UNIVERSITY OF WARWICK

Proposal Form for New or Revised Modules (MA1- version 4)

Approval information	
Approval Type	☒ New module ☐ Revised module ☐ Discontinue module
Date of Introduction/Change	01/10/2011
If new, does this module replace another? If so, enter module code and title:	NA
If revised/discontinued, please outline the rationale for the changes:	NA
Confirmation that affected departments have been consulted:	TBC

Module Summary	
1. Module Code (if known)	CS***
2. Module Title	Interdisciplinary Modelling through Constructivist Computing
3. Lead department:	Computer Science
4. Name of module leader	Meurig Beynon
5. Level	UG: ☐ Level 4 (Certificate) ☐ Level 5 (Intermediate) ☒ Level 6 (Honours) PG: ☐ Level 7 (Masters) ☐ Level 8 (Doctoral) See Guidance Notes for relationship to years of study
6. Credit value(s) (CATS)	15
7. Principal Module Aims	This module will bring together students from different disciplines to address core themes through teamwork based on computer-based modelling. The principal

Module Summary	
	objective is to study ways in which computing technology can support such interdisciplinary modelling, and to highlight the merits of constructivist computing in this role. Constructivist computing is based on the principles, tools and potential applications of Empirical Modelling (EM), a new approach to modelling and computing that has been developed by staff and students here in Computer Science at Warwick over many years. Students will be instructed in fundamental EM concepts and techniques; be shown how EM is informed by perspectives relating to computer science, philosophy and constructivist thinking; be introduced to the fundamental notion of construal as it applies to key application areas; and participate in a collaborative modelling study directed at topics aligned to their personal interests and disciplinary specialisms.
8. Contact Hours (summary)	72
9. Assessment methods (summary)	Written report of Lab work - 25% Model analysis & documentation - 25% Group presentation - 25% Individual oral examination - 25%

Module Context

10. Please list all departments involved in the teaching of this module. If taught by more than one department, please indicate percentage split.

Computer Science

11. Availability of module

Degree Code	Title	Study Year	C/OC/ A/B/C	Credits
Example	Accounting and Finance **International Summer School students**	1 **2 or 3**	C	12 **15**

12. Minimum number of registered students required for module to run

8

13. Pre- and Post-Requisite Modules

NA

Module Content and Teaching

14. Teaching and Learning Activities

Lectures	**10 (Week 1) + 10 (Week 2) + 2 external speakers**
Seminars	**5 seminars / demos on key models (weeks 2-3)**
Tutorials	**1 hour tutorial time for each student in total (20 minutes per week) with 3 students per tutorial session**
Laboratory sessions	**5 x 3 hours (Week 1) + 5 x 3 hours (Week 2)** **3 x 3 hours (Week 3)** **These will comprise practical sessions involving construction and analysis together with group work (Week 3)** **Practical workshops relating to core themes (3 hrs)** **Group presentation sessions (2 hours)**
Total contact hours	**72**
Module duration (weeks)	**3**
Other activity *(please describe): e.g. distance-learning, intensive weekend teaching etc.*	

15. Assessment Method (Standard)

Type of assessment	Length	% weighting

Module Content and Teaching		
Examinations	0 Hours	**0**
Assessed essays/coursework	Lab report / Model analysis + documentation - each equivalent to 2000 Words	**25% + 25%**
Other formal assessment	**Team presentation with individual contributions (2 hours) Individual oral exam (30 minutes)**	**25% + 25%**

16. Methods for providing feedback on assessment.

General feedback via module webpage/forum, individual feedback via tutorials. Detailed written feedback at the end of the module.

17. Outline Syllabus

The module will be delivered over three weeks, each week (comprising 5 days) representing a different phase of teaching and learning activity. In broad terms, Week 1 will motivate constructivist computing and introduce the basic principles, concepts and tools, Week 2 will be an in-depth study of papers and models relating to specific application areas for constructivist computing, and Week 3 will be devoted to group-based collaborative modelling focusing on one of three core themes that will be advertised in advance of the module.

The portfolio of core themes for the module proposed for Summer 2012 is:

A. Modelling the musical and dramatic production and appreciation of Mozart's opera The Magic Flute. In this context, model-building could address the structural, thematic and harmonic aspects of the musical score, the characteristics of the cast (e.g. vocal range and physical characteristics) both as given and as ideally required, the layout of scenes and cast members involved in them, the design of costume and scenery, the movement of singers as planned by the artistic director, characteristics of specific performances such as relate to choice of tempi and potential cuts, the libretto and narrative, and the logistics (e.g.) of timetabling rehearsals and organising scene changes during a performance. Such a modelling exercise would draw on skills from computing, music, drama, business, design and education.

