advanced Computer Architecture

Lecturer: Dr C J Debono
Credits: 4 ECTS
Lectures: 2 hrs/wk
Tutorials: 4 hrs
Prerequisite: CCE 2002
Lecturer: Dr Ing. Carl James Debono
Semester: 1

Pipeline Architectures
- Techniques used to enhance throughput including: instruction lookahead, execution overlap (pipelining).
- Branching, data, and instruction dependencies.
- CISC, RISC and superscalar architectures.
- Fast multipliers and barrel shifters.
- Efficiency of pipeline hardware.

Parallel Architectures
- Parallel Processing.
- Parallel Architectures such as: Mesh, Star, Ring, Hypercube.
- Switching Elements.
• Arithmetic pipelines.
• Performance Considerations.

Assessment
Coursework – 10% Exam – 90%

Textbooks

CCE 4013 – Real-Time Networks

Lecturer: Dr C J Debono
Credits: 4 ECTS
Lectures: 2 hrs/wk
Tutorials: 4 hrs
Prerequisite: CCE 2002, CCE 3021
Lecturer: Dr Ing. Carl James Debono
Semester: 1

Real-Time System Modeling
• Concepts of soft real-time and hard real-time design and modeling.
• Concepts and generation of behavioral models.
• Real-time computing, static and dynamic priority scheduling.
• Rate monotonic scheduling.
• Fault tolerant systems.

Neural Networks
• Multilayer perceptron, Radial basis function networks, Kohonen networks, Hopfield networks.
• Recurrent neural networks.
• Supervised and Unsupervised learning.
• Parallel architectures for neural networks.

Assessment
Coursework – 10% Exam – 90%

Textbooks

►CCE4004 Computer Networks II

Lecturer: Dr. S. Zammit
Prerequisites: CCE3022
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 1

Wide Area Networks

- Point-to-Point Protocols: ASCII data link control, HDLC, X21.
- Frame Relay: framing, packet control and error recovery.
- Session Layer: end-to-end transmission using various protocols such as TCP/IP, X25, ATM, X-400.

Method of Assessment:
Examination (90%) Coursework (10%)

Textbooks:
- Halsall *Data Communications, Computer Networks, and Open Systems* (4th edition) Addison-Wesley

► CCE4101 Communication Systems

Lecturer: Dr. A. Muscat
Prerequisites: CCE3102
Credits: 4
Lecturers/Tut: 28 hrs
Labs: 14 hrs
Semester: 1

Objective
Communication systems for the transmission of multi-media information.

Detailed Syllabus

*General considerations in the design of communication systems*
- System capacity
- Traffic load profiles
- Trunking efficiency
- Models for communication systems
- Multiple access techniques and wireless protocols
- Bandwidth requirements and spectrum utilisation

*The above concepts are illustrated by considering the following*
- Public mobile radio systems, TETRA
- Cordless telephone systems, DECT, BlueTooth
- Cellular telephone systems, GSM, CDMA
- Cable UHF and satellite television
- Strong devices: magnetic and optical media; DVE, CD-ROM
- Cable and radio modems, ADSL, X-25

Assessment:
Exam 90% Coursework 10%

Textbooks:

► CCE4201 Digital Signal Processing

Lecturer: Prof. P. Micallef
Prerequisites: CCE3201
Credits: 4
Lectures: 28  
Tutorials/Practical Labs: 14  
Semester: 1

- Discrete Fourier Transform: Circular convolution. Relationship between z-transform and DFT.
- Fast Fourier Transform.
- Hilbert Transform.
- Homomorphic Filtering. The cepstrum and its applications
- Energy and power in a DSP system. The periodogram
- Time-Frequency Relationships: Windows. The short time DFT.
- Special transforms: WignerVille, Wavelet.
- Two dimensional DSP for image processing.

Method of Assessment:  
Examination (90%) Coursework (10%)

Textbooks:  

► CCE 4102 Coding for Communications Systems

Lecturer: Dr V. Buttigieg  
Prerequisites: CCE3201  
Credits: 4  
Lectures: 28  
Tutorials/Practicals: 14  
Semester: 1

Objective  
This unit deals with the efficient transmission of information over insecure and noisy channels.

