北京邮电大学　计算机学院

《人工智能原理》实验报告

姓名＿＿王睿嘉＿＿

学号 2015211906

班级 2015211307

一、 实验要求和原理描述

1. 实验要求

1） 对 CLIPS 进行熟悉；

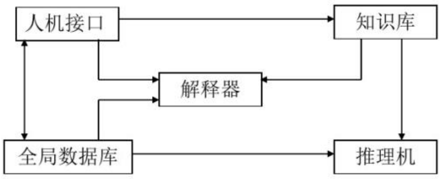
2） 对 Animal.clp 程序进行阅读分析：

结合一个运行实例，写出系统具体推理的步骤，并根据代码和系统的处理方法分析其求解过程。

2. 实验原理

2.1 组成

知识库和推理机是专家系统的核心。随着系统的不断完善，形成了人机接口、知识库、推理机、解释器和全局数据库等 5 个部分，并采用产生式规则作为基本的知识表示方式。系统结构如下图所示：



知识库是问题求解所需的领域知识的集合，包括基本事实、规则和其他有关信息。知识的表示形式可以是多种多样的，包括框架、规则、语义网络等。知识库中的知识源于领域专家，是决定专家系统能力的关键，即知识的质量和数量决定着专家系统的水平。一般来说，专家系统的知识库与专家系统程序是相互独立的，用户可以通过改变和完善知识库中的知识内容来提高专家系统的性能。

推理机是问题求解的核心执行机构，它实际上是对知识进行解释的程序。根据语义，按一定策略找到知识，并进行解释执行，把结果记录到动态库的适当空间中。推理机的程序与知识库的具体内容无关，即推理机和知识库是分离的，这是专家系统的重要特征。优点是，对知识库的修改无需改动推理机，但是纯粹的形式推理会降低问题求解的效率。因此，将推理机和知识库相结合也不失为一种可选方法。

人机接口是系统与用户进行交流时的接口。通过该接口，用户输入基本信息，并回答系统提出的相关问题，系统输出推理结果及相应解释。

全局数据库也称为动态库、工作存储器或综合数据库，是反映当前问题求解状态的集合，用于存放系统运行过程中所产生的信息及所需要的原始数据，包括用户输入的信息、推理的中间结果、推理过程的记录等。全局数据库中各种事实、命题和关系，既是推理机选用知识的依据，也是解释机制获得推理路径的来源。

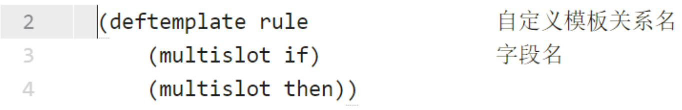
解释器对求解过程做出说明，并回答用户的提问。两个最基本的问题是“why”和“how”。解释机制涉及程序的透明性，它让用户理解程序正在做什么及为什么这样做，向其提供了一个关于系统的认识窗口。很多情况下，解释机制是非常重要的。为了回答“为什么”得到某个结论的询问，系统通常需要反向跟踪动态库中保存的推理路径，并把它翻译成用户能够接受的自然语言表达方式。

动物识别的专家系统，同样是由以上部分组成。

1） 事实

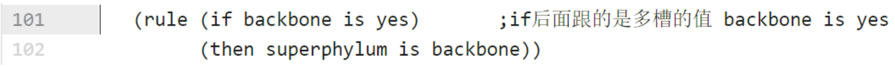
事实由关系名（relation name）后跟零或多个槽（slot）及它们的相应值组成，用来表示已知的数据或信息。可用自定义模板和自定义事实结构来表示事实，形如：

- 自定义模板



```
(deftemplate rule                  自定义模板关系名
      (multislot if)               字段名
      (multislot then))
```

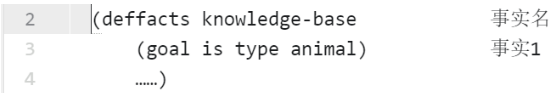
在示例中，字段名后设置的是多槽（multislot）。槽可以使系统摆脱事实字段必须顺序输入的约束，输入事实时指明槽名即可，例如：



```
    (rule (if backbone is yes)        ;if后面跟的是多槽的值 backbone is yes
          (then superphylum is backbone))
```

- 自定义事实

用自定义事实关键字（deffacts）来构建知识库。自定义事实的声明，必须在deffacts后面指定一个事实名，如knowledge–base。事实名后，便是将要被声明到事实表中的事实。

