UNIVERSITY OF WARWICK

Fourth Year Examinations: Summer 2008

Introduction to Empirical Modelling

Time allowed: 3 hours

Answer **Question 1**, and **TWO** other questions.
Question 1 carries 20 marks. The other questions carry 15 marks.
The total mark awarded will be out of 50.

Read carefully the instructions on the answer book and make sure that the particulars required are entered on each answer book.

Calculators are not required and not permitted.

Credit will be given for evidence of familiarity with a variety of standard illustrative models.

---

1. (a) Describe the principles of *modelling with definitive scripts* with reference to:

   i. the role played by observables, dependencies and agents; [2]

   ii. the way in which the semantics of the script is determined; [2]

   iii. the nature and role of *definitive notations*. [2]

   (b) Briefly explain how programming based on modelling with definitive scripts differs from:

   i. functional programming; [2]

   ii. procedural programming; [2]

   iii. programming with dependency; [2]

   iv. designing and using spreadsheets. [2]

   Illustrate your answers to (a) and (b) with reference to suitable modelling and programming exercises. [6]

Continued

2. The game of **15** is a two person game in which players take turns to select a digit in the range 1 to 9. The game is won by the first player who holds three digits that sum to 15; it is drawn if all digits have been selected and neither player has achieved this goal. There is a well-known correspondence between playing the game of 15 and playing Noughts-and-Crosses. This exploits the fact that the digits 1 to 9 can be arranged in a 3-by-3 'magic square' so that the digits in each horizontal, vertical and diagonal line sum to 15. [Hint: Match the eight essentially different ways in which 15 can be expressed as a sum of three distinct digits to the eight winning lines.]

Describe how you might develop definitive scripts in EDEN to set up:

i. a model of the state-of-play in a game of **15**; [6]

ii. a viewer that enables a player to interpret this state as a state-of-play in a game of Noughts-and-Crosses. [4]

How might LSD accounts of player agents and associated artefacts be used to reflect the distinction between playing **15** with and without the viewer in place? [5]

---

3. Empirical Modelling has been proposed as an alternative conceptual framework for computing. Explain the motivation for seeking such an alternative framework with reference to:

i. the difficulties encountered in software development; [5]

ii. the limitations of knowledge representation based on logic; [5]

iii. the problems of reconciling database theory with practice. [5]

Your answer should refer to key papers on these topics by Brooks, Harel, McDermott, Cantwell-Smith and Ridley that have been discussed in the lectures. Credit will be given for explaining with reference to specific examples how Empirical Modelling relates to concerns and proposals set out in these papers.

---

4. (a) Explain how, in an LSD account:

i. agents are described and

ii. observations are classified [5]

(b) Describe the computational framework of the Abstract Definitive Machine. [5]

(c) Explain the principles behind the use of the Abstract Definitive Machine in the animation of an LSD account. [5]

Illustrate your answers to (a), (b) and (c) with reference to the animation of a train arrival-departure protocol discussed in the lectures.

Continued

5. The attached dependency graph shows some basic dependencies amongst the EDEN representations of observables extracted from the `heapsortBeynon1998` model that was discussed in the lectures. The observables depicted in the dependency graph are associated with the third element of the array, whose value is recorded as the third value in the EDEN list observable `val`. The screen display for the `heapsortBeynon1998` model shows the array of numbers that is being sorted, together with the nodes of the associated heap in the form of a binary tree. The nodes and edges in the tree display are colour-coded to indicate whether or not the heap condition is valid at a node, and the significant order relations between the values associated with nodes.

(a) Illustrate, by identifying specific examples of nodes in each class, how the nodes of the dependency graphs can be classified into:

- observables with explicitly and with implicitly defined values;
- special-purpose functions defined by the modeller, and
- standard data types and operators associated with the EDEN interpreter. [3]

(b) Identify the vertex in the dependency graph that:

i. determines the colour of the circle at the tree node associated with the third array element. [1]

ii. specifies the edge that joins the tree node associated with the third array element to that associated with the sixth array element. [1]

iii. defines a line that borders on the third cell in the array. [1]

(c) Building on your knowledge of the model:

i. explain the significance of the observable `hc3`, and that of each of the dependency relations associated with a directed edge terminating at this observable. [2]

ii. explain the mechanism used to specify the visual representation of the edges that connect tree nodes. [1]