Detailed Syllabus  
Source coding  
Information theory and the source coding theorem  
Basic code properties  
Run-length coding  
Huffman coding  
Arithmetic coding  
Lepel Ziv coding  
Lossy source coding  
Sound (mp3), image (jpg) and video coding (mpeg)

Error-Control Coding  
The channel coding theorem  
Hamming distance  
Random error detection and correction capability of linear codes  
Syndrome decoding  
Cyclic codes  
Convolutional codes  
Trellis coded modulation  
Cryptography  
Shannon's theory of cryptography  
Classical cryptosystems  
Types of attacks  
Perfect security – the one time pad  
Unicity distance  
Homophonic coding
Data Encryption Standard (DES)
Public key cryptosystems
Authentication
Digital signatures

Assessment:
Examination 90% Coursework 10%

Reading List:

►ELE4607 Introduction to Digital VLSI

Lecturer: Prof. J. Micallef
Prerequisites: ELE3607
Credits: 4
Lectures: 28
Tutorials/Practical Labs: 14
Semester: 2

Part (i)

Logic implementation. Combinational logic. Random logic (inverter, NAND, NOR gates) NMOS and CMOS implementation - transistor sizes and stick diagrams.

Structured logic - transistor switch arrays, distributed gate arrays, programmable logic arrays. Static and dynamic flip-flops. MOS clocking schemes - dynamic MOS storage circuits, clocked CMOS logic.

Laboratory Work
Semi-custom design techniques of a digital integrated circuit using the CADENCE simulation, routing and bonding, and loaded simulation, up to the shipment stage.

Method of Assessment: to be announced

Textbooks:
ISBN 0-07-100728-81
7. STUDENT GUIDE

B.Sc.IT(Hons.) Informatics, Computer Science, or Communication and Computer Systems Engineering

As mentioned in the introduction, this degree programme aims to produce graduates fit to occupy responsible positions in the Information Technology industry. Students registering on this programme are not required to know which stream of Information Technology they will ultimately choose. In fact, those who are undecided can postpone the choice of stream until the end of the second year. Having said that, students in consultation with their tutor, should be judicious in their choice of units each term as some units are prerequisite for others: the wrong choice may mean difficulty to follow advanced topics at a later stage.

Students should consult the entry requirements and regulations of this degree programme (Appendices A, B1 and B2 respectively).
Citation and Interpretation

1(1) These regulations may be cited as the Bachelor of Science in Information Technology - B.Sc.I.T. and Bachelor of Science in Information Technology (Honours) - B.Sc.I.T.(Hons.) - (Informatics, Computer Science, and Computer Systems Engineering) Degree Course Regulations, 1997.

(2) In these regulations, unless the context otherwise requires:

“the Board” means the Board of Studies for Information Technology appointed by Senate from members of the Departments of Computer Information Systems, Computer Science and Artificial Intelligence, and Communications and Computer Engineering.

“the Degree” except where this text is used specifically in relation to the ordinary B.Sc. I.T. degree referred to in these regulations means the Degree of Bachelor of Science in Information Technology (Honours) - B.Sc.I.T. (Hons.) - (Informatics or Computer Science or Computer Systems Engineering).

“the Course” means the course leading to the Degree.

Applicability

2 These regulations shall be applicable to courses starting in October 1995 and later.

Eligibility for the Degree

3 To be eligible for the Degree a candidate must:

(a) be registered as a regular student in the Course in terms of the Admission Regulations of the university;

(b) complete the course of studies, qualify in the examinations and satisfy any other requirements as prescribed in these Regulations; and

(c) satisfy any other requirements prescribed in any other relevant Statutes and regulations of the University.

Registration and special course requirements

4 (1) Subject to the provisions of paragraph (2) of this regulation, to be registered as a regular student in the Course a candidate must, in addition to the general entry requirements specified in the Admission Regulations of the University,

(a) for courses starting in 1995 and 1996, be in possession of two passes at Advanced Matriculation level at Grade C or better in

(i) Pure Mathematics, and

(ii) Another subject from the following list:
for courses starting in October 1997 or later, have obtained a pass with Credit or Distinction in Pure Mathematics and another subject included in the list shown in sub-paragraph (a) (ii) of the paragraph above, or as approved by the board, taken at Advanced Level as part of the Matriculation Certificate Examination.

or

(2) In terms of Regulation 6.3 of the Admissions Regulations be in possession of:

(i) five passes in the Secondary Education Certificate Examination including English Language, Maltese; and

(ii) a pass in Systems of Knowledge at Intermediate Matriculation Level; and

(iii) a pass at Advanced Level in Pure Mathematics at Grade C or better in the Matriculation Certificate Examination of the University or at corresponding level of other examinations recognised by Senate for the purposes of the Admission Regulations; and

(iv) either

(a) a Higher Technician Diploma of the City and Guilds of the London Institute Course No. 8000 in Electrical Engineering.

