Programme Specification – Computer Science

Master of Computer Science (4 Years), BA Computer Science (3 Years)

1. Awarding institution/body	**University of Oxford**
2. Teaching institution	**University of Oxford**
3. Programme accredited by	**n/a**
4. Final award	**M.Comp.Sci. or B.A.Comp.Sci**
5. Programme	**Computer Science**
6. UCAS code	**G400**
7. Relevant subject benchmark statement	**Computing, published 2007**
8. Date of programme specification	**September 2009**

9. Educational aims of the programme

- To provide a course of the highest academic quality in Computer Science in a challenging and supportive learning environment that attracts the very best students from the UK and elsewhere.
- To provide students with a broad, balanced knowledge of core areas and advanced topics in Computer Science, as defined by the QAA Benchmark Statement issued in 2007.
- To develop in students the ability to evaluate primary evidence critically and the conceptual understanding to marshal and present arguments and solutions based on primary data, relevant theory and the application of sound reasoning.
- To develop transferable skills relating to problem solving and spoken and written communication.
- To bring students to a position on graduation where they can choose confidently from a wide range of careers, both within the Information Technology sector and outside it.

10. Programme outcomes

A. Students will develop a knowledge and understanding of:

- the general theoretical and practical principles of Computer Science.
- a broad range of topics in theoretical Computer Science and the architecture and implementation of information systems.
- relevant mathematical theories and techniques and their application to practical design problems.
- methods of software development.

The course is in line with the criteria set out in the QAA benchmark statement for Computing, 2007. That benchmark statement recognizes the need for diversity of provision in Computing, and the Oxford course remains firmly established at the theoretical end of the spectrum of degree courses. Topics from the body of knowledge outlined in the benchmark statement are covered as follows in the course:

Core material

Architecture (part), Concurrency and Parallelism (part), Data Structures and Algorithms, Programming Fundamentals, Theoretical Computing (part).

Options

Architecture (part), Comparative Programming Languages, Computer Networks, Compilers and Syntax Directed Tools, Computational Science (part), Concurrency and Parallelism (part), Distributed Computer Systems, Developing Technologies (part), Graphics and Sound (part), Operating Systems.

Advanced options

Artificial Intelligence, Computer Vision and Image Processing, Databases, Intelligent Information Systems Technologies (part), Natural Language Computing, Security and Privacy, Theoretical Computing (part), Information Retrieval (part).

Not covered

Computer Based Systems, Computer Communications, Computer Hardware Engineering, Document Processing, E-Business, empirical Approaches, Games Computing, Human-Computer Interaction, Information Systems, Management Issues, Middleware Professionalism, Multimedia, Simulation and Modelling, Software Engineering, Systems Analysis and Design, Web-based Computing.

There is no expectation on the part of QAA that any degree course will cover all, or even most, of the topics listed in the benchmark statement; this list shows that the proposed structure covers all of the more theoretical parts of the body of knowledge. The omissions are principally in the areas of electronic design and software practice.

Related teaching/learning methods and strategies

- *Lectures* are the principal mode in which the course content is delivered. Lecturers will generally use lectures that outline the areas of knowledge they wish students to become familiar with, and to emphasize particularly important concepts.
- *Private study* using books and problem sheets allows students to consolidate their knowledge of the course content, to check their understanding, and to explore topics beyond the syllabus. Substantial periods of private study during both term and vacations are necessary and expected.
- *Tutorials* provide a flexible form of student-centred small group teaching. Tutorials are typically based on sets of problems that help students to develop their understanding, and to draw out deeper aspects of the material covered in lectures. The small size of the group (one or two students) and the personal relationship that develops between tutor and pupils allows the tutorial to be closely matched to the needs of the student.

- *Advanced classes* (6-12 students) on some parts of the course content allow students access to expert specialists in particular areas.
- *Laboratory classes* allow students to examine more closely the practical applications of the material covered in lectures.
- *Independent project work* under the supervision of a member of staff allows students to gain experience in scientific investigation and in the development of more substantial software and hardware systems.

Assessment

Public examinations at university level are taken at the end of each year of the course.

Honour Moderations in Computer Science is a classified examination taken at the end of the first year, and consists of five papers; four are each of 3 hours' duration and the fifth is of 2 hours' duration. Students must either pass this exam, or pass a Preliminary Examination in September, in order to proceed into the second year of the course.

At the end of the second year, students take Part A of the Final Examination, which consists of one 3-hour paper and six 1.5-hour papers.

