## Computer Systems Engineering wIE BEng (Hons) 2013-2014

### Summary

<table>
<thead>
<tr>
<th>UCAS code</th>
<th>Award</th>
<th>Title</th>
<th>Duration</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>HHQ6</td>
<td>Beng</td>
<td>Computer Systems Engineering wIE BEng (Hons)</td>
<td>4 years</td>
<td>FT + Placement year</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Schools</th>
<th>Faculty</th>
<th>Awarding Institution</th>
<th>Programme Accreditation</th>
<th>Relevant QAA benchmark(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Science</td>
<td>Engineering and Physical Sciences</td>
<td>University of Manchester</td>
<td>BCS, IET</td>
<td>Computing</td>
</tr>
</tbody>
</table>
Aims and intended learning outcomes

Sophisticated electronic systems permeate all aspects of life. Typical examples include MP3 players, games consoles, mobile phones, vehicle control systems and radar. All of these are embedded systems, which typically contain one or more microprocessors, memory, a communications capability, and application-specific hardware and software. Our Computer Systems Engineering programmes aim to develop the wide range of knowledge and skills are needed to support their development, including digital electronics, software engineering and computer architecture.

Our programmes aim to:

1) enable graduates to exhibit a high level of practical and theoretical skills over a broad range of Computer Science together with a knowledge of currently available techniques and technologies.
2) explore the principles that support developments in a rapidly changing subject.
3) provide opportunities for students to understand the wide range of research challenges facing Computer Science, as well as the breadth and depth of research undertaken in this top-rated school, so they are prepared to embark on research here or elsewhere.
4) develop competent professionals able to play a leading part in many different commercial, industrial and academic activities and adapt rapidly to changing technology.
5) meet industry demand for high calibre graduates who will take a lead in continuing technological change.
6) prepare students for the social, organisational and professional context in which they will be working.
7) meet the educational requirements of the Engineering Council thus enabling graduates to progress to professional membership of the BCS and IET and attain the highest professional status of Chartered Engineer.
8) In addition, the with Industrial Experience programmes aim to: give extensive practical experience of an industrial or business environment where students are able to apply and develop their skills, both technical and personal.
9) In addition, the MEng programmes aim to: prepare high fliers for professional practice in Computer Science by enhanced depth and breadth of study together with increased emphasis on industrial relevance through industrially related group projects.

Intended learning outcomes

Knowledge & understanding

A1 Know and understand the essential mathematics relevant to computer science.
A2 Understand and apply a wide range of principles and tools available to the software engineer, such as design methodologies, choice of algorithm, language, software libraries and user interface techniques
A3 Demonstrate a grasp of the principles of computer systems, including architecture, networks and communication
A4 Recognise and appreciate the professional and ethical responsibilities of the practising computer professional, including understanding the need for quality
A5 Know and understand the principles and techniques of a number of application areas informed by the research directions of the subject, such as artificial intelligence, databases and computer graphics
A6 Apply their knowledge of computing in a commercial or industrial context
A7 Show a critical understanding of the broad context within which Computer Science resides, including issues such as quality, reliability, enterprise, employment law, accounting and health and safety
A8 Have a comprehensive knowledge and critical awareness of selected specialist fields at the forefront of computer science, studied at masters level

Intellectual (thinking) skills

B1 Solve a wide range of problems related to the analysis, design and construction of computer systems.
B2 Design and implement a software or hardware system of significant size
B3 Identify a range of solutions and critically evaluate and justify proposed design solutions

B4 Solve computer science problems with pressing commercial or industrial constraints

B5 Generate an innovative design to solve a problem containing a range of commercial and industrial constraints

**Practical skills**

C1 Plan and undertake a major individual project

C2 Prepare and deliver coherent and structured verbal and written technical reports

C3 Give technical presentations suitable for the time, place and audience

C4 Use the scientific literature effectively and make discriminating use of Web resources

C5 Design, write and debug computer programs in appropriate languages

C6 Use appropriate computer-based design support tools

C7 Apply computer science skills in a commercial or industrial environment

C8 Demonstrate initiative taking, innovation and self-management in an industrially related group project

C9 Integrate previously acquired skills and apply them to new, demanding situations

**Transferable skills**

D1 Display an integrated approach to the deployment of communication skills

D2 Use IT skills and display mature computer literacy

D3 Work effectively with and for others

D4 Strike the balance between self-reliance and seeking help when necessary in new situations

D5 Display personal responsibility by working to multiple deadlines in complex activities

D6 Employ discrete and continuous mathematical skills as appropriate

D7 Demonstrate significantly enhanced group working abilities

D8 Further develop career plans and personal objectives

D9 Communicate effectively with non-specialist as well as computer scientist professionals at a range of levels

D10 Undertake a range of technical roles within a team and be able to display leadership
Teaching, learning and assessment methods

Learning and Teaching on all our programmes aims to combine an understanding of fundamental CS principles, development of strong practical skills and the group-working, learning and communication skills that are essential for any computing professional.

