Programming Teaching Working Group – First Report
P Anderson, H Leather, M Rovatsos, D Sannella, A Simpson, P Stevens, P Wadler
Informatics Teaching Committee, 21st May 2014

Summary
The working group, formed by Teaching Committee to investigate options for improving our teaching provision of programming skills has recently held its first meeting. It became clear that the overall problem is very broad, complex, and multi-faceted, and that we currently lack the information necessary to make any informed recommendations for substantial changes. However, there are low-impact steps that can be taken over the current and next academic year to improve our understanding of the problems surrounding programming skills.

Observations
Discussion so far has focussed on two particular questions:
(1) Is our provision of programming teaching in years 1 and 2 adequate for what is expected by subsequent courses in higher years?
(2) Do our graduates have the programming skills needed by employers, in particular for the high-quality job opportunities we might want them to enjoy?

Regarding (1), there is consensus that
- students are entering our degrees with very different programming backgrounds (often with no background at all); we do not want to require all students to have previous programming experience; pre-sessional instruction activities would create an expectation that this is needed, and are therefore not considered an adequate solution;
- it would be possible to offer two versions of the Inf1-OP course to address this, though it's not clear whether a more basic or more advanced version should be compulsory; these would likely have to run in parallel in semester 2, as teaching OP in parallel with Inf1-FP would likely overwhelm many students; switching the ordering of how the two languages are taught is undesirable, as it is preferrable to begin with a language almost no students have experience with;
- there is a suspicion that since there is little coverage of programming skills in year 2, it is likely that weaker students enter year 3 without the skills expected of them in other courses; students and staff have specifically pointed out that this is the case in the “systems” area, where a better knowledge of C/C++ would be helpful, but we also see it in the limited contribution to actual systems development in SDP groups, and in the fact that even though Large Practicals are (according to stronger students) not very demanding, the spread of marks in those is relatively normal; introducing more programming-specific courses in year 2 would, however, require dropping some other elements of the year 2 curriculum.

Regarding (2), while there is some evidence that it is possible for students to graduate without a proven ability to develop significant pieces of software from scratch independently, and that some students perform poorly on fairly simple programming tasks, there is no conclusive evidence that this is due to a lack of opportunities to develop programming skills, or that, in fact, it is keeping students from obtaining the kind of employment they are aiming at (or that their other skills would be regarded as a sufficient qualification for).
Recommendations
The group has identified the following measures that can be taken before making any major steps to adapt the curriculum (should those be necessary) for 2015/16:

1. Data gathering: We need more input from students, staff, and employers to get a better understanding of the problem (or to determine if there is a problem at all). As part of our efforts to re-establish the Industrial Advisory Board, we should attempt to get more input on the programming skills of our graduates and interns who apply to them. This could be broadened to enquiring with companies that give career presentations, host undergraduate interns, or award prizes to our students. Staff and student input could be elicited through questionnaires or an analysis of SSCC meeting minutes, though this requires careful design to distinguish problems with other aspects of coursework as opposed to programming skills. We should also engage with alumni to get their views on the issue.

2. Self-assessment and remedial labs: We suggest to gather more information on students’ programming abilities by running a self-assessment programming exam as part of the induction to the Large Practical courses (ideally to be held for all Year 3 students during induction week). Students who perform poorly in this exam should attend a small number of remedial lab sessions which we’re proposing to run at the start of year 3. The labs will be held during the first few weeks of term (maybe even in a very concentrated way over a few days) to avoid overlap with other coursework. Timetabling and resourcing for these labs needs to be discussed, though plenty of material is available and Hugh Leather has indicated that he would be interested in running them. The opportunity of using peer-assisted learning to staff these labs should be explored. Making these labs available also to stronger students is an option that could be discussed, though it would obviously increase the number of demonstrators required. Experience with this scheme will help determine whether something similar should be offered in Year 2, either on the basis of self-assessment or on the basis of performance in the Year 1 programming courses. If the standard achieved at Year 2 is adequate, but for Year 3 it is not, then we would need to look into possible changes to the Year 2 curriculum.

3. Exploring the option of an additional OP course in Year 1: Perdita Stevens will introduce flexible streamed tutorials in Inf1-OP in 2014/15. Experience with these labs will allow us to better assess the potential value of expanding such streaming to two independent courses (e.g. with Inf1-OP being taught at a faster pace, and students with no background being required to take a 20-point version of the course provides additional lectures, labs, and tutorials and dedicates more time to basic material). The objective of this would be to increase the standard of programming skills at the end of Year 1, assuming that the data we obtain through items 1. and 2. suggest students are often ill-prepared for Year 2 or Year 3.