Following the Part A examination, students continue either towards a 3-year BA in Computer Science, or towards a 4-year Master of Computer Science degree.

In the third year, students on both the 3-year and the 4-year degree take Part B of the Final Examination, which consists of six 1.5-hour papers and a project.

For those on the 3-year BA course, degree classification is based on their performance in Parts A and B together.

In order to proceed to Part C, a student working towards the 4-year M.Comp.Sci. degree must achieve second class honours standard in Parts A and B together.

In the fourth year, students take Part C of the Final Examination, which consists of four 2-hour papers or mini-projects, and a project.

For those on the 4-year M.Comp.Sci. course, degree classification is based on their performance in Parts A and B together, and on their performance in Part C.

For each examination, students' participation and performance in practical work is taken into account.

Formative assessment is carried out continually during the whole of the course through tutorials and classes, termly college examinations, and laboratory classes.

B. Skills and other attributes

Students will have the opportunity to develop the following skills during the course:

I. Intellectual skills

1. The ability to demonstrate knowledge of key concepts and topics in Computer Science, both

explicitly and implicitly in the solution of problems.

2. The ability to understand and analyse problems, formulating them in terms of appropriate theoretical frameworks, in order to facilitate their solution.

3. An understanding of how Computer Science concepts and theories may be applied to the solution of problems including, where appropriate, an understanding that this may give only a partial solution.

4. The ability to select and apply appropriate techniques and processes.

5. The ability to construct and develop logical arguments, with clear identification of assumptions and conclusions.

6. The ability to present arguments and conclusions with clarity and accuracy.

Teaching/learning methods and strategies

These skills are acquired through lectures, classes, tutorials, independent study and project work.

Assessment:

These intellectual skills are assessed summatively in the examinations at the end of each year of the programme, and formatively in weekly classes and tutorials and in termly college examinations.

II. Practical skills

1. The ability to design and implement computer programs to solve specified problems.

2. The ability to consider, develop, assess and criticise the architecture of larger computer-based systems on a rigorous basis.

3. The ability to formulate precise specifications of computer-based systems.

Teaching/learning methods and strategies

These practical skills are developed throughout the course in work done for tutorials and classes, but especially in the supervised practical work that is attached to most lecture courses, and in the extended, supervised project that is carried out in the third year and the fourth year.

Assessment

The project counts for one third of the marks in the third year examination, and one third of the marks in the fourth year examination. Formative assessment of these skills takes place through tutorials, classes and practical classes, and through project supervision by a member of academic staff.

III. Transferable skills	*Teaching/learning methods and strategies*
1. The ability to study and learn independently	A learning process that requires students to assimilate material from several sources, including lectures, tutorials, practical classes, books, and online sources, largely self-guided.
2. The ability to analyse and solve problems and to reason logically and creatively.	Weekly problem sheets with tutorial or class support, often requiring significant development of ideas beyond material found in lectures or books.
3. Effective verbal and written communication and presentation.	Weekly tutorial and class assignments, with discussion and defence of written work in tutorials and presentation of solutions in classes.
4. Independent time management.	Requirement to produce substantial amounts of written work to strict tutorial and class deadlines. Need to balance academic and non-academic activities.
5. Teamwork.	Tutorial partners are encouraged to explore ideas together, thereby enhancing each other's learning.
6. The ability to think critically about problems and their solutions, and to defend an intellectual position.	Discussion and criticism in tutorials.

Assessment: The tutorial system provides continual formative assessment of all aspects of students' intellectual development.

11. Programme Structures and Features

The programme is offered as a four-year course leading to the degree of Master of Computer Science or, for those choosing to leave after 3 years, a BA in Computer Science.

Learning: Year 1

Subjects

- Functional Programming
- Design & Analysis of Algorithms
- Imperative Programming I
- Imperative Programming II
- Digital Hardware
- Discrete Mathematics
- Logic and Proof
- Linear Algebra
- Calculus
- Probability

Assessment

Five written papers (four of 3 hours each and one of 2 hours), together with assessed practical work.

Year 2

Subjects
Core courses (37.5%):
- Object-Oriented Programming
- Concurrency
- Models of Computation

Options (62.5%) including:
- Formal Program Design
- Numerical Analysis
- Compilers
- Principles of Programming Languages
- Advanced Data Structures and Algorithms
- Computer Graphics
- Computer Architecture
- Networks and Operating Systems
- Concurrent Programming

Assessment

Written papers the equivalent of three 1.5 hours each on core courses, plus five 1.5-hour papers on options, together with assessed practical work.