Course units which involve practical elements all have associated laboratory exercises, usually in timetabled sessions with staff and demonstrator support. Most labs operate a system of face-to-face marking in the lab so that students receive immediate feedback on their work. Units without labs all have regular coursework exercises to support skills development and feedback.

Most units are lecture based, with lab or coursework exercises used to reinforce and enhance knowledge and skills first encountered in lectures. The first year team project deviates significantly from this model and takes an Enquiry Based Learning approach. This unit aims to encourage students to be more actively engaged with, and responsible for, their own learning, to develop skills in problem solving, communication, independent learning, and group work, and to signal the importance we attach to independent learning. This approach is followed up in the second year with the workshop based approach used in the compulsory Software Engineering unit, which also contains a major group working component.

The choice of units offered to students on our programmes is very broad, but they also require depth in particular subject areas. This is achieved using the notion of Themes. A theme is a group of related second and third year course units which form a coherent whole; usually one second year and two third year units. All single honours students are required to complete at least two themes; those on the Computer Science programme can choose any pair of themes, but those on specialist programmes must complete at least two themes associated with their specialism. It is the choice of these themes that characterises the specialist programmes. The Programme Structure section below indicates the Themes for this group of programmes and the tables in that section show the Theme for each course unit.

All students undertake an individual 3rd year project, supervised by a member of academic task, which usually involves the development of significant software or hardware product. Assessment of this unit involves presentations of plans and results and a major written report.

Assessment in almost all units is a combination of lab/coursework and examination.

Learning, Teaching and Assessment of intended learning outcomes

Knowledge and Understanding

<table>
<thead>
<tr>
<th>Learning and Teaching Processes</th>
<th>Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lectures (A1, A2, A3, A4, A5, A6, A7)</td>
<td>Unseen written examinations (A1, A2, A3, A4)</td>
</tr>
<tr>
<td>Laboratory sessions (A2, A3)</td>
<td>Marked tutorial exercises (A1, A2, A3, A4)</td>
</tr>
<tr>
<td>Personal tutorials (A1, A2, A3, A4, A7)</td>
<td>Laboratory reports (A2, A3)</td>
</tr>
<tr>
<td>Problem solving classes (A1, A2, A3, A4)</td>
<td>Project reports (individual and group) (A3, A4, A5, A6, A7)</td>
</tr>
<tr>
<td>Problem-based learning (A2, A3, A4, A5, A6, A7)</td>
<td>Oral presentations (individual and group) (A3, A4, A5, A6, A7)</td>
</tr>
<tr>
<td>Projects (A3, A4, A5, A6, A7)</td>
<td></td>
</tr>
<tr>
<td>Industrial seminars (A4, A5, A6, A7)</td>
<td></td>
</tr>
</tbody>
</table>

Intellectual Skills

<table>
<thead>
<tr>
<th>Learning and Teaching Processes</th>
<th>Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lectures (B1, B2, B4, B5)</td>
<td>Unseen written examinations (B1, B2, B4)</td>
</tr>
<tr>
<td>Laboratory sessions (B1, B2)</td>
<td>Marked tutorial exercises (B1, B2)</td>
</tr>
<tr>
<td>Personal tutorials (B1, B2, B4)</td>
<td>Laboratory reports (B1, B2)</td>
</tr>
<tr>
<td>Problem solving classes (B1, B2, B4)</td>
<td>Project reports (individual and group) (B1, B2, B3, B4, B5)</td>
</tr>
<tr>
<td>Problem-based learning (B1, B2, B4)</td>
<td>Oral presentations (individual and group) (B1, B2, B3, B4, B5)</td>
</tr>
<tr>
<td>Projects (B1, B2, B3, B4, B5)</td>
<td></td>
</tr>
</tbody>
</table>