Or

(b) The Diploma in Industrial Electronics with Credit of the Fellenberg Training Centre.

Or

(c) The Advanced Diploma in Applied Information Technology of the City and Guilds of London Institute Course No. 7235.

Or

(d) any other qualification from a recognised Institute considered by the Board to be equivalent to any of (a), (b) or (c).

(2) A candidate, who is a graduate of a University or has other qualifications considered to be sufficient, may be exempted by the University Admissions Board, on the advice of the board of Studies Admissions Committee, from the whole or part of these special course requirements.
Appendix B1

Course Structure

5 (1) The Course shall normally extend over a period of four years of full-time study, and shall be divided into two parts: Part One and Part Two.

(2) In the first two years (Part One), a suite of units will be offered covering Informatics, Computer Science, and Computer Systems Engineering together with some complementary compulsory units in miscellaneous subjects such as Mathematics, Statistics, Operations Research, and Management. The aim of the first year is to give a broad perspective of various concepts in Information Technology and how they interrelate: the first year units are referred to as Introductory Units. The second year refines the spirit of the first by exposing students to foundation topics in some detail. Second year units are referred to as Intermediate Units.

(3) By the end of the second year, students must decide on a specific stream: Informatics, Computer Science, or Computer Systems Engineering. In the third and fourth years (Part Two), a candidate shall be expected to concentrate on the specific stream chosen as specified in paragraph 6 of this regulation. Part Two units are referred to as Advanced Units. Advanced Units are three-credit units divided into two parts: the first, worth one credit is covered in the third year (year 1 of Part two), the second worth two credits, in the fourth year (year 2 of Part two).

(4) The Degree shall be qualified by the name of the stream chosen at the end of the second year: Informatics, Computer Science, or Computer Systems Engineering.

(5) The Departments of Computer Information Systems, Computer Science and Artificial Intelligence, and Communication and Computer Engineering shall be responsible, respectively, for the Informatics, Computer Science, and Computer Systems Engineering streams.

(6) The Course shall consist of study-units to which not less than 103 credits are assigned as follows:

(a) in the 1st year, 28 credits: 21 credits for Introductory Units, 4 credits for complementary compulsory units, and 3 optional credits one of which may be an extra-curricular credit.

(b) in the 2nd year, 29 credits:
for those students who are in the 2nd year in 1997: 22 credits for Intermediate Units, 6 credits for complementary compulsory units, and 1 optional credit which may be an extra-curricular credit;

for those students who are in the 2nd year after 1997: 20 credits for Intermediate Units, 6 credits for complementary compulsory units, and 3 optional credits one of which may be an extra-curricular credit;

(c) in the 3rd year, 24 credits; 12 of which obtained for assigned practical tasks in the chosen stream, a minimum of 6 credits for the first part of Advanced Units obtained in the chosen stream, 6 credits obtained from amongst first part of Advanced Units of any stream; and

(d) in the 4th year, 22 credits: 10 of which obtained for an extensive project in the chosen stream as approved by the respective Department responsible, 8 or 10 credits for the second part of Advanced Units in the chosen stream, and the remaining 4 or 2 credits for the second part of Advanced Units in any stream other than the one chosen by the student on entry to Part Two of the course.

(7) The Board shall draw up a catalogue of study-units for each year of the course. The Catalogue shall indicate the credits assigned to each study-unit, which units are compulsory, and which units need pre-requisites. The Board shall publish the catalogue prior to the commencement of each academic year.

(8) The Board may exempt a candidate from the requirement of obtaining a study-unit if he has obtained from the University, or an institution recognised by Senate for the purpose, a qualification the study for which is, in the opinion of the Board, at least equivalent in content and standard to that required for the study-unit concerned.
(9) A candidate may not complete the course earlier than four years or later than five years from first registration, except in special circumstances with the permission of Senate, acting on the advice of the Board.