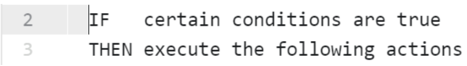


```
(deffacts knowledge-base              事实名
    (goal is type animal)             事实1
    ……)
```

当CLIPS系统启动推理时，会把所有用deffacts定义的事实自动添加到工作存储器中，并提供assert命令，把事实添加到工作存储器中；retract命令，删除指定事实；modify命令，修改自定义事实的槽值；duplicate命令，复制事实；clear命令，删除所有事实。

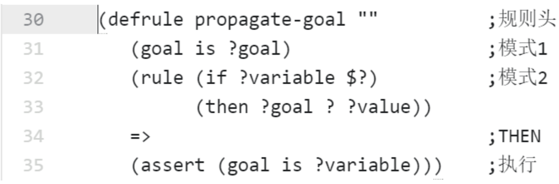
2） 规则

为了完成一项有价值的工作，专家系统必须得有事实和规则。规则用来表示系统推理的有关知识。if–then规则可以用自然语言与计算机语言混合表示，如下所示（伪代码）：



```
IF   certain conditions are true
THEN execute the following actions
```

采用CLIPS语法将上面的伪代码写成一个事实和一个命名为propagate–goal的规则。其中，规则名紧跟在关键字defrule后面：



```
(defrule propagate-goal ""           ;规则头
    (goal is ?goal)                  ;模式1
    (rule (if ?variable $?)          ;模式2
          (then ?goal ? ?value))
    =>                               ;THEN
    (assert (goal is ?variable)))    ;执行
```

可以在一条规则中加入多条模式或行为。只有当规则中所有的模式都被事实表中的事实满足时，规则才能被触发。这种约束类型被称为逻辑与条件元素（logical AND conditional element），是关于布尔型的“与”关系。

2.2 推理与控制机制

产生式系统的推理循环可分为4个阶段：

1） 模式匹配

从知识库的第一条规则开始，依次扫描所有规则，把规则的前提与综合存储器中的当前事实相匹配，以搜索满足条件的规则。

2） 冲突消解

在发生冲突，即多条规则同时被匹配时，根据预先确定的冲突消解策略，确定触发的规则。

3） 激活规则

调用匹配触发规则所有子目标的事实。

4） 动作

把触发规则的结论添加到综合存储器。

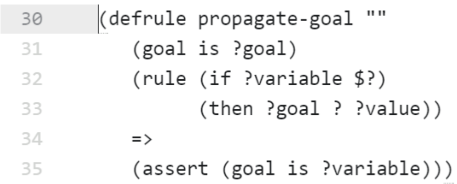
CLIPS 推理机重复上述循环，不断地扫描规则的模式，并把和事实匹配的规则激活，放入议程（Agenda）中。议程实际上是一个堆栈，所有激活的规则按优先级次序压入堆栈。若新压入规则的优先级小于栈顶规则的优先级，则被压入到栈的下部，直到所有比它优先级高的规则都在此规则的上面。

CLIPS 提供两种具体的技术控制规则的执行：优先级和模块。可以在定义规则时，设置规则的优先级，使议程中的多条规则按优先级顺序执行。对大型的具有复杂规则库的情况，可以定义不同的模块来划分知识库，通过改变模块的焦点控制当前被执行的模块。

2.3 代码说明

只对推理机制的代码进行分析，从而在讲解之后的运行实例时，可以更好地进行整体过程的说明而不用再进行具体代码细节的剖析。（每个规则的说明顺序和系统运行调用的先后顺序一致）

在 CLIPS 专家系统中，规则语句进行的是正向推理，即将 LHS 里的所有模式和知识库中的事实匹配，均匹配上时，=>后的行动才会被执行。



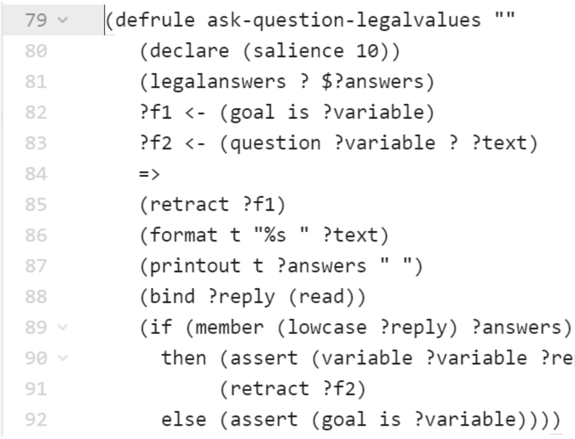
```
(defrule propagate-goal ""
   (goal is ?goal)
   (rule (if ?variable $?)
         (then ?goal ? ?value))
   =>
   (assert (goal is ?variable)))
```

