UG3 Large Practicals
Change to 20pts course

Stephen Gilmore & Paul Patras & Alan Smaill

March 8, 2016

1 Introduction

This is a proposal to align the UG3 Large Practicals with the new "Workload and Assessment in Taught Courses" policy in the School. These practicals are currently worth 10 points; the proposal is to change this to a 20 points while also giving the students greater freedom to shape the practical to their own interests.

As these three practicals fulfil similar roles within the UG3 syllabus (being required for Informatics UG degrees, and mutually exclusive), we present a unified framework here. Aspects of the individual courses are addressed in the appendices.

2 Coursework

The UG3 Large Practical courses are assessed entirely by coursework timetabled during Semester 1, and thus uses the full 14 weeks of Semester 1 for the coursework. The coursework for the revised courses will be in three parts:

Part 1: A proposal document, putting forward suggestions with respect to aspects of the practical to be undertaken by the student. This part is *formative assessment, for feedback only.* The deadline is 16:00 on Friday of Week 4, Semester 1.

Part 2: An assessed submission, covering a substantial part of the work associated with the practical. This part is *summative assessment, worth 50% of the assessment overall.* The deadline is 16:00 on Friday of Week 8, Semester 1.

Part 3: A final assessed submission, suitable for overall judgement of work on the practical This part is *summative assessment, worth 50% of the assessment overall.* The deadline is 16:00 on Friday of Week 14, Semester 1.

All coursework is submitted electronically, using the `submit` command which is available on DiCE.

3 Workload

This course re-structuring seeks to give the students more scope for independent learning and self-guided investigation within the framework of the practical.

Weeks 1–4: These weeks are for independent learning about technologies and concepts which might be used within the practical, and fitting these together into a high-level plan for the practical. Work done in this period of the semester contributes to the not-for-credit Proposal document, and is exploratory learning in nature.

Weeks 5–6: These weeks are for further independent study and planning, considering technologies to be used in the practical.

Weeks 7–8: By this point, feedback has been received on Part 1, so these weeks are for incorporating that feedback into the week 8 submission.

Weeks 8–9: These weeks are for further independent study and planning.

Weeks 10–11: By this point, the assessment of Part 2 has been received, so that the associated feedback can be used to guide the later stages of the practical.

Weeks 12–14: These last weeks are given over to concentrated work on the goals of the final assessed coursework.

Other Informatics courses will have finished with their assessed work, allowing students to focus on the implementation work of the Large Practical.

This re-structuring of the course should help to reduce the often-reported feeling of over-assessment, by clarifying that there are independent study weeks within the semester where assessed work is not being carried out.

4 Points value

The points value of the course is largely determined by the difficulty of the practical exercise which is set. The work which is set for the academic year 2016/2017 will be more difficult than the work set in previous years, but there is no intention to make it *twice* as difficult, because of the revision of (formative and summative) submissions associated with the course.

A AILP DRPS entry with changes in bold font

Undergraduate Course: Artificial Intelligence Large Practical (INFR09018)

Course Outline			
School	School of Informatics	*College*	College of Science and Engineering
Credit level (Normal year taken)	SCQF Level 9 (Year 3 Undergraduate)	*Availability*	Not available to visiting students
SCQF Credits	**20**	*ECTS Credits*	**10**
Summary	Students will gain experience in how to: • Designing a well structured system • Implementing such a system • Designing and running experiments • Reporting and analysing results		
Course description	**This project gives students experience in developing a non-trivial computational solution to an AI problem domain. In particular the student gains practical experience of the following:** • Gentle introduction to the issues and requirements of the more demanding fourth-year project. • Experience of reading published papers and identifying their essential content. • **Experience of describing a problem area and a proposed computational exploration of a solution.** • Exercise of reporting on modest pieces of scientific work: students have to explain what they did, and why, and what conclusions they reached, and why, and they have to do this clearly and convincingly. • Experience of writing programs to investigate specific questions: students must write well-structured, well-documented programs because they too are acts of scientific communication. Relevant QAA Computing Curriculum Sections: Artificial Intelligence		

Entry Requirements (not applicable to Visiting Students)			
Pre-requisites		*Co-requisites*	
Prohibited Combinations	Students MUST NOT also be taking Computer Science Large Practical (INFR09040) OR AI Large Practical (INFR09018). Students MUST NOT also be taking Informatics Research Proposal (INFR11071)	*Other requirements*	This course is open to all undergraduate Informatics students including those on joint degrees. For external students where this course is not listed in your DPT, please seek special permission from the course organiser.

