Unit 3 formative activities

Activity 1

Attempt the following questions from the module core text:

- *Chapter 2 Question 4 - The barber's paradox*.
- *Chapter 3 Question 4 - A Canadian variant of an old puzzle*.

Answer: Not possible to execute as these pages are not included in the access provided to the core text.

Activity 2

Read the paper by Palomino et al (2005) and review the 'crossing problem' diagram provided in the Lecturecast.

- *Create a set of statements in first order logic (FOL) that represent the states shown – for example you may define two functions left and right and therefore the first state could be represented as: Left(F) and left(W) and Left(G) and left(C). Define your own set of FOL statements for the entire diagram*.

Answer:

To represent the crossing problem using First Order Logic (FOL), we need to define some predicates and functions. For this problem, we can define the following predicates:

- Left(X): Indicates that the entity X is on the left side of the river.
- Right(X): Indicates that the entity X is on the right side of the river.

Additionally, we define constants for the entities:

- F: The farmer
- W: The wolf
- G: The goat
- C: The cabbage

Let's represent the different states using these predicates:

Initial State:
Everyone is on the left side.

- $\text{Left}(F) \land \text{Left}(W) \land \text{Left}(G) \land \text{Left}(C)$

State 1:
Farmer takes the goat across the river.

- $\text{Right}(F) \land \text{Right}(G) \land \text{Left}(W) \land \text{Left}(C)$

State 2:
Farmer goes back alone.

- Left(F) ∧ Right(G) ∧ Left(W) ∧ Left(C)

State 3:
Farmer takes the wolf across the river.

- Right(F) ∧ Right(W) ∧ Right(G) ∧ Left(C)

State 4:
Farmer takes the goat back.

- Left(F) ∧ Right(W) ∧ Left(G) ∧ Left(C)

State 5:
Farmer takes the cabbage across.

- Right(F) ∧ Right(W) ∧ Right(C) ∧ Left(G)

State 6:
Farmer goes back alone.

- Left(F) ∧ Right(W) ∧ Right(C) ∧ Left(G)

State 7:
Farmer takes the goat across.

- Right(F) ∧ Right(W) ∧ Right(G) ∧ Right(C)

This sequence represents a safe crossing without the goat being left alone with the wolf or the cabbage. Each state follows the constraints given in the problem statement.

- *You have been provided with solutions to the 'crossing problem' written in Lisp (here), Prolog (here) and Maude (here). Compare your FOL clauses with the various implementations – which of the implementations provides the closest match to your FOL version?*

Answer: Prolog