Year 3
Subjects
Options (66.7%) including some or all of:
- Intelligent Systems
- Integer Programming
- Databases
- Computational Complexity
- Geometric Modelling
- Numerical Solution of Differential Equations
- Lambda Calculus and Types
- Computer Security
- Logic of Multi-Agent Information Flow

And second year options (max 2).

Project work (33.3%)

Assessment

The equivalent of three written papers of 3 hours each, plus a project report and assessed practical work.

Year 4

Subjects

Options (66.7%) including some or all of:

- Categories, Proofs & Processes
- Computer Animation
- Game Semantics
- Information Retrieval
- Computational Linguistics
- Automata, Logic and Games
- Program Analysis
- Randomised Algorithms
- Theory of Data & Knowledge Bases
- Computer-Aided Formal Verification
- Software Verification
- Database System Implementation
- Probabilistic Model Checking

Project work (33.3%)

Assessment

A 2-hour written paper or take-home assignment on each of four courses, plus a project report and assessed practical work.

12. Support for Students and their learning

College Support. All students have a college tutor who oversees their academic progress. The college tutor arranges college teaching (tutorials and classes), advises on general academic matters such as the choice of options, oversees library provision of relevant texts in the college, arranges college examinations, and provides careers advice and references.

Students may also turn for help and advice to other officers of their college, including the Senior Tutor or Tutor for Undergraduates, the College Dean or Junior Dean, the College Chaplain, College Counsellor, College Doctor or Nurse, or the Head of the College. All students have access to college and university hardship funds and travel funds.

Role of college teaching. Undergraduates have tutorials and classes in their colleges during the first and part of the second year of the programme, typically two paired tutorials or one tutorial and one class per week. In the third and fourth year, undergraduates are supported by specialist inter-college classes, which typically contain 6—10 students, and are given by a class tutor assisted by a marker. For both tutorials and classes, students submit their written answers to the assigned problems before the tutorial or class for marking. The advantage of the class system is that undergraduates can receive specialist tuition on the more advanced topics that are studied in later parts of the programme. A record of attendance and achievement is kept for inter-college classes, and made available to tutors via a database, together with a report at the end of term. Tutors are alerted if poor attendance or achievement gives cause for concern.

Library provision. The University’s libraries provide all students with excellent resources. College

libraries and the University's lending library for science undergraduates provide students with all primary course texts (often with one copy per student), and provide a wealth of supplementary reading. Other written resources include excellent local bookshops, printed lecture notes and supporting material distributed via the World Wide Web, several text books written by members of staff, and past examination papers and examiners' reports that are accessible online.

Computing facilities. The Computing Laboratory maintains a network of 83 PCs for undergraduate practical and project work. The teaching machines are provided with a comprehensive collection of general-purpose software, together with practical materials that have been specifically developed to support learning in the degree programme. Examples of these materials include simulation studies of systems performance (Operating Systems), a range of interpreters and compilers for example languages (Compilers and Programming Languages), simulations of computer arithmetic and processor architecture (Digital Hardware and Computer Architecture), Concurrent Programming Libraries (concurrent programming) and application programming projects (Procedural Programming and Object-Oriented Programming). The Laboratory provides demonstrating staff who assist undergraduate learning during practical classes.

In addition, the University is rich in computing facilities of various kinds, and operates a high-speed data network that links all colleges and departments. Colleges provide computing facilities on the network that are easily accessible to students, and the majority of undergraduate rooms are now linked to this network, with most Computer Science undergraduates having their own PCs. The Laboratory's workstations are remotely accessible to students over the network.

13. Criteria for Admission

Applications are made to colleges of the University, not to the faculty/department, in the case of undergraduates.

Grades required are usually AAA at A-level, including A in Mathematics, or other equivalent international qualifications.

Offers are made on the basis of students' academic record, the recommendations of their teachers, and their performance in a written test and in interviews held in colleges in December.

Applicants are required before interview to take a written test of basic manipulative ability in Mathematics. This test is based on that part of the common core of Mathematics A-levels that will have been covered by applicants by this time, and includes questions designed to assess Computer Science ability.

The purpose of the interviews is to determine those students, from an excellent cadre of applicants, who might best benefit from the intensive, tutorial-based learning methods employed in the University.

14. Methods for evaluating and improving the quality and standards of learning

Student feedback on lectures and inter-college classes is encouraged by the distribution and collection of anonymous multiple-choice questionnaires and comment forms towards the end of each course. Responses to these questionnaires and comments are collated each term, reported to the teaching staff concerned, and considered by the Course Review Committee of the Computing Laboratory for action to be taken where appropriate.

