CS4051

UNIVERSITY OF WARWICK

MSc Summer Examinations 2011/12

Introduction to Empirical Modelling

Time allowed: 3 hours

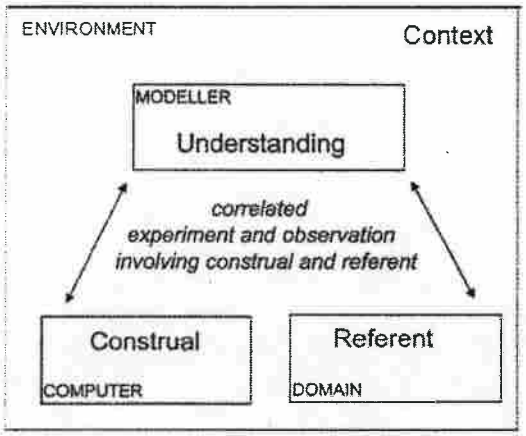
Answer question 1, and two other questions.

Question 1 carries 40 marks. The other questions carry 30 marks.
Credit will be given for evidence of familiarity with a variety of standard illustrative models.

Calculators are not allowed.

---

1. Give an informal account of Empirical Modelling (EM) with reference to the diagram:



illustrating your answer with examples drawn from the construals relating to bubblesort, modes in the vi editor, the games of Pjawns and Nim, beam detection and scoring in the game of cricket that were introduced in the module as the subject of a construal comprehension exercise.

Structure your account by addressing the following issues:

(a) the nature of the **Construal**, the **Referent**, the modeller's **Understanding** and the **Context**. [4]

(b) the ways in which the construal, referent, understanding and context can evolve. [4]

(c) the nature and role of the key concepts of **observable**, **dependency** and **agency**. [4]

(d) the way in which **definitive notations** may be used to support EM. [4]

(e) how the **Empirical Modelling Presentation Environment**, the **Dependency Modelling Tool** and **the LSD notation** can be used to document construals. [4]

and include illustrative examples relating to each of the terms in bold. [20]

The 20 marks for illustration will be allocated as follows: 2 marks for the concepts highlighted in parts (a) and (c), 3 marks in part (d), and 1 mark for each in part (c).

(Continued)

2. A key idea in deploying EM for educational applications is that learning is associated with becoming familiar with particular patterns of interaction and interpretation with a construal.

Discuss this notion and its significance for educational technology with reference to:

(a) the Experiential Framework for Learning (EFL) [6]
(b) the possibility of developing a construal through "cognitive layering" [6]
(c) the extent to which EM accommodates situation, ignorance and nonsense [6]
(d) the associations between theoretical and experimental knowledge that EM supports [6]
(e) the scope for integrating interactions with a construal in the roles of designer, teacher and learner, and the implications for constructionism. [6]

Credit will be given for reference to appropriate illustrative examples.

---

3. (a) Compare and contrast an EM perspective on computing with the traditional view of computing as rooted in "computational thinking". Frame your answer by drawing up one-page annotated diagrammatic representations (similar to those presented in the module) to depict the key concepts, principles and orientation of each viewpoint on computing. Present these representations on facing pages of your answerbook. [16]

(b) Discuss the potential implications of adopting an EM viewpoint on computing with reference to:
(i) the notion of *human computing* [7]
(ii) an alternative curriculum for computer science. [7]

Credit will be given for reference to appropriate illustrative examples.

---

4. Discuss the scope and prospects for EM as an approach to programming with reference to:

(a) what is directly experienced by the programmer when conceiving a program in traditional ways, such as: as a Turing machine, as a procedural program, as a functional program, and when developing a program using EM. [6]
(b) how – in an appropriate context – an EM construal can be viewed as a program, and how interaction with this construal then relates to a formal specification of the functionality of the program. [6]
(c) how the development of a construal using EM may be compared and contrasted with established software development processes. [6]

Illustrate your answers with reference to appropriate construals, such as the Jugs, Heapsort, Railway Animation and Clayton Tunnel construals introduced in the module. [12]

(Continued)

5. (a) Briefly describe and **illustrate** the characteristics of the following tools that have been proposed and applied for supporting EM:
   (i) Cadence [5]
   (ii) tkeden [6]
   (iii) JS-EDEN [4]
   (iv) the Abstract Definitive Machine [5]

   (b) Compare and contrast the nature of the support for EM offered by each of these tools with particular attention to the problems of developing suitable interfaces, constructing large-scale models, and exploiting concurrent and distributed implementation techniques. [7]

   (c) How far do the dependency-related features that can be found in many modern software tools go towards meeting the needs of EM? [3]

(End)