规则 propagate-goal 的模式 1 要求匹配的事实是 goal is 和一个单字段值，且这个字段值会被约束到变量?goal 上。

模式 2 要求匹配的事实使用 rule 模板，if 槽需为变量?goal 后跟零或多个字段值，then 槽第一个字段值必须与变量?goal 相同，最后一个字段值被约束到变量?value 上。

前提满足后，该规则会插入 goal is ?variable，这里?variable 是指之前约束到?variable 的值。

这个规则反映的是逆向推理的过程。综合数据库中的 f-1（goal is type.animal），是系统推理的目标，将其与 if-then 语句匹配，并将匹配成功的 if-then 的前提提取出来作为新的目标。



```
(defrule ask-question-legalvalues ""
   (declare (salience 10))
   (legalanswers ? $?answers)
   ?f1 <- (goal is ?variable)
   ?f2 <- (question ?variable ? ?text)
   =>
   (retract ?f1)
   (format t "%s " ?text)
   (printout t ?answers " ")
   (bind ?reply (read))
   (if (member (lowcase ?reply) ?answers)
     then (assert (variable ?variable ?reply))
          (retract ?f2)
     else (assert (goal is ?variable))))
```

规则 ask-question-legalvalues 用 declare 声明了该规则的优先值为 10，这样可以保证在与其他低优先值的激活规则相比，它可以优先执行。

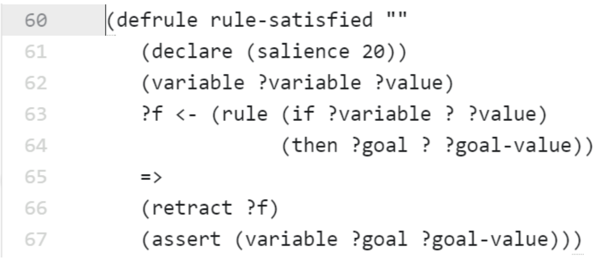
模式 1 要求匹配 legalanswers 后跟通配符和多字段值，并将多字段值约束到变量?answers。

模式 2 将匹配事实 goal is ?variable 的索引存入变量?f1，且 goal is 后面的单字段值约束到变量?variable。

模式 3 将匹配事实 question ?variable ? ?text 的索引存入变量?f2，?variable 的字段值必须与模式 2 中的相同。通配符后的单字段值约束到变量?text。

前提满足后，该规则会撤销约束到?f1 的事实，将?text 的单字段值变成字符串格式，并与约束在变量?answer 上的字段值一同打印显示。使用关键字 bind 将 read 操作从终端输入的单字段值绑定到变量?reply 中。紧接着，进行条件函数推理。用 member 检查输入值是否为变量?answer 中约束的多字段值之一。由于存在返回值，若为 1，则插入事实（variable ?variable ?reply）；否则，撤销约束到?f2 的事实。

该规则的推理是根据当前的目标属性进行提问，并用提问的答案和当前的目标属性生成一个事实插入综合数据库。



```
(defrule rule-satisfied ""
    (declare (salience 20))
    (variable ?variable ?value)
    ?f <- (rule (if ?variable ? ?value)
                (then ?goal ? ?goal-value))
    =>
    (retract ?f)
    (assert (variable ?goal ?goal-value)))
```

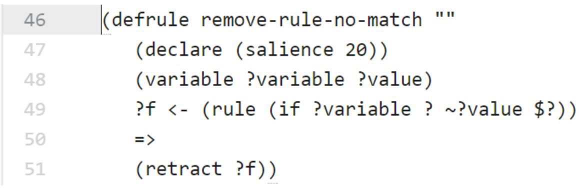
规则 rule—satisfied 用 declare 声明了该规则的优先值为 20，这样可以保证与其他低优先值的激活规则相比，它可以优先执行。