Undergraduates' representatives attend meetings of the Faculty of Computer Science which oversees the programme, and student concerns are also discussed at termly meetings of the Joint Consultative Committee for Undergraduates.

Students' comments on tutorial provision in colleges are commonly sought by the Senior Tutor of each college, and are reviewed and acted upon in ways that vary from college to college.

The Teaching Committee of the Computing Laboratory is responsible for day-to-day running of the programme, oversees minor changes to the syllabus, and publishes annually a course handbook giving detailed syllabuses, synopses and reading lists for each lecture course.

Responsibility for the programme is vested in the Faculty of Computer Science, to which the Teaching Committee reports, and which is part of the Mathematical, Physical & Life Sciences Division. The divisional board has formal responsibility for the maintenance of educational quality and standards in its broad subject area, and exercises its responsibility through the divisional academic committee, and in particular gives scrutiny to new course proposals and proposed course revisions, to reports of examiners, and to more general questions of academic policy.

The divisional board is also responsible for academic appointments and for the arrangements (including mentoring, appraisal, and reviews of performance) for the support of newly appointed lecturers and for monitoring their teaching competence.

15. Regulation of assessment

Final Examination

The final examination is conducted in two parts for the three-year degree, and in three parts for the four-year degree, by a Board of Examiners consisting of five internal examiners and one external examiner. Examiners are nominated by the Computing Laboratory to make up the Examination Board that examines this degree and (in collaboration with the Mathematics examiners) the joint degree in Mathematics and Computer Science, subject to approval by the Vice-Chancellor and Proctors on behalf of the University. Examiners normally serve for a period of three years.

The Board of Examiners, under its elected Chair, is responsible for setting all papers, and marking the scripts of the examinees. They may appoint Assessors to assist in the setting and marking of the more specialist papers. Normally, the assessor for a subject will be the lecturer who has taught the topic in question. After scripts have been marked, the Board of Examiners meets to classify the students.

The examiners are guided by a set of Examination Conventions that are drawn up and reviewed by the Teaching Committee of the Computing Laboratory, endorsed by the Division of Mathematical, Physical & Life Sciences, and published on the Web. Oversight of all university examinations is carried out by the Junior Proctor, who considers all complaints and applications for special consideration, for example on medical grounds.

External Examiners are appointed in order:

1. To verify that standards are appropriate to the award, in part by comparison with the standards of comparable institutions, and to ensure that the assessment procedures and the regulations governing them are fair and otherwise appropriate.

2. To ensure that the conduct of the examination and the determination of awards has been fairly conducted, and that individual student performance has been judged in accordance with the regulations and conventions of the Examining Board. This will entail signing the Class List as an endorsement that the processes of examination and classification have been fairly conducted.

External Examiners are expected to report to the Vice-Chancellor in each year in which they act. Their reports are expected to cover all the following points:

- the standards demonstrated by the student
- the extent to which standards are appropriate for the award
- the design, structure and marking of assessments
- the procedures for assessment and examinations
- whether or not external examiners have had sufficient access to, and the power to call upon, any material necessary to make the required judgments
- students' performance in relation to their peers in comparable courses
- the coherence of the policies and procedures relating to external examiners and their consonance with the explicit roles required of them
- the basis and rationale for any comparisons made
- the strengths and weaknesses of the students as a cohort
- the quality of teaching and learning which may be indicated by student performance

The report is addressed to the Vice-Chancellor, and will be considered by the relevant divisional board, the faculty/department and by the University's Educational Policy and Standards Committee.

Where an external examiner's report contains particular suggestions or criticisms, it is the responsibility of the faculty/department to ensure that full consideration is given to these, to institute further discussion or action, and to inform the external examiner within a reasonable time of what is done.

16. Indicators of quality and standards

- The reports of External Examiners regularly address issues of quality and standards.

- The annual returns on first destinations of graduates demonstrate the success of graduates from Computer Science in the job market and in obtaining admission to postgraduate study.

- The External Advisory Panel of the Computing Laboratory includes representatives of a number of potential employers of graduates from the programme. It meets annually with officers of the Laboratory's Teaching Committee, and is consulted and kept informed about the content of the course.

- The Teaching Committee of the Computing Laboratory routinely monitors the indicators listed above, together with statistics on applications and acceptances for entry into the programme, statistics on the success of students in examinations, and feedback from current students.

Contact for queries:
Academic Administrator
Oxford University Computing Laboratory
Wolfson Building
Parks Road,
Oxford OX1 3QD