Practical Skills

<table>
<thead>
<tr>
<th>Learning and Teaching Processes</th>
<th>Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lectures (C4, C6)</td>
<td>Laboratory reports (C1, C2, C3, C4, C5, C6, C7)</td>
</tr>
<tr>
<td>Laboratory sessions (C1, C2, C3, C4, C5, C6, C7)</td>
<td>Project reports (individual and group) (C3, C4, C5, C6)</td>
</tr>
<tr>
<td>Projects (C3, C4, C5, C6, C9)</td>
<td>Oral presentations (individual and group) (C6, C8, C9)</td>
</tr>
</tbody>
</table>
Transferable skills

<table>
<thead>
<tr>
<th>Learning and Teaching Processes</th>
<th>Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lectures (D3, D4, D5, D7)</td>
<td>Laboratory reports (D1, D3, D5, D6)</td>
</tr>
<tr>
<td>Laboratory sessions (D1, D3, D5, D6)</td>
<td>Essays (D2, D3)</td>
</tr>
<tr>
<td>Personal tutorials (D1,D2, D3, D4, D7)</td>
<td>Project reports (individual and group) (D1, D2, D3,D4, D5, D6, D7)</td>
</tr>
<tr>
<td>Problem solving classes (D4)</td>
<td>Oral presentations (individual and group) (D1, D3, D6, D7, D8)</td>
</tr>
<tr>
<td>Problem-based learning (D1, D2, D3, D4, D5, D6)</td>
<td>Industrial placement reports (D8)</td>
</tr>
<tr>
<td>Projects (D1, D2, D3, D4, D5, D6, D7)</td>
<td></td>
</tr>
<tr>
<td>Industrial placement (D8)</td>
<td></td>
</tr>
</tbody>
</table>
Programme structure

Level 1 - compulsory units
All of the units in this pool are mandatory.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP10120</td>
<td>First Year Team Project</td>
<td>20</td>
</tr>
<tr>
<td>COMP11120</td>
<td>Mathematical Techniques for Computer Science</td>
<td>20</td>
</tr>
<tr>
<td>COMP11212</td>
<td>Fundamentals of Computation</td>
<td>10</td>
</tr>
<tr>
<td>COMP12111</td>
<td>Fundamentals of Computer Engineering</td>
<td>10</td>
</tr>
<tr>
<td>COMP14112</td>
<td>Fundamentals of Artificial Intelligence</td>
<td>10</td>
</tr>
<tr>
<td>COMP15111</td>
<td>Fundamentals of Computer Architecture</td>
<td>10</td>
</tr>
<tr>
<td>COMP16121</td>
<td>Object Oriented Programming with Java 1</td>
<td>20</td>
</tr>
<tr>
<td>COMP16212</td>
<td>Object Oriented Programming with Java 2</td>
<td>10</td>
</tr>
<tr>
<td>COMP18112</td>
<td>Fundamentals of Distributed Systems</td>
<td>10</td>
</tr>
</tbody>
</table>

Level 2 options
You have 90 credits of compulsory course units listed in the table "compulsory units" below.

Out of the remaining 30 credits of free choice:

You can choose a minimum of 10 credits for optional COMP course units from "option pool 1" and "option pool 2" below AND/OR you can choose up to 20 credits of optional course units that are external to the Department. You can choose any Level 1 or 2 options for which you meet any pre-requisites and fits with your timetable, these may be:

- Business and Management course units: https://ughandbook.portals.mbs.ac.uk/Non-AllianceMBSstudents.aspx
- University College course units
- Language course units: https://www.languagecentre.manchester.ac.uk/learn-a-language/courses-for-students/

Please note: to enrol on some external course units (such as Language) will require permission from the associated School/Department.

To select any external course units outside of the list given above will require permission from the <a href="/ugt/index.php#tutors">2nd Year Tutor</a>.

If you take a 20 credit whole year course unit you are not permitted to drop this unit when course unit selection reopens at the start of semester 2.

You must ensure your credits are balanced over the academic year (60 credits in each semester).

This programme requires 2 themes to be completed from the following list.