模式 1 要求匹配事实(variable ?variable ?value)，两个单字段值分别约束到变量?variable 和?value。

模式 2 将匹配事实（rule（if ?variable ? ?value））（then ?goal ? ?goal—value）的索引存入变量?f，?variable 和?value 的值要与模式 1 中的相同。Then 的两个槽值分别约束到变量?goal 和?goal—value。

前提满足后，该规则会撤销约束到?f 的事实，并插入新的事实（variable ?goal ?goal—value）。

该规则的推理过程是，当得到确切的一个动物属性值（通常是在 ask—question—legalvalues 规则后进行）时，将它插入综合数据库。



```
(defrule remove-rule-no-match ""
    (declare (salience 20))
    (variable ?variable ?value)
    ?f <- (rule (if ?variable ? ~?value $?))
    =>
    (retract ?f))
```

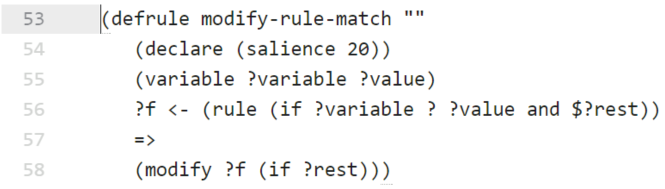
规则 remove—rule—no—match 用 declare 声明了该规则的优先值为 20，这样可以保证与其他低优先值的激活规则相比，它可以优先执行。

模式 1 要求匹配事实(variable ?variable ?value)，两个单字段值分别约束到变量?variable 和?value。

模式 2 将匹配事实 rule（if ?variable ? ~?value $?）的索引存入变量?f。其中，?variable 的值要与模式 1 中的相同。~?value 表示约束的字段值不能等于模式 1 中?value 的值，$?表示可以有零或多个字段值。

前提满足后，该规则会撤销约束到?f 的事实。

该规则体现的推理过程是，当得到确切的一个动物属性值（通常是在 ask—question—legalvalues 规则后进行）时，将那些和这个属性不匹配的事实从综合数据库中移除，这样可以减轻系统做规则匹配的工作量，提高效率。



```
(defrule modify-rule-match ""
    (declare (salience 20))
    (variable ?variable ?value)
    ?f <- (rule (if ?variable ? ?value and $?rest))
    =>
    (modify ?f (if ?rest)))
```

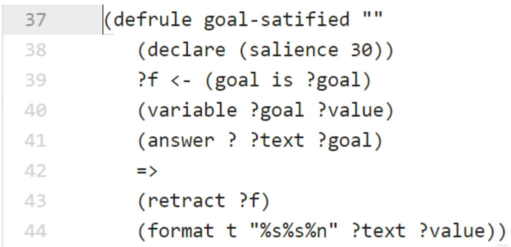
规则 modify—rule—match 用 declare 声明了该规则的优先值为 20，这样可以保证与其他低优先值的激活规则相比，它可以优先执行。

模式 1 要求匹配事实(variable ?variable ?value)，两个单字段值分别约束到变量?variable 和?value。

模式 2 将匹配事实 rule（if ?variable ? ?value and $?rest）的索引存入变量?f。其中，?variable 和?value 的值要与模式 1 中的相同，and 后的多字段值被约束到变量?rest。

前提满足后，该规则修改?f 中 if 的槽值，将其替换成?rest 的值，并作为新的事实被添加到综合数据库中，撤销约束到?f 的事实索引。

该规则体现的推理过程是，当得到确切的一个动物属性值（通常是在 ask—question—legalvalues 规则后进行）时，对那些需要同这个属性一起作条件的其他属性的事实进行相应修改，并去除有关这个属性的条件语句。



```
(defrule goal-satified ""
    (declare (salience 30))
    ?f <- (goal is ?goal)
    (variable ?goal ?value)
    (answer ? ?text ?goal)
    =>
    (retract ?f)
    (format t "%s%s%n" ?text ?value))
```

规则 goal—satisfied 用 declare 声明了该规则的优先值为 20，这样可以保证与其他低优先值的激活规则相比，它可以优先执行。

模式 1 要求匹配 goal is 和一个单字段值，且该字段值会被约束到变量?goal 上，相应索引存入变量?f 中。

模式 2 要求匹配事实（variable ?goal ?value）。variable 后的两个字段值分别约束到变量?goal 和?value。其中，?goal 的值要与模式 1 中的相同。

