Computing Science FAQs

February 2021

1. What are the compulsory modules/classes I will be taking in first year?

At the time of writing, for our single Honours degree programmes there are 4 compulsory modules as shown below, although these can change slightly from year-to-year. Joint Honours students must take a prescribed subset of these.

<table>
<thead>
<tr>
<th>First Term</th>
<th>Second Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Programming for Sciences and Engineering</td>
<td>• Object-oriented Programming</td>
</tr>
<tr>
<td>• Modelling and Problem-solving for Computing</td>
<td>• Computer Architecture</td>
</tr>
</tbody>
</table>

These courses are designed to provide a solid foundation in core areas of computing (programming, reasoning, computer systems). We do not assume previous background in computing or programming before the first term courses.

2. Can I take optional modules/classes? How many? Are there any restrictions to what I can take?

In each term you will normally take a total of 4 modules, so you can choose optional courses to fill up any spaces not occupied by compulsory modules. Usually, this is 1 or 2 per semester. There are very few restrictions, provided the modules fit with the timetable of your compulsory modules. This can include other sciences, business, languages, music etc. When you are choosing modules, you will have access to a timetable checker tool that lets you check there are no clashes before selecting your modules. For subjects for which we offer joint degrees, including Maths, Physics, certain Business topics, and Music, timetables are engineered in advance so that clashes are minimized.

3. What will my timetable in first year look like? How much contact time will I have with lecturers?

The standard format for a computing module consists of 11 weeks of lectures and 10 weeks of small group sessions in practical labs or tutorials. During each week, you’ll typically have 2 hours of lectures and 2 hours of practical sessions per module, i.e., around 42 hours of contact time per module across the term. In addition, we pride ourselves on being an open friendly department, with staff operating an ‘open-door’ policy to help address any concerns a student may have, leading to additional contact time as needed.
4. What are the typical first year class sizes like?

There are around 60-80 students typically enrolled in single and joint Computing programmes. However, other students take our courses as electives, and the largest class is Programming for Sciences and Engineering with around 200 students. However, you will also take part in small group exercises (labs, tutorials). Here, with face-to-face teaching you will typically be in a group of no more than 20 students; online classes are sometimes slightly larger.

5. Are there work placements available and are they built into the degree?

A number of our students undertake a year-long placement in industry, usually as part of the MSci Computing Science with Industrial Placement programme. It is very often possible to transfer to the MSci from BSc Computing Science, and also in the opposite direction. The placement is typically between the 3rd and 4th years of study, but is sometimes also done after the 4th year. Once they graduate, students often find employment where they undertook the work placement.

6. Will there be field trips in my classes?

Computing is predominantly a classroom and computer-lab based subject. However, we do run regular events, giving you an opportunity to network with potential employers.

7. Will there be any group work/group projects?

Several of our modules have a group work component. In the 3rd year Software Engineering module you will undertake a year-long group project to build a large, industrially-relevant piece of software, preparing you to work as part of a software development team.

8. How are my modules/classes assessed? Exam, essays, in-class work?

Assessments differ between modules. Most modules have a significant continuous assessment component, such as programming exercises, projects, reports, essays, presentations, or in-class tests. Some modules use final exams to test more theoretical learning outcomes.

9. Is there an opportunity to study languages alongside my degree?

In the 1st and 2nd year, students take slightly under 50% non-computing subjects, providing significant scope for language study or investigating other topics. In the 3rd and 4th years, students work mainly on computing modules, but it can sometimes be possible for a small number of non-computing modules to be taken.

10. Should I be preparing for first year by reading anything specific? Do you have any suggestions?

We expect no subject-specific knowledge for students entering first year. In particular, we do not assume that students have previous programming experience. However, practicing computational thinking has been found to help students achieve on the programme, and certain books are engaging and help in this regard. On the practical side, the main programming language currently used in the first year is Python and a good introduction for those with no previous experience is The Python Workbook by Ben Stephenson. The book Algorithmics: The Spirit of Computing by David Harel gives an introduction to key concepts. General reasoning skills can be sharpened with The Lady or the Tiger & Other Logic puzzles by Raymond M Smullyan.
11. Are there any societies which would be good to join? E.g. societies that are for those studying my degree.

There is a very active computing society run by students, which offers programming competitions, hackathons, talks and support.

12. Can I study abroad with this degree? Where can I go?

Students have travelled and studied overseas as part of exchange programmes. Our students have often gone to study across Europe, in the USA, and the Far East in the recent past.

13. How much time will I spend in labs each week?

You will typically spend 2 hours per module per week in tutorials or computer labs. Subjects with emphasis on practical skills such as programming or databases will use mainly labs rather than tutorials. However, you will spend significantly more time than this on independent work on practical topics. This can usually be done either on personal machines or in computer classrooms.