* Computer Architecture (COMP25111, COMP25212 & COMP35112)
* System-on-Chip (COMP22111 & COMP32212)

Level 2 - compulsory units
All of the units in this pool are mandatory.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
<th>Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP22111</td>
<td>Processor Microarchitecture</td>
<td>10</td>
<td>System-on-Chip</td>
</tr>
<tr>
<td>COMP22712</td>
<td>Microcontrollers</td>
<td>10</td>
<td>None</td>
</tr>
<tr>
<td>COMP23111</td>
<td>Fundamentals of Databases</td>
<td>10</td>
<td>Web and Distributed Systems</td>
</tr>
<tr>
<td>COMP23420</td>
<td>Software Engineering</td>
<td>20</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>COMP25111</td>
<td>Operating Systems</td>
<td>10</td>
<td>Computer Architecture</td>
</tr>
<tr>
<td>COMP25212</td>
<td>System Architecture</td>
<td>10</td>
<td>Computer Architecture</td>
</tr>
<tr>
<td>COMP26120</td>
<td>Algorithms and Imperative Programming</td>
<td>20</td>
<td>Programming and Algorithms</td>
</tr>
<tr>
<td>COMP28512</td>
<td>Mobile Systems</td>
<td>10</td>
<td>Mobile Computing and Networks</td>
</tr>
</tbody>
</table>

Level 2 - option pool 1
From this option pool choose 10 credits.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
<th>Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP21111</td>
<td>Logic and Modelling</td>
<td>10</td>
<td>Rigorous Development</td>
</tr>
<tr>
<td>COMP24111</td>
<td>Machine Learning and Optimisation</td>
<td>10</td>
<td>Learning and Search in Artificial Intelligence</td>
</tr>
<tr>
<td>COMP28411</td>
<td>Computer Networks</td>
<td>10</td>
<td>Mobile Computing and Networks</td>
</tr>
<tr>
<td>UCIL20021</td>
<td>Leadership in Action</td>
<td>10</td>
<td>None</td>
</tr>
</tbody>
</table>

Level 2 - option pool 2
From this option pool choose 10 credits.
**COMP24412** Symbolic AI 10 Natural Language, Representation and Reasoning
**COMP27112** Computer Graphics and Image Processing 10 Visual Computing
**COMP28112** Distributed Computing 10 Web and Distributed Systems
**HSTM20282** The Information Age 10 None
**UCIL20022** Leadership in Action 10 None

## Level 3 options

You have 60 credits of compulsory course units listed in the table "compulsory units" below.

Out of the remaining 60 credits of free choice:

You must choose at least 40 credits of optional COMP course units from option pool 1 - 3 below. Maximum 30 from "option pool 1", maximum 30 from "option pool 2" and maximum 20 from "option pool 3". The minimum number of credits of optional COMP course units selected is 40 and the maximum is 60.

If you take a 20 credit whole year course unit you are not permitted to drop this unit when course unit selection reopens at the start of semester 2.

You can also choose up to 20 credits of optional course units that are external to the Department. You can choose any Level 2 or 3 options for which you meet any pre-requisites and fits with your timetable, these may be:

- Business and Management course units: https://ughandbook.portals.mbs.ac.uk/Non-AllianceMBSstudents.aspx
- University College course units
- Language course units: https://www.languagecentre.manchester.ac.uk/learn-a-language/courses-for-students/

Please note: to enrol on some external course units (such as Language) will require permission from the associated School/Department.

To select any external course units outside of the list given above will require permission from the <a href="/ugt/index.php#tutors">3rd Year Tutor</a>.

You must ensure your credits are balanced over the academic year (60 credits in each semester).

This programme requires 2 themes to be completed from the following list.

* Computer Architecture (COMP25111, COMP25212 & COMP35112)
* System-on-Chip (COMP22111 & COMP32212)

### Level 3 - compulsory units

All of the units in this pool are mandatory.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
<th>Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP38040</td>
<td>3rd Year Project (Single Honours 40 Credits)</td>
<td>40</td>
<td>None</td>
</tr>
<tr>
<td>COMP32212</td>
<td>Implementing System-on-Chip Designs</td>
<td>10</td>
<td>System-on-Chip</td>
</tr>
<tr>
<td>COMP35112</td>
<td>Chip Multiprocessors</td>
<td>10</td>
<td>Computer Architecture</td>
</tr>
</tbody>
</table>

