Project

Duc Minh Le (Tom Le)

ID: 3561723

Athabasca University

COMP 456 – Artificial Intelligence

Dr. Sidi Esmahi

Steve Leung

# Expert Shell

Modified version of the expert shell (EXSHELL) as detailed in Chapter 6 of "AI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and Java" by George F. Luger and Willian A. Stubblefield.

The following changes were made so that I would work with GNU Prolog 1.4.5:

/\* Predefined prolog's predicates \*/

:- dynamic(known/2).

:- dynamic(assert/1).

To prevent this error



Changing not(X) to \+ X

Usage:

Compile then use “run.” to start the expert shell and request a knowledge base.

You may enter any knowledge base name of your choice.

Text

Description automatically generated

# Knowledge Base

Knowledge base of solving the Farmer Wolf Goat and Cabbage puzzle for the expert shell.

State represents the current position of the Farmer Wolf Goat and Cabbage and are in the following syntax: state(Farmer, Wolf, Goat, Cabbage)

Start(state(e, e, e ,e)), the starting state, everyone is on the east bank of the river  
Goal(state(w, w, w ,w)), the goal state, everyone has crossed the river to its west bank.

Askable(start), default askable, always enter “100.” to start the solving process

Askable(two\_paths), ask the user if they would like to see only the 1st path “-100.”, or both paths “100.”

**Usage**:

Compile by enter the knowledge base name when prompted by the exshell

solve(path(X), CF). - Where X is the possible paths  
User query: start – Always answer “100.” Aka yes

User query: two\_paths - “-100.” aka no for solving only the 1st path, “100.” Aka yes for both paths

State syntax: state(Farmer, Wolf, Goat, Cabbage)

1st path

F W G C

e e e e

w e w e

e e w e

(w w w e)

(e w e e)

w w e w

e w e w

w w w w

2nd path

F W G C

e e e e

w e w e

e e w e

(w e w w)

(e e e w)

w w e w

e w e w

w w w w