EXPERT SYSTEM RULES

This appendix is included for reference to help the interested reader to see the complete set of rules used to develop the small expert system described in Chapter 4. This appendix includes the whole file in textual form, which should facilitate the reader’s understanding of the components that comprise a simple expert system. The interested reader can use this appendix further to build his or her own simple expert system as well.

As a guide, the expert system code below is divided into several parts, which each part beginning with the symbol “%” followed by a short descriptor of its content. For instance, the third line below is, “% folder”. This starts the first part, which is the folder section. This particular expert system does not utilize extensively the folder feature of the shell tool.

The second part begins with “% knowledgebase”. This is where the problem-solving goal (i.e., “goals = [technology_use]”) is expressed, where the main action of the expert system is specified, and where some HTML formatting is accomplished.

The third part begins with “% fact”. This is where declarative, factual information is stored. If this expert system had been loaded with pre-existing facts, then such declarative chunks would be included here. Such facts are used in many systems for aspects of the world that remain constant or at least very stable. This particular expert system does not utilize facts. The reason is twofold. First, few aspects of asynchronous instructional tools remain constant, so facts entered in this section would have to be updated frequently. But second and more importantly, this small, simple, illustrative application gathers all of its necessary factual information from users, and it was programmed with many facts “hardwired” into the rules that follow. This does not represent first-class programming practice. But it suffices to development the expert system for demonstration.

The fourth part begins with “% question”. This is the part that guides the expert system’s interrogative interaction with the user and hence where it acquires factual information to use for inference and decision-making. For instance, the first question, labeled “prompt = text(“5. What computer skills do you expect students to have?”),” is asking about student computer skills. Notice the program offers a predetermined set of choices, labeled: “choices = [“none”, “browsing”, “programming”]”. Of course alternate approaches can be pursued as well.

The sixth part begins with “% sql”. This small, simple, illustrative expert system does not utilize the SQL feature.

The seventh part begins with “% rule_set”. This is where procedural, explicit knowledge is stored. The two rules included in the figure discussed in Chapter 4 are taken from this part. This is where the inferential problem solving and decision making logic of the expert system is programmed.
The eighth and ninth parts begin with "% rules_table" and "% data_table", respectively. As noted above in connection with some other parts, this small, simple, illustrative expert system does not utilize these features.

The tenth part begins with "% text". This is where prescripted textual messages are included. As with the questions above, such textual messages are presented interactively with the user.

***

knowledgewright_jig(basic, 11).
knowledgewright_license(academic_personal, '[]').

% folder
:- indexed folder(1,0,0).
% knowledgebase
:- indexed knowledgebase(1,0,0).
knowledgebase(main, /, [description = "Implements part of the Bates & Poole (2003) SECTIONS Model for selecting and using education technology.",
goals = [technology_use],
date_format = 'm/d/y',
odbc = "",
charset = "",
question_separator = "<P>",
menu_separator = "<BR>",
value_separator = "<P>",
question_top = text("<HTML><HEAD></HEAD><BODY><FORM METHOD="POST" ACTION="/cgi-bin/kwcgibasic.exe" + system(cgi_parameters) + ">")",
question_bottom = text("<P><INPUT NAME="Submit" TYPE="Submit" VALUE="Submit"></INPUT><INPUT TYPE="Reset" VALUE="Reset"></INPUT></FORM></BODY></HTML>"),
output_top = text("<HTML><HEAD></HEAD><BODY>"),
output_continue = text("<FORM METHOD="POST" ACTION="/cgi-bin/kwcgibasic.exe" + system(cgi_parameters) + ">")<INPUT NAME="Submit" TYPE="Submit" VALUE="Continue"></INPUT></FORM>"),
output_bottom = text("</BODY></HTML>"),
}

% fact
:- indexed fact(1,0,0).
% question
:- indexed question(1,0,0).
question(student_skills, /, [
prompt = text("5. What computer skills do you expect students to have?"),
question_type = menu_single_choice,
question_style = listbox,
choices = ["none", "browsing", "programming"],
}
‘rule-display_choices’ = [["rule_text", "display_text"]],
answer_type = text,
length = 20,
height = 1,
default = "",
ask_also = ‘[]’
}).