Course Delivery Information			
Academic year 2016/17, Not available to visiting students (SS1)		**Quota: None**	
Course Start	**Semester 1**		
Learning and Teaching activities	**Total Hours: 200 (Lecture Hours 8, Formative Assessment Hours 26, Summative Assessment Hours 52, Programme Level Learning and Teaching Hours 4, Directed Learning and Independent Learning Hours 110)**		
Assessment	**Written Exam 0%, Coursework 100%, Practical Exam 0%**		
Additional Information (Assessment)	**One large design, implementation and evaluation project, done in three parts. The first part consists of a proposal document outlining the problem area, and proposing a solution technique and design. The second part is an initial implementation, providing basic capabilities for a full system. The third part is covers experimentation, analysis and reporting of results.**		

Learning Outcomes

On completion of this course, the student will be able to:

1. Design and implement a complex system.
2. Consider alternative designs, both for internal properties, and as ways of tackling a given problem.
3. Read technical papers, and explain their relevance to the chosen approach.
4. Design and carry out appropriate experiments, and explain the methodology involved.
5. Write a scholarly report, suitably structured and with supporting evidence.

Additional Information	
Course URL	`http://course.inf.ed.ac.uk/ailp`

Contacts			
Course organiser	Dr Alan Smaill Tel: (0131 6)50 2710 Email: A.Smaill@ed.ac.uk	Course secretary	Miss Beth Muir Tel: (0131 6)51 7607 Email: beth.muir@ed.ac.uk

B CSLP DRPS entry with changes in bold font

Undergraduate Course: Computer Science Large Practical (INFR09039)

Course Outline			
School	School of Informatics	*College*	College of Science and Engineering
Credit level (Normal year taken)	SCQF Level 9 (Year 3 Undergraduate)	*Availability*	Not available to visiting students
SCQF Credits	**20**	*ECTS Credits*	**10**
Summary	The Computer Science Large Practical exposes students to the problems that arise with the design and implementation of large scale computer systems, and to methods of coping with such problems. • Design clearly and coherently structured systems • Choose the appropriate means of implementation • Discover and use relevant information • Schedule their work load • Present their work in a clear and concise way.		
Course description	This project gives students experience in developing a non-trivial software system and providing some analysis of its behaviour. In particular the student gains practical experience of: • Design: ensuring the requirement is well scoped and defined and constructing a design at a level of detail that allows some prediction of the behaviour of the system without committing to implementation detail. • Implementation: implementing and testing a working prototype of the design. • Analysis: providing some analysis of the characteristics of the design. Relevant QAA Computing Curriculum Sections: Computer Based Systems, Software Engineering, Systems Analysis and Design, Professionalism		

Entry Requirements (not applicable to Visiting Students)			
Pre-requisites		*Co-requisites*	
Prohibited Combinations	Students MUST NOT also be taking Computer Science Large Practical (INFR09040) OR AI Large Practical (INFR09018). Students MUST NOT also be taking Informatics Research Proposal (INFR11071)	*Other requirements*	This course is open to all undergraduate Informatics students including those on joint degrees. For external students where this course is not listed in your DPT, please seek special permission from the course organiser.

Course Delivery Information			
Academic year 2016/17, Not available to visiting students (SS1)		**Quota: None**	
Course Start	**Semester 1**		
Learning and Teaching activities	**Total Hours: 200 (Lecture Hours 8, Formative Assessment Hours 26, Summative Assessment Hours 52, Programme Level Learning and Teaching Hours 4, Directed Learning and Independent Learning Hours 110)**		
Assessment	**Written Exam 0%, Coursework 100%, Practical Exam 0%**		
Additional Information (Assessment)	**In the first part, an application of the base system is presented to the students. Students must refine the application specification, and propose a design. The second part consists of a partial implementation of the system that comprises a subset of the required functionality. In the third part, students fully implement their design, and submit both their implementation and a report that presents and analyses their specification, design, implementation and tests**		

