UNIVERSITY OF ABERDEEN **SESSION 2017-2018**

Examination in CS3025 (Knowledge-based Systems)

13 December 2017 **12:00 – 14:00**

Candidates are not permitted to leave the Examination Room during the first or last half hours of the examination.

Appendix (including some tables and algorithms used in lectures) is available after the questions.

Each question is worth 25 marks; the marks for each part of a question are shown in brackets.

1.

(a) Given the following competency question: "Which is the shortest way to walk to the library?", answer the following two questions:

1) What are the features of the competency question? [2]
2) What are the elements of the competency question? [2]

(b) Given the following Book table:

bookID	hasTitle	hasPublisher
B001	Ontology-Driven Software Development	Springer
B002	Semantic Web Enabled Software Engineering	AKA

(1) Transform the four records into the corresponding RDF statements in N3 syntax, if they are representable in RDF. [2]
(2) Construct an interpretation for this knowledge base [4]
(3) Assume that we will consider a new concept Popular, with the following interpretation $Popular^I=\{Springer\}$, please give the interpretation for the following class description: [2]

$$Book \sqcap \exists hasPublisher.Popular$$

(c) Consider the ontology O consisting of the following axioms:

- Class(Cow partial Herbivore)
- Class(MadCow partial (intersectionOf(Cow restriction(eat someValuesFrom(AnimalComponent)))))
- Class(Herbivore partial restriction(eat allValuesFrom(Vegetable)))
- DisjointClasses(AnimalComponent Vegetable)

Here are your tasks:

i) Transform the above ontology into DL syntax. [2]
ii) Use the tableaux algorithm to check if MadCow is satisfiable. [9]
iii) If MadCow is not satisfiable, which axiom(s) need to be changed to make it satisfiable. [2]

TURN OVER

2.

(a) Explain how the ordering of activated rules on the agenda can be influenced by a Jess program. [4]

(b) Use the while function to write Jess code which **multiplies** the **odd** numbers from 1 to 100, and then prints out the result. [6]

(c) Based on the partially completed rule extract-list and the facts in the working memory below, work out the following questions:

```
RULE-1:

(defrule extract-list
       (data ?first variable_A)
       =>
       Statement_1
)
```

```
WM-1:

  Fact-1: (data tiger whale crow snake spider)
  Fact-2: (data Bill Bob Mary Jill)
  Fact-3: (data tiger)
```

i. In Rule-1, given that variable A is a multifield variable, complete the pattern on the LHS of the rule. NB: you may create a new name for *variable_A*. [2]

ii. Suppose *statement_1* prints out the value of *variable_A*. Write down such a statement in Jess code. [2]

iii. Given the facts in the WM-1, how many times will the rule be activated? Write down the output for all the possible activation(s)/execution(s). [3]

iv. If we extend RULE-1 into RULE-2, by adding a multifield wildcard "$?", and given the facts in the WM-1, please complete the pattern on the LHS of the rule, and then answer how many times will the rule be activated? Write down the output for all the possible activation(s)/execution(s). [4]

```
RULE-2:

(defrule extract-list
       (data ?first $? variable_A)
       =>
       Statement_1
)
```

```
WM-1:

  Fact-1: (data tiger whale crow snake
spider)
  Fact-2: (data Bill Bob Mary Jill)
  Fact-3: (data tiger)
```

v. For security reason, we want to hide the word (only only word) after "secret". Given RULE-3 and the facts in the WM-3, please complete the pattern ***(data .. .. ..)*** on the LHS to hide the word, and complete the ***Statement_3*** to display the facts without the hidden word, and then write down the output for all the possible activation(s)/execution(s). [4]

```
RULE-3:

(defrule extract-list
       (data .. .. ..  )
       =>
       Statement_3
)
```

```
WM-3:

  Fact-1: (data tiger whale crow secret spider)
  Fact-2: (data secret Bill Mary Jill)
```

TURN OVER (APPENDIX AVAILABLE)