question(student_priorDL, /, [
prompt = text("7. What prior approaches to learning are students likely to have?")

question_type = menu_single_choice,
question_style = listbox,
choices = ["classroom only", "classroom and VTE", "classroom and media", "classroom and webbased"],
‘rule-display_choices’ = [["rule_text", "display_text"]],
answer_type = text,
length = 20,
height = 1,
default = "",
ask_also = ‘[]’
]).

question(student_demographics, /, [
prompt = text("2. What are the likely demographics of the students you will be teaching?")

question_type = menu_single_choice,
question_style = listbox,
choices = ["resident students", "professionals ashore", "officers at sea"],
‘rule-display_choices’ = [["rule_text", "display_text"]],
answer_type = text,
length = 20,
height = 1,
default = "",
ask_also = ‘[]’
]).

question(equipment_cost, /, [
prompt = text("6. Will students be able to justify the marginal cost of technology required for the course?")

question_type = menu_single_choice,
question_style = listbox,
choices = ["no", "yes"],
‘rule-display_choices’ = [["rule_text", "display_text"]],
answer_type = text,
length = 20,
Appendix

height = 1,
default = "",
ask_also = '[]'
]).
question(student_access, /, [
prompt = text("3. To which technologies are students likely to have regular
access?"),
question_type = menu_single_choice,
question_style = listbox,
choices = ["T1+", "DSL or cable", "modem", "other", "none"],
'rule-display_choices' = [["rule_text", "display_text"]],
answer_type = text,
length = 20,
height = 1,
default = "",
ask_also = '[]'
]).

% sql
:- indexed sql(1,0,0).
% rule_set
:- indexed rule_set(1,0,0).
rule_set(access, /, [
description = "",
type = single_value,
rules = [[conditions, value], [student_access = "modem", text("low bandwidth")], [student_access = "DSL or cable", text("high bandwidth")], [student_access = "T1+", text("high bandwidth")], [student_access = "other", text("low bandwidth")], [student_access = "none", text("no bandwidth")]]
]).
rule_set(cost, /, [
description = "",
type = single_value,
rules = [[conditions, value], [equipment_cost = "no", text("not cost-effective")], [equipment_cost = "yes", text("cost-effective")]]
]).
rule_set(feasible, /, [
description = "",
type = single_value,
rules = [[conditions, value], [cost = "not cost-effective", text("infeasible")], [access = "low bandwidth" and cost = "cost-effective", text("feasible")], [access = "high bandwidth" and cost = "cost-effective", text("feasible")], [access = "no bandwidth" and cost = "cost-effective", text("feasible")]]
)
rule_set(technology, /, [description = "", type = single_value, rules = [[conditions, value], [feasible = "feasible" and access = "low bandwidth", text("use minimal graphics & interaction via technology")], [feasible = "feasible" and access = "high bandwidth", text("use full graphics & interaction via technology")], [default, text("current technology does not appear to support your plan")]]). rule_set(class_type, /, [description = "", type = single_value, rules = [[conditions, value], [student_demographics = "resident students", text("an enhanced, mediated, or web-based course")], [student_demographics = "professionals ashore" and access = "no bandwidth", text("a web-based course")], [student_demographics = "officers at sea" and access = "no bandwidth", text("a web-based course")], [default, text("your current course plan appears infeasible")]]).  

% rules_table :- indexed rules_table(1,0,0).
% data_table :- indexed data_table(1,0,0).
% text :- indexed text(1,0,0).
text(technology_use, /, [description = "", type = text, file = "", text = text("Given the information you have provided, SECTIONS Advisor recommends: " + class_type + "; " + technology + ". Key factors in this recommendation include: " + access + ", " + student_demographics + " and " + cost + ". Thank you for using SECTIONS Advisor. 
")].