B. Making a model of a complex medical condition, such as HIV/AIDS, from multiple perspectives, in such a way as to be useful to trainee doctors, nursing staff with less specialist medical training, epidemiologists, patients, partners, and people without relevant medical knowledge. In this context, a model might simulate the case history of an AIDS patient according to current understanding of the pathophysiology of AIDS, taking account of the stages by which the HIV

Module Content and Teaching

infection impacts on a patient, adjusting their susceptibility to other infections, and assessing the risk of vertical transmission. A complementary analysis of the observables, dependencies and agencies at work would address factors that might influence the spread of the illness, such as the physical appearence of a patient and impact of cultural beliefs and customs. Relevant specialisms in this context include computing, medicine, education, statistics and social and anthropological studies.

C. Developing a suitable environment and associated suite of models to showcase the potential for using contructivist computing in teaching and learning mathematics and theoretical computer science. Different perspectives to be represented in this modelling might relate to the role being adopted by the model-builder (e.g. whether interaction is in the mode of developer, teacher or learner); what aspects of the curriculum are being targeted (e.g. basic mathematics for computer science, algorithmics, relational database theory, boolean circuit theory); the educational level (from elementary through intermediate to advanced level in school to undergraduate level in Computer Science and Mathematics); the focus (e.g. recreational mathematics, applied mathematics, mathematical research). Relevant areas of expertise in this context would include mathematics, computer science and education with particular emphasis on logical, computing, puzzle and problem-solving skills.

Each model-building study will be broad, have many aspects, and draw on several disciplines. In Week 3, each core theme will be addressed by a team of five students who will work alongside a Warwick Computer Science student with prior experience of EM in relevant areas of application. Though the quality of the models developed for specific aspects may not compete with what could be created using special-purpose commercial tools, the modelling has a quite distinctive characters and produces models unlike traditional computer models. Particularly significant is the flexibility, openness to reinterpretation and extension, and the way in which different aspects of the modelling are integrated and can be reconfigured.

Week 1 will introduce the notion of constructivist computing, as supported by Empirical Modelling (EM). The key notion of a construal will be introduced and discussed in relation to modelling and programming. The key concepts of observables, dependencies and agents will be explained and illustrated with standard examples. Practical model-building will be complemented by paper-based techniques for analysing complex systems based on classifying observables according to how they are perceived and manipulated by agents. The nature of applications characteristic of constructivist computing will be explored with reference to topics such as learning technologies, personalised applications, models for medical use, model-building to support experimental science or mathematical research, business modelling and humanities computing. An appropriate philosophical orientation for constructive computing will be briefly discussed with reference to critiques of the traditional "rationalistic" account of

Module Content and Teaching

computing, Latour's proposals for rehabilitating the concept of construction in his paper "The Promises of Constructivism", and William James's radical empiricism.

In keeping with the spirit of constructivist computing, practical model building will be an essential strand of the module. The tools to be introduced will include the Cadence and EDEN interpreters and auxiliary environments such as the EM presentation environment (EMPE) and the dependency modelling tool (DMT).

Week 1 will involve ten lectures and ten laboratory sessions. Through personal consultations with students, we shall identify their particular strengths and interests and direct them towards different aspects of the core themes to be addressed in Week 3.

Week 2 will look in more detail at specific applications of EM relating to the chosen core themes, as represented by papers and model drawn from the EM archive. Each of the five days will be devoted to in-depth study of one or more existing models devoted to a specific theme (cf. the range of different kinds of observables, dependencies and agents represented in modelling strands associated with the core themes mentioned above, for which there are precedents in the EM archive in the form of construals of musical compositions, organs of the human body, timetabling models, mathematical algorithms and puzzles, visualisations and animations etc). Relevant key papers drawn from the EM archive will be studied in parallel. During this week, students will work on exercises in analysing and documenting models, typically making use of special-purpose tools such as the EMPE and the DMT.

In Week 2, there will be one or two introductory lectures and one or more sessions involving the analysis and demonstration of models relating to various aspects of the core themes. For the rest of the week, students will work in laboratory sessions alongside a small team of Warwick assistants (preferably undergraduates familiar with EM principles and tools who ideally would have themselves already worked on the modelling exercises and prototyped models related to the core themes).

Week 3 will be devoted to group work aimed at documenting and enhancing existing prototype models and conceiving and/or developing new models addressing the core themes. In the constructivist spirit, practical modelling activity will play a central role in this; even the conception of models will involve some experiments in model-building. An additional component of this activity that involves no conceptual shift within the EM framework might be enhancement of the modelling tools better to address the specific demands of a core theme.

The assessment for the module will be based on laboratory work in Week 1, the analysis and documentation exercise in Week 2, and the contribution to the group work in Week 3, as expressed through practical work and assessed through group presentations and individual oral examinations to be held at the end of Week 3.

