Computer Science (Human Computer Interaction) BSc (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>I140</td>
<td>BSc</td>
<td>Computer Science (Human Computer Interaction) BSc (Hons)</td>
<td>3 years</td>
<td>FT</td>
</tr>
</tbody>
</table>

**Schools**
- Computer Science

**Faculty**
- Engineering and Physical Sciences

**Awarding Institution**
- University of Manchester

**Programme Accreditation**
- BCS, IET

**Relevant QAA benchmark(s)**
- Computing
Aims and intended learning outcomes

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>
<tr>
<td>Code</td>
<td>Title</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<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.