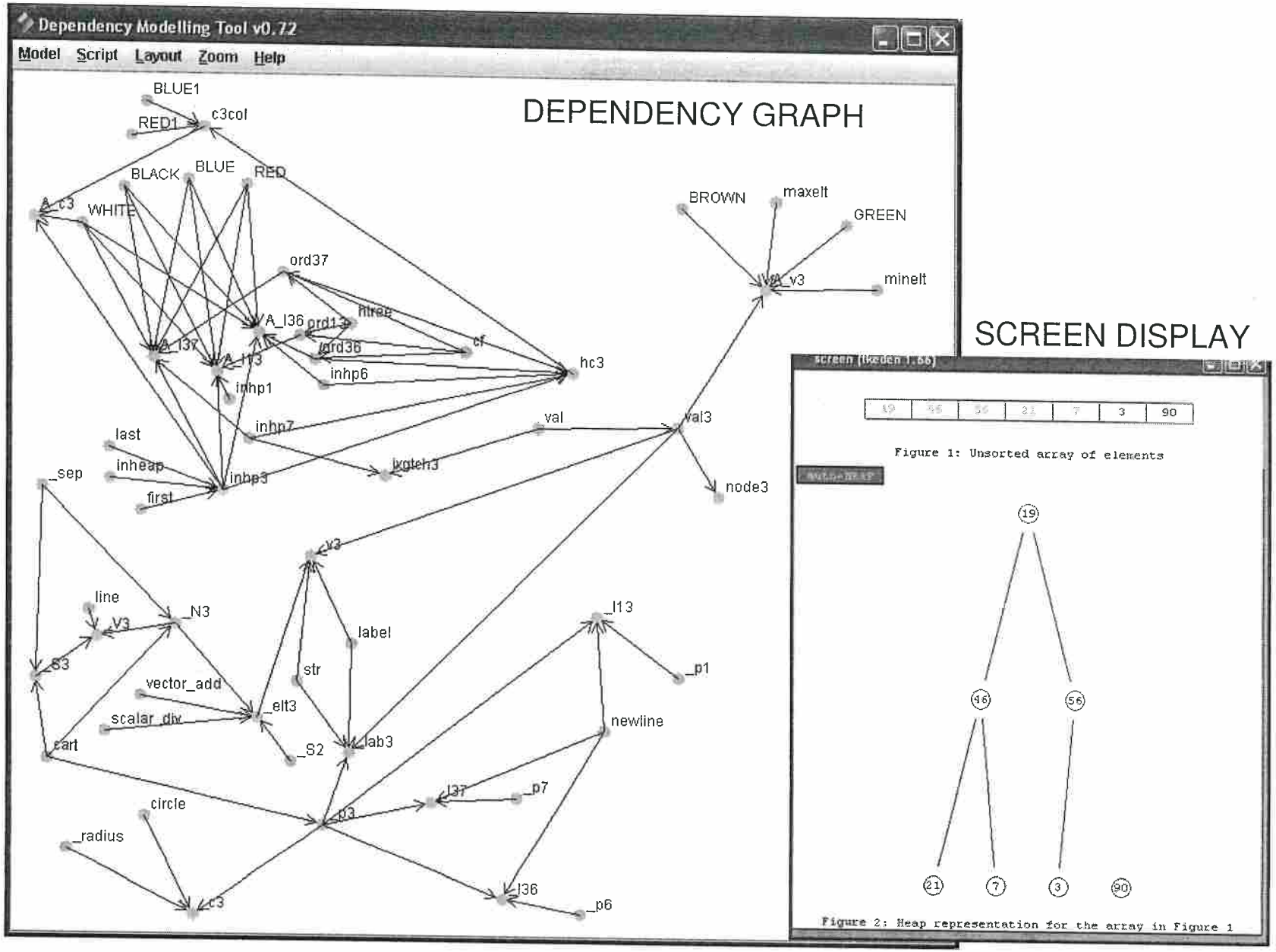
iii. interpret the sequence of updates to observables that occur through propagation via dependency relations when `val[3]` is redefined. [2]

iv. infer from your answer to iii what significant dependency relation is missing from the dependency graph. [1]

(d) In what sense does Empirical Modelling reflect the motivating idea behind introducing a data structure to model an abstract algorithm more faithfully than a conventional implementation? [1]

(e) In what respects does Empirical Modelling resemble traditional engineering rather than conventional software development? [2]

End


Dependency Modelling Tool v0.72
Model Script Layout Zoom Help
DEPENDENCY GRAPH
BLUE1
c3col
RED1
BLACK
BLUE
RED
WHITE
ord37
A_l36
ord13
three
A_l37
A_l13
ord36
cf
inhp6
inhp1
hc3
inhp7
val
val3
last
inheap
first
inhp3
lxgteh3
node3
_sep
v3
line
V3
_N3
label
_l13
_p1
str
vector_add
scalar_div
elt3
newline
cart
_S2
lab3
l37
_p7
circle
_radius
p3
l36
_p6
BROWN
maxelt
GREEN
minelt
v3
SCREEN DISPLAY
Figure 1: Unsorted array of elements
Figure 2: Heap representation for the array in Figure 1