(10) Each year of the Course shall comprise two semesters of study, and, except in the third and fourth year, by a period of not less than four weeks of work in the Information Technology industry as advised by the Board.

Assessment and Progress

6 (1) Credits shall be assigned according to the following grades in descending order of merit: A, B, C, D and F where F indicates Fail. The grades shall take into account the candidate’s performance in coursework, oral, and written exams as deemed appropriate by the board on the advice of the respective Department responsible.

(2) (a) Candidates who, in the first year of the course fail to obtain 20 credits after their first sit, shall not be allowed re-sits and shall not be allowed to proceed to the next year of the course.

(b) Students who in the second year of the course fail to obtain 21 credits after their first sit, shall not be allowed re-sits and shall not be allowed to proceed to Part II of the course.

(3) A candidate who is not restricted from proceeding under the last preceding paragraph and who fails to obtain the credit(s) for a study-unit in the first session shall be allowed to be re-assessed in that study-unit only once. In the re-assessment, the grade awarded cannot exceed grade D.

(4) A candidate who after re-assessment, fails in four or more credits, or fails to successfully complete any compulsory study-unit shall not be allowed to proceed to the next year.

(5) Supplementary written exams shall be held only in September.

(6) A candidate who fails to satisfy the requirements to proceed to the next year of the Course shall be allowed to repeat the study programme for the year, provided that he has not already repeated a year during the Course (except in extraordinary circumstances with the approval of the Senate).

(7) Candidates who are allowed to repeat a year may at the discretion of the Board be exempted from being re-assessed in study-units for which they have obtained a credit during the failed year.

(8) In Part One, candidates shall be examined in study-units completed in a semester at the end of that semester.

(9) A candidate shall be allowed to proceed to part two if he has obtained a minimum of 53 of the 57 credits prescribed in sub-paragraph (a) and (b) of paragraph (6) of regulation 5, including all the compulsory credits. Moreover, the candidate must have completed to the satisfaction of the board at least 4 weeks of work in the I.T. industry as advised by the Board.

(10) In the first year of Part Two (the third year), candidates shall:

(a) obtain the twelve credits allocated to assigned practical tasks intended to bring together practical skills in some chosen stream. Grades shall be assigned according to the quality of a written report documenting the attempt at fulfilling the tasks, the quality of any implemented artefacts or results obtained and the candidate’s performance in an oral exam related to the subject-matter covered by the tasks;

(b) obtain a minimum of 8 of the 12 credits assigned to study-units according to performance in written exams. Study-units completed in a semester shall be examined at the end of that semester.
(11) A candidate who fails to satisfy the examiners in the practical tasks may, at the discretion of the Board, be re-examined through a viva-voce. If successful, the maximum grade that can be awarded shall be grade D. If unsuccessful, the candidate will fail the degree irrespective of performance in written papers unless he still has the option to repeat the year and chooses to do so.

(12) A candidate who fails in more than four credits of the third year study-units will fail the Degree unless he still has the option to repeat the year and chooses to do so. There will be no resits for third year credits.

(13) The performance of a candidate who meets the requirements in paragraph 10 of this regulation shall be determined by the average performance in the credits covered in the first three years. Candidates are classified in descending order of merit in categories I, IIA, IIB, III and Fail as determined for each stream. Candidates classified IIB or better shall be eligible to proceed to the fourth year of the course. A list of those candidates who are so eligible shall be published: if interested to proceed, these candidates should register their intent within three days from the publication of the list.

(14) Third year candidates who are not eligible or who do not wish to proceed to the fourth year shall be awarded a B.Sc. I.T. degree which shall be classified in the categories as listed in paragraph (13) of this regulation.

(15) In the second year of Part Two (the fourth year), candidates shall:

(a) obtain the ten credits allocated to the project in their chosen stream and grades shall be assigned according to the quality of a written project report and the quality of any implemented artefacts or results;

(b) pass in six two-credit examination papers, one for each Advanced Unit they have chosen and each Advanced Unit assumes that the candidate has covered a certain number of pre-requisite Introductory and Intermediate Units, therefore the examination papers in the fourth year will be synoptic in the sense that they will assume knowledge of related material covered in Part One and the first year of Part Two.

(16) A candidate who fails to satisfy the examiners with the project may, at the discretion of the Board, be allowed to re-submit the project or to complete some other appropriate task by not later than the end of September of that year. If successful, the maximum grade that can be awarded shall be grade D.