### Level 3 - option pool 1

From this option pool choose a maximum of 40 credits and a minimum of 20 credits.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
<th>Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP31111</td>
<td>Verified Development</td>
<td>10</td>
<td>Rigorous Development</td>
</tr>
<tr>
<td>COMP33411</td>
<td>Natural Language Systems</td>
<td>10</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>COMP34411</td>
<td>Advanced Algorithms I</td>
<td>10</td>
<td>Natural Language; Representation and Reasoning</td>
</tr>
<tr>
<td>COMP36111</td>
<td>Understanding Programming Languages</td>
<td>10</td>
<td>Programming and Algorithms</td>
</tr>
<tr>
<td>COMP36411</td>
<td>Compilers</td>
<td>10</td>
<td>Computer Languages</td>
</tr>
<tr>
<td>COMP37111</td>
<td>Advanced Computer Graphics</td>
<td>10</td>
<td>Visual Computing</td>
</tr>
<tr>
<td>COMP38411</td>
<td>Cryptography and Network Security</td>
<td>10</td>
<td>Mobile Computing and Networks</td>
</tr>
</tbody>
</table>

### Level 3 - option pool 2

From this option pool choose a maximum of 20 credits and a minimum of 10 credits.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
<th>Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP31212</td>
<td>Concurrency and Process Algebra</td>
<td>10</td>
<td>Rigorous Development</td>
</tr>
<tr>
<td>COMP33512</td>
<td>User Experience</td>
<td>10</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>COMP33812</td>
<td>Software Evolution</td>
<td>10</td>
<td>Agile Methods</td>
</tr>
<tr>
<td>COMP36212</td>
<td>Advanced Algorithms II</td>
<td>10</td>
<td>Programming and Algorithms</td>
</tr>
<tr>
<td>COMP36512</td>
<td>Compilers</td>
<td>10</td>
<td>Computer Languages</td>
</tr>
<tr>
<td>COMP37212</td>
<td>Computer Vision</td>
<td>10</td>
<td>Visual Computing</td>
</tr>
</tbody>
</table>
Level 3 - option pool 3
From this option pool choose a maximum of 20 credits and a minimum of 0 credits.

<table>
<thead>
<tr>
<th>Code</th>
<th>Title</th>
<th>Credits</th>
<th>Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMP3620</td>
<td>First Year Team Project</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3010</td>
<td>Mathematical Techniques for Computer Science</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3011</td>
<td>Fundamentals of Computer Architecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3012</td>
<td>Management in Enterprises for Computer Scientists</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3013</td>
<td>Fundamentals of Artificial Intelligence</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3014</td>
<td>Fundamentals of Database Systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3015</td>
<td>Logic and Modelling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3016</td>
<td>Processed Microarchitecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3017</td>
<td>Microcontrollers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3018</td>
<td>Fundamentals of Databases</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3019</td>
<td>Software Engineering</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3020</td>
<td>Machine Learning and Optimisation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3021</td>
<td>Object-Oriented Programming with Java 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3022</td>
<td>Object-Oriented Programming with Java 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3023</td>
<td>Fundamentals of Distributed Systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3024</td>
<td>Symbolic AI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3025</td>
<td>Operating Systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3026</td>
<td>System Architecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3027</td>
<td>Algorithms and Image Processing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3028</td>
<td>Computer Graphics and Image Processing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3029</td>
<td>Distributed Computing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3030</td>
<td>Computer Networks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3031</td>
<td>Mobile Systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3032</td>
<td>Software Evolution</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3033</td>
<td>Natural Language Systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3034</td>
<td>Chip Multiprocessors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3035</td>
<td>Advanced Algorithms I</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3036</td>
<td>Advanced Algorithms II</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3037</td>
<td>Understanding Programming Languages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3038</td>
<td>Computer Vision</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3039</td>
<td>Advanced Computer Graphics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3040</td>
<td>Cryptography and Network Security</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMP3041</td>
<td>Digital Wireless Communication and Networks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HSTM3020</td>
<td>The Information Age</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCEL3001</td>
<td>Enterprise Management for Computer Scientists</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCEL3002</td>
<td>Managing Finance in Enterprises for Computer Scientists</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UCIL2001</td>
<td>Leadership in Action</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UCIL2002</td>
<td>Leadership in Action</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Mechanisms for programme revision

Course units are reviewed annually by the Undergraduate Committee, as part of the Annual Review process, taking into account the results and comments from Course Unit Evaluation Questionnaires. Input is also received from the Teaching Assessment Panel, which has a responsibility for monitoring teaching quality in the School.

Programmes have been reviewed regularly by groups created specifically for this purpose; the last major review resulted in a new programme portfolio design which started in the first year in 2008-9. The responsibility for leadership of programme review is now in the hands of the Director of Teaching Strategy (currently Dr Steve Pettifer) who chairs a School Teaching Strategy Committee.