模式 3 要求匹配事实(answer ? ?text ?goal)。通配符后的单字段值约束到变量?text。其中，?goal 的值要与模式 1 中的相同。

前提满足后，该规则会撤销约束到?f 的事实，并将?text 和?value 的单字段值以字符串格式打印显示。

该规则体现的推理过程是，用户通过回答问题使系统推理到最终目标，并将其显示到终端上。

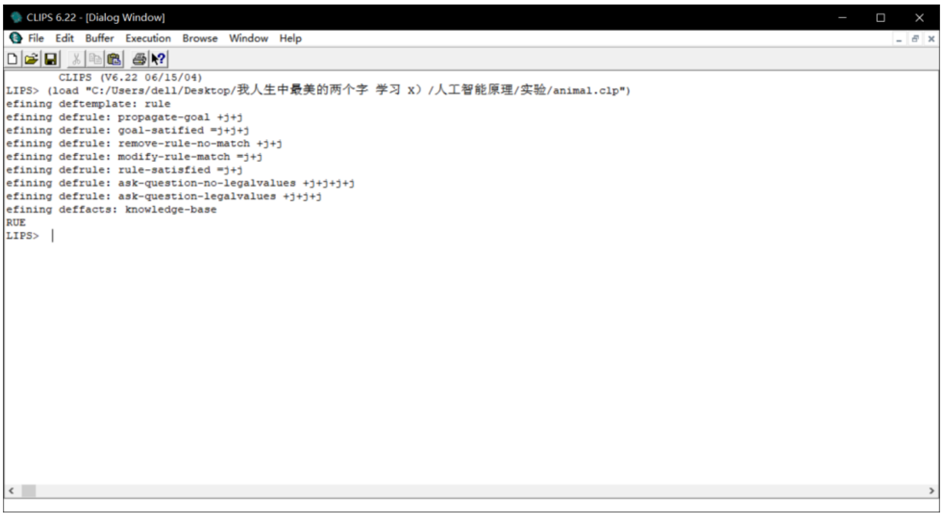
规则 ask—question—no—legalvalues 不做解释，因为在实例执行中没有使用。

二、 实验内容和心得

1. 实验内容

1.1 加载

加载后，编译程序界面如下：



```
         CLIPS (V6.22 06/15/04)
LIPS> (load "C:/Users/dell/Desktop/我人生中最美的两个字 学习 x) /人工智能原理/实验/animal.clp")
efining deftemplate: rule
efining defrule: propagate-goal +j+j
efining defrule: goal-satified =j+j+j
efining defrule: remove-rule-no-match +j+j
efining defrule: modify-rule-match =j+j
efining defrule: rule-satisfied =j+j
efining defrule: ask-question-no-legalvalues +j+j+j+j
efining defrule: ask-question-legalvalues +j+j+j
efining deffacts: knowledge-base
RUE
LIPS> |
```

此步骤中，CLIPS 加载了一个自定义模板，包含 7 个自定义规则和 1 个自定义事实。其中，knowledge-base 作为动物识别的综合数据库提供了所有事实。当专家系统进行推理时，会将规则 LHS 中的模式与综合数据库中的事实做匹配。