(17) A candidate who fails in not more than two written examinations may be re-examined through a viva-voce. If a candidate fails to satisfy the examiners in the viva-voce, or if a candidate has failed in more than two written examinations, then he may repeat the year if he still has that option.

(18) Any candidate who fails to satisfy the provisions of sub-paragraph 15 of this regulation despite the provisions in paragraph (16) and paragraph (17) of this regulation, and who has previously exhausted the option to repeat a year shall not be awarded an honours degree. In this case the candidate shall be awarded a B.Sc. I.T. degree in the category obtained in the third year as listed in paragraph (13) of this regulation.

(19) For the purposes of B.Sc.I.T. (Hons.) degree classification, the final year synoptic examinations and project shall have a 60% weighting and the performance in the first three years shall have a 40% weighting.

Results and Award of the Degree

7 (1) The Degree shall be awarded to candidates who

(i) have obtained not less than 95 credits throughout the course,
(ii) have successfully completed all the study-units indicated as compulsory by the Board, and
(iii) have satisfied examination requirements.

(2) The names of all candidates who qualify for the award of the Degree shall be published in a list in alphabetical order in one of the classifications:
- First Class Honours
- Second Class Honours (Upper Division)
- Second Class Honours (Lower Division)
- Third Class Honours.
EDUCATION ACT
(CAP. 327)

Bye-Laws of 2005 in terms of the General Regulations for University Undergraduate Awards, 2004 for the Degrees of Bachelor of Science in Information Technology (Honours) - B.Sc. I.T. (Hons) - and Bachelor of Science in Information Technology - B.Sc. I.T. - under the auspices of the Board of Studies for Information Technology

IN exercise of the powers conferred upon him by sections 30(5) and 31(6) of the Education Act (Cap. 327), the Chancellor of the University of Malta has promulgated the following bye-laws made by the Board of Studies for Information Technology in virtue of the powers conferred upon it by section 37 (1) of the said Act and which have been approved by the Senate of the University of Malta as required by section 37 (2) of the said Act:

Citation and Interpretation

1. (1) These bye-laws may be cited as the Bye-Laws of 2005 in terms of the General Regulations for University Undergraduate Awards, 2004 for the degrees of Bachelor of Science in Information Technology (Honours) - B.Sc. I.T. (Hons) - and Bachelor of Science in Information Technology - B.Sc. I.T. - under the auspices of the Board of Studies for Information Technology.

(2) In these bye-laws, unless the context otherwise requires -

"the Board of Studies" means the Board of Studies for Information Technology appointed by Senate from members of the Departments of Computer Information Systems, Computer Science and Artificial Intelligence, and Communications and Computer Engineering;

"the Course" means the course programme leading to the degree of Bachelor of Science in Information Technology (Honours) - B.Sc. I.T. (Hons) - or to the degree of Bachelor of Science in Information Technology - B.Sc. I.T. - as appropriate;

"the Degree" means the degree of Bachelor of Science in Information Technology (Honours) - B.Sc. I.T. (Hons) - or the degree of Bachelor of Science in Information Technology - B.Sc. I.T. - as appropriate;

"the Honours Degree" means the degree of Bachelor of Science in Information Technology (Honours) - B.Sc. I.T. (Hons);

"the Ordinary Degree" means the degree of Bachelor of Science in Information Technology - B.Sc. I.T.; and

"the Principal Regulations" means the General Regulations for University Undergraduate Awards, 2004.

Applicability

2. These bye-laws shall be applicable to courses commencing in October 2003 or later.
Areas of Study

3. The Degree shall be awarded in one of the following areas of specialisation:

(a) Informatics;
(b) Computer Science; and
(c) Computer Systems Engineering.