Learning Outcomes

A student who has successfully completed this course should be able to:

1. **Plan to manage complex systems with competing requirements, read technical papers and extract relevant content.**
2. **Consider and compare possible structures for a design.**
3. **Plan to manage their time and resources in completing a large project.**
4. **Implement and debug a computer system of medium to large size.**
5. **Analyse the performance of a system, and write clear and concise documentation.**

Additional Information	
Course URL	`http://course.inf.ed.ac.uk/selp`

Contacts			
Course organiser	Dr Paul Patras Tel: (0131 6)50 4408 Email: ppareas@inf.ed.ac.uk	Course secretary	Miss Beth Muir Tel: (0131 6)51 7607 Email: beth.muir@ed.ac.uk

C SELP DRPS entry with changes in bold font

Undergraduate Course: Software Engineering Large Practical (INFR09039)

Course Outline			
School	School of Informatics	*College*	College of Science and Engineering
Credit level (Normal year taken)	SCQF Level 9 (Year 3 Undergraduate)	*Availability*	Not available to visiting students
SCQF Credits	**20**	*ECTS Credits*	**10**
Summary	The Software Engineering Large Practical exposes students to the problems that arise with the design and implementation of large scale software systems, and to methods of coping with such problems. Students will gain experience in how to: • Design clearly and coherently structured systems • Choose the appropriate means of implementation • Discover and use relevant information • Schedule their work load • Present their work in a clear and concise way.		
Course description	This project gives students experience in developing a non-trivial software system and providing some analysis of its behaviour. In particular the student gains practical experience of: • **Proposal: negotiating well-scoped and defined requirements for the practical.** • **Design: constructing a design at a level of detail that allows some prediction of the behaviour of the system without committing to implementation detail.** • Implementation: implementing and testing a working prototype of the design. Relevant QAA Computing Curriculum Sections: Computer Based Systems, Software Engineering, Systems Analysis and Design, Professionalism		

Entry Requirements (not applicable to Visiting Students)			
Pre-requisites		*Co-requisites*	
Prohibited Combinations	Students MUST NOT also be taking Computer Science Large Practical (INFR09040) OR AI Large Practical (INFR09018). Students MUST NOT also be taking Informatics Research Proposal (INFR11071)	*Other requirements*	This course is open to all undergraduate Informatics students including those on joint degrees. For external students where this course is not listed in your DPT, please seek special permission from the course organiser. **Students are assumed to have competence in design, debugging, documentation and programming in a language such as Java.**

Course Delivery Information			
Academic year 2016/17, Not available to visiting students (SS1)		**Quota: None**	
Course Start	**Semester 1**		
Learning and Teaching activities	**Total Hours: 200 (Lecture Hours 8, Formative Assessment Hours 26, Summative Assessment Hours 52, Programme Level Learning and Teaching Hours 4, Directed Learning and Independent Learning Hours 110)**		
Assessment	**Written Exam 0%, Coursework 100%, Practical Exam 0%**		
Additional Information (Assessment)	**One large design and implementation project, done in three parts. The first part consists of a proposal document specifying functional and non-functional requirements on the project. The second part is a design document, presenting the plan of the implementation work which will realise the design. The third part is the implementation. This should be a well-engineered implementation of the previously-supplied design.**		

Learning Outcomes

1. **Show awareness of current software engineering technologies and demonstrate the ability to select the appropriate technologies for a project.**
2. **Present a convincing proposal for a software development project.**
3. **Show awareness of the difference between design and implementation in software development.**
4. **Demonstrate proficiency with modern software development platforms and frameworks.**
5. **Exhibit the ability to manage a medium- to large-scale software development project from proposal to final implementation.**

Additional Information	
Course URL	`http://course.inf.ed.ac.uk/selp`

Contacts			
Course organiser	Prof Stephen Gilmore Tel: (0131 6)50 5189 Email: S.Gilmore@ed.ac.uk	Course secretary	Miss Beth Muir Tel: (0131 6)51 7607 Email: beth.muir@ed.ac.uk