1.2 Reset

有 3 个动作，如下：

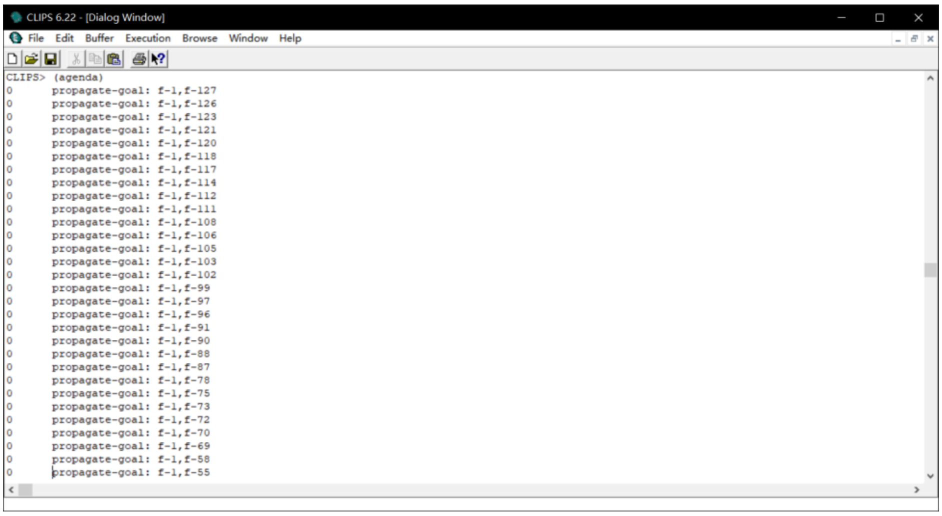
a) 自动 focus MAIN；

b) 预定义事实添加到 MAIN 模块的 FACTS 中。共有 130 条事实，包括 intial-fact f-0。

事实列表会在报告最后以附录的形式说明，在这里简要列举：

f-0	(initial-fact)
f-1	(goal is type.animal)
f-2	(legalanswers are yes no)
f-3˜f-128	rule or question
f-129	(answer is "I think your animal is a " type.animal)

c) Agenda 中出现 42 条被激发的规则：

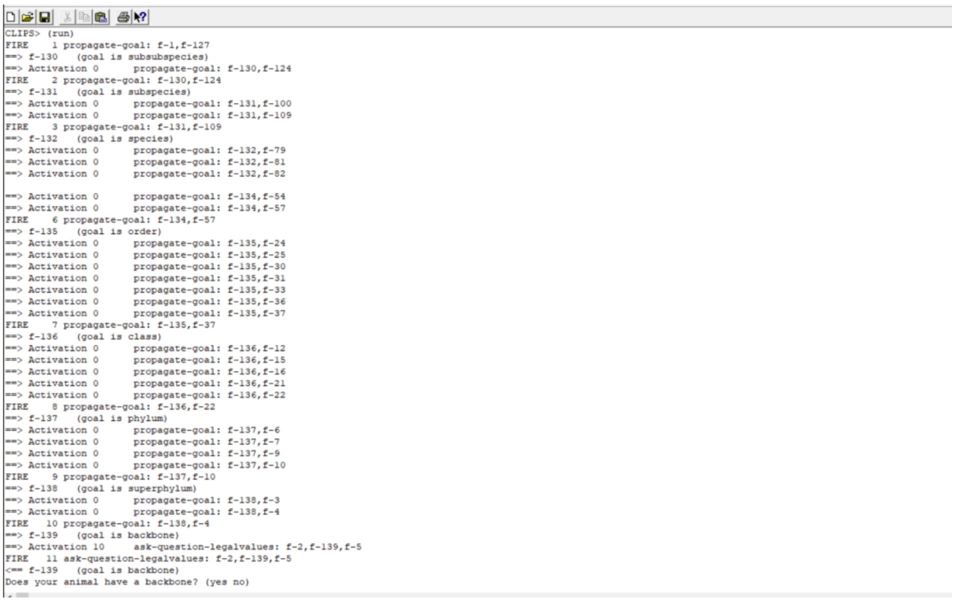


```
CLIPS> (agenda)
0       propagate-goal: f-1,f-127
0       propagate-goal: f-1,f-126
0       propagate-goal: f-1,f-123
0       propagate-goal: f-1,f-121
0       propagate-goal: f-1,f-120
0       propagate-goal: f-1,f-118
0       propagate-goal: f-1,f-117
0       propagate-goal: f-1,f-114
0       propagate-goal: f-1,f-112
0       propagate-goal: f-1,f-111
0       propagate-goal: f-1,f-108
0       propagate-goal: f-1,f-106
0       propagate-goal: f-1,f-105
0       propagate-goal: f-1,f-103
0       propagate-goal: f-1,f-102
0       propagate-goal: f-1,f-99
0       propagate-goal: f-1,f-97
0       propagate-goal: f-1,f-96
0       propagate-goal: f-1,f-91
0       propagate-goal: f-1,f-90
0       propagate-goal: f-1,f-88
0       propagate-goal: f-1,f-87
0       propagate-goal: f-1,f-78
0       propagate-goal: f-1,f-75
0       propagate-goal: f-1,f-73
0       propagate-goal: f-1,f-72
0       propagate-goal: f-1,f-70
0       propagate-goal: f-1,f-69
0       propagate-goal: f-1,f-58
0       propagate-goal: f-1,f-55
```

事实被插入到综合数据库后，CLIPS 系统会用之前的规则和事实进行匹配，将满足条件的规则激发并加入议程。因为被激发的规则较多，会在报告最后以附录的形式列举，在这里仅简要截图。