Special Course Requirements for Admission

4. (1) To be registered as regular students in the Course, applicants shall, apart from satisfying the general requirements for admission as specified in the Admission Regulations, be in possession of:

(a) passes at Advanced Matriculation Level at Grade C or better in Pure Mathematics and in one of the following subjects: Applied Mathematics, Biology, Chemistry, Computing, Economics, English, Information Technology, Music, Philosophy, and Physics;

or

(b) (i) five passes in the Secondary Education Certificate Examination including English Language, and Maltese; and
(ii) a pass in Systems of Knowledge at Intermediate Matriculation Level; and
(iii) a pass at Advanced Matriculation Level at Grade C or better in Pure Mathematics or at a corresponding level of other examinations recognised by Senate for the purpose of the Admission Regulations; and
(iv) either
(a) a Higher Technician Diploma of the City and Guilds of London Institute Course No. 8000 in Electrical Engineering;
(or)
(b) the Diploma in Industrial Electronics, with Credit, of the Fellenberg Training Centre;
(or)
(c) the Advanced Diploma of the City and Guilds of London Institute Course No. 7235 in Applied Information Technology;
(or)
(d) any other qualification from a recognised institute considered by the Board of Studies to be equivalent to any of (a), (b) or (c).

(2) As from October 2005, students who lack a grade in only one of the subjects of the special course requirements required at Advanced Matriculation Level indicated in paragraph (1) of this bye-law shall be admitted under those conditions as the Board of Studies may impose to compensate for the missing element. If, by the end of the first year, such students do not successfully complete all the requirements to progress regularly to the second year of the Course, they shall be required to withdraw from the Course and shall neither be entitled to repeat the year nor to progress conditionally as normally permitted under the Principal Regulations.

Course Duration

5. The Course shall extend over a period of four years of full-time study.
Course Programme

6. (1) The Course shall consist of two parts: Part One, comprising the first three years at the end of which the Ordinary Degree may be awarded, and Part Two, comprising the fourth year at the end of which the Honours Degree may be awarded.

(2) The first two years of the Course shall be common to all students and shall comprise study-units in Informatics, Computer Science, and Computer and Telecommunication Systems Engineering and in other complementary subjects.

(3) At the end of the second year of the Course, students shall be required to choose an area of specialisation from Informatics, Computer Science, and Computer and Telecommunication Systems Engineering on which they will be required to concentrate studies during the rest of the Course.

(4) At the end of Part One of the Course, students whose average mark calculated in accordance with the criteria specified in paragraph (2) of by-law 10 would entitle them to the Ordinary Degree with Category IIB or better shall be allowed to proceed to Part Two of the Course unless they opt to stop and be awarded the Ordinary Degree.

(5) Students who successfully complete Part One of the Course but whose average mark over the three years entitles them to the Ordinary Degree with Category III shall not be eligible to proceed to Part Two but shall be awarded the Ordinary Degree.

(6) Students who proceed to Part Two of the Course but who do not complete it successfully shall be entitled to the award of the Ordinary Degree.

(7) The course programme shall consist of study-units to which 240 credits are assigned as follows:

(a) in the first year, 60 credits from the three different areas of specialisation;

(b) in the second year, 12 credits in each area of specialisation; 12 credits from any area of specialisation; and 12 credits as directed by the Board of Studies;

(c) in the third year, 30 credits for Assigned Practical Tasks; and 30 credits, of which not less than 16 and not more than 20 credits are to be taken in the chosen area of specialisation; and

(d) in the fourth year, 24 credits for an extensive project in the chosen area of specialisation as approved by the department responsible for the area of specialisation; 28 credits, of which not less than 16 and not more than 24 credits are to be taken in the chosen area of specialisation and 8 other credits as directed by the Board of Studies.

7. The Board of Studies shall publish the programme of study for each area of specialisation drawn up in accordance with the relevant provisions of the Principal Regulations, and approved by Senate before the commencement of each Course.

8. In any year of the Course students may not register for study-units to which more than 60 credits are assigned unless they are required to register for referred study-units in terms of regulation 54 of the Principal Regulations.

9. Notwithstanding the provisions of regulation 45 of the Principal Regulations, in any year of the Course, students shall not be awarded more than 12 credits for study-units passed by compensation and if necessary the students shall be required to choose which study-unit/s of the eligible study-units are to be considered as passed by compensation, to a maximum value of 12 credits.
Classification of the Awards

10. (1) The Final Weighted Average Mark for the classification of the Ordinary Degree shall be based on the results obtained in Part One of the Course, each year of which shall be weighted as follows:

- First Year Average Mark 20%
- Second Year Average Mark 40%
- Third Year Average Mark 40%.

(2) The Final Weighted Average Mark for the classification of the Honours Degree shall be based on the results obtained in the four years of the Course, which shall be weighted as follows:

- Part One Weighted Average Mark as in paragraph (1) 60%
- Part Two (Fourth Year) 40%.