Module Content and Teaching

To complement the above teaching programme, we shall organise one or two seminars from external speakers with specialist expertise relating to core themes, such as an expert in humanities computing or former EM doctoral students who have worked on (e.g.) collaborative group work using EM. We shall also engage other teaching assistants (e.g. from the Warwick Student Opera Group) to enact authentic activities relating to a core theme that help to illuminate the processes involved in the application (such as a "blocking rehearsal" for an ensemble from the Magic Flute, at which the precise movements and interactions of the singers are planned out in detail).

18. Illustrative Bibliography

Selected papers and models drawn from the EM publications and archived models at http://www.dcs.warwick.ac.uk/modelling
Fred Brooks, The Mythical Man Month Re-Visited, Addison-Wesley, 1995
Brian Cantwell-Smith, "The Foundations of Computing", In Scheutz, M.(ed) Computationalism: New Directions, MIT Press, p23.58, 2002
Chris Date and Hugh Darwen, Relational Database Writings (several books from 1985-1999), Addison-Wesley.
Paul Dourish, Where the Action is, MIT Press 2001
David Gooding, Experiment and the Making of Meaning: Human Agency in Scientific Observation and Experiment, Kluwer Academic, 1990
William James, Essays in Radical Empiricism, Bison Books 1996
Michael Jackson, "What can we expect of program verification?", IEEE Computer, 39(10):53{59, October 2006
David Harel and Rami Marelly, Come, Let's Play: Scenario-Based Programming Using LSCs and the Play-Engine, Springer-Verlag, 2003
William Kent, Data and Reality, 1st Books Library, 2000
Bruno Latour, "The Promises of Constructivism", In Ihde, D. (ed.) Chasing Technoscience: Matrix of Materiality, 2006
Willard McCarty, Humanities Computing, Palgrave-MacMillan, 2005
Drew McDermott, "A Critique of Pure Reason", Computer Intelligence 3 pp. 151-160 (1987) + subsequent responses in same journal.
Peter Naur, "Intuition in Software Development", TAPSOFT, Vol. 2, pages 60-79, 1985
Bonnie Nardi, A Small Matter of Programming, MIT Press, 1993.
Terry Winograd and Fernando Flores, Understanding Computers and Cognition: A New Foundation for Design, Addison-Wesley, 1986.

19. Learning outcomes

Successful completion of the module leads to the learning outcomes. The learning outcomes identify the knowledge, skills and attributes developed by the module.

Learning Outcomes should be presented in the format "By the end of the module students should be able to..." using the table at the end of the module approval form:

Resources

Resources
20. List any additional requirements and indicate the outcome of any discussions about these.

Approval	
21. Module leader's signature	
22. Date of approval	
23. Name of Approving Committee (include minute reference if applicable)	
24. Chair of Committee's signature	
25. Head of Department(s) Signature	

Examination Information		
A1. Name of examiner (if different from module leader)		
A2. Indicate all available methods of assessment in the table below		
% Examined	**% Assessed by other methods**	**Length of examination paper**
A3. Will this module be examined together with any other module (sectioned paper)? If so, please give details below.		
A4. How many papers will the module be examined by?	☐ 1 paper	☐ 2 papers
A5. When would you wish the exam take place (e.g. Jan, April, Summer)?		
A6. Is reading time required?	☐ Yes	☐ No
A7. Please specify any special exam timetable arrangements.		
A8. Stationery requirements		
No. of Answer books?		
Graph paper?		
Calculator?		
Any other special stationery requirements (e.g. Data books, tables etc)?		
A9. Type of examination paper		
Seen?	☐ Yes	☐ No
Open Book?	☐ Yes	☐ No
Restricted?	☐ Yes	☐ No

Examination Information	
If restricted, please provide a list of permitted texts:	

LEARNING OUTCOMES		
(By the end of the module the student should be able to....)	**Which teaching and learning methods enable students to achieve this learning outcome? (reference activities in section 15)**	**Which summative assessment method(s) will measure the achievement of this learning outcome? (reference activities in section 16)**
Appreciate the conceptual framework for interdisciplinary modelling that constructivist computing affords	Lectures and laboratory sessions Individual and group practical work.	All components of the assessment
Demonstrate familiarity with the basic principles of Empirical Modelling and the basic concepts of construals that embody patterns of observables, dependencies and agency	Lectures and laboratory sessions	Model analysis and documentation Lab report Individual oral examination
Understand, exercise and present standard existing EM construals	Lectures (especially those which incorporate model demonstrations) and laboratory sessions	Model analysis and documentation
Apply the principal tools and techniques deployed in constructivist computing	Laboratory sessions	Model analysis and documentation Lab report
Recognise the challenges presented by developing computer-based techniques and tools to support group work	Laboratory sessions devoted to group work	Group presentation Individual oral examination