1.3 Run

CLIPS 系统按顺序执行议程中被激活的规则(注意:此时所有被激活的都是 propagate—goal 规则，因而优先级相同，系统顺序执行)。接下来，对重点的运行步骤进行讲解:



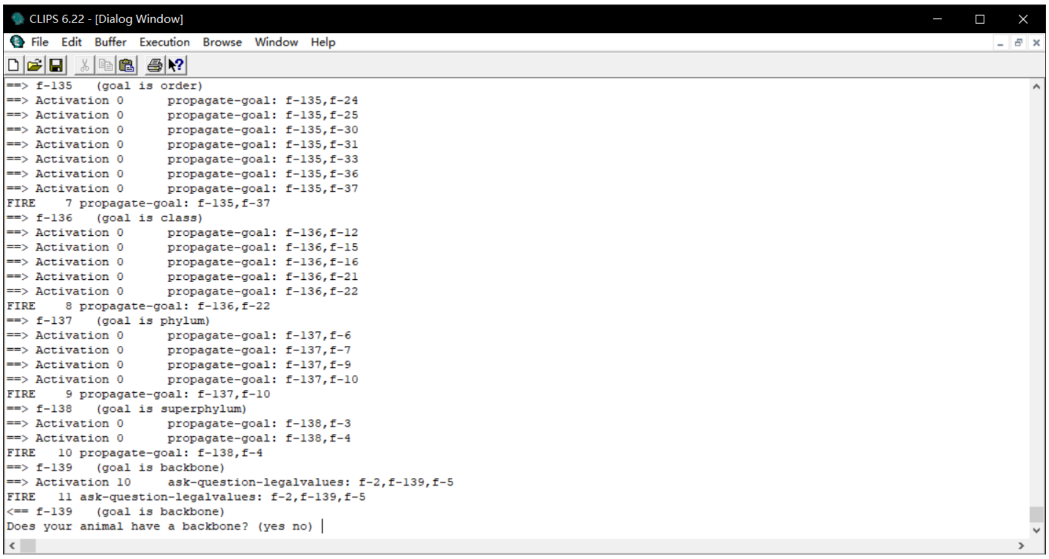
```
CLIPS> (run)
FIRE    1 propagate-goal: f-1,f-127
==> f-130   (goal is subsubspecies)
==> Activation 0      propagate-goal: f-130,f-124
FIRE    2 propagate-goal: f-130,f-124
==> f-131   (goal is subspecies)
==> Activation 0      propagate-goal: f-131,f-100
==> Activation 0      propagate-goal: f-131,f-109
FIRE    3 propagate-goal: f-131,f-109
==> f-132   (goal is species)
==> Activation 0      propagate-goal: f-132,f-79
==> Activation 0      propagate-goal: f-132,f-81
==> Activation 0      propagate-goal: f-132,f-82

==> Activation 0      propagate-goal: f-134,f-54
==> Activation 0      propagate-goal: f-134,f-57
FIRE    6 propagate-goal: f-134,f-57
==> f-135   (goal is order)
==> Activation 0      propagate-goal: f-135,f-24
==> Activation 0      propagate-goal: f-135,f-25
==> Activation 0      propagate-goal: f-135,f-30
==> Activation 0      propagate-goal: f-135,f-31
==> Activation 0      propagate-goal: f-135,f-33
==> Activation 0      propagate-goal: f-135,f-36
==> Activation 0      propagate-goal: f-135,f-37
FIRE    7 propagate-goal: f-135,f-37
==> f-136   (goal is class)
==> Activation 0      propagate-goal: f-136,f-12
==> Activation 0      propagate-goal: f-136,f-15
==> Activation 0      propagate-goal: f-136,f-16
==> Activation 0      propagate-goal: f-136,f-21
==> Activation 0      propagate-goal: f-136,f-22
FIRE    8 propagate-goal: f-136,f-22
==> f-137   (goal is phylum)
==> Activation 0      propagate-goal: f-137,f-6
==> Activation 0      propagate-goal: f-137,f-7
==> Activation 0      propagate-goal: f-137,f-9
==> Activation 0      propagate-goal: f-137,f-10
FIRE    9 propagate-goal: f-137,f-10
==> f-138   (goal is superphylum)
==> Activation 0      propagate-goal: f-138,f-3
==> Activation 0      propagate-goal: f-138,f-4
FIRE   10 propagate-goal: f-138,f-4
==> f-139   (goal is backbone)
==> Activation 10     ask-question-legalvalues: f-2,f-139,f-5
FIRE   11 ask-question-legalvalues: f-2,f-139,f-5
<== f-139   (goal is backbone)
Does your animal have a backbone? (yes no)
```

FIRE 关键字后是被激活的规则。根据 propagate—goal 的代码，事实 f—1 (goal is type.animal) 和 f—127 (rule (if subsubspecies is nofleece and domesticated is no) (then type.animal is deer/moose/antelope)) 会产生新的事实 f—130 (goal is subsubspecies)，并将其插入综合数据库。

当规则运行完毕，综合数据库更新，CLIPS 系统再次匹配规则和事实，激活新的规则 propagate—goal: f—130，f—124。事实 f—130 (goal is subsubspecies) 和 f—124 (rule (if subspecies is nohorn and fleece is no) (then subsubspecies is nofleece)) 会产生新的事实 f—131 (goal is subspecies)，并将其插入综合数据库。

之后的执行步骤大致相同，直到事实 f—139 (goal is backbone) 插入综合数据库。CLIPS 系统再次进行规则和事实的匹配，ask—question—legalvalues 被激活:



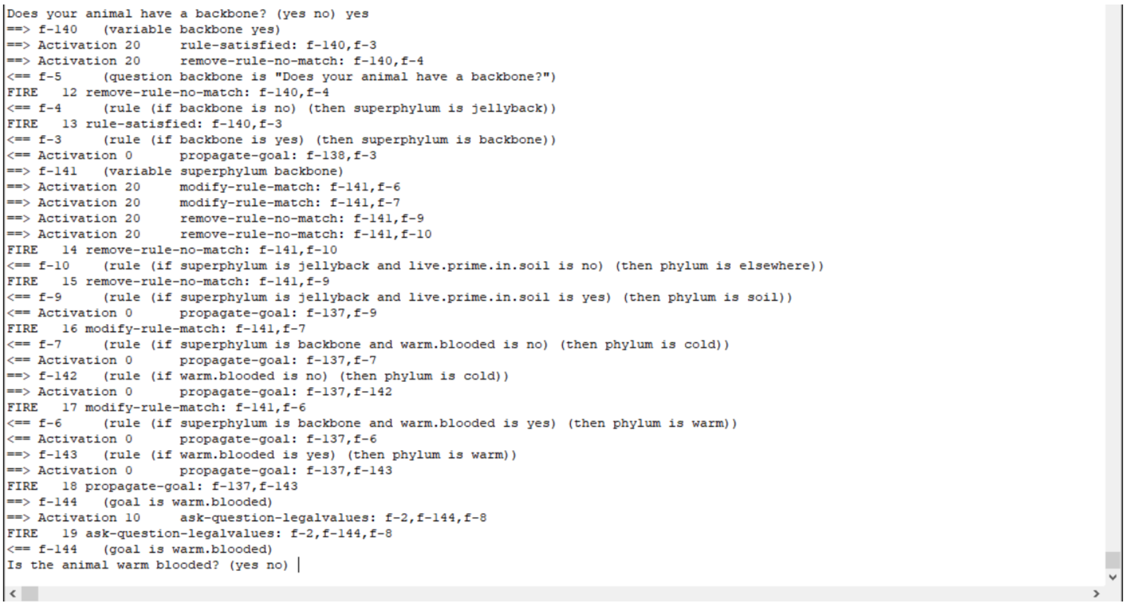
```
==> f-135   (goal is order)
==> Activation 0      propagate-goal: f-135,f-24
==> Activation 0      propagate-goal: f-135,f-25
==> Activation 0      propagate-goal: f-135,f-30
==> Activation 0      propagate-goal: f-135,f-31
==> Activation 0      propagate-goal: f-135,f-33
==> Activation 0      propagate-goal: f-135,f-36
==> Activation 0      propagate-goal: f-135,f-37
FIRE    7 propagate-goal: f-135,f-37
==> f-136   (goal is class)
==> Activation 0      propagate-goal: f-136,f-12
==> Activation 0      propagate-goal: f-136,f-15
==> Activation 0      propagate-goal: f-136,f-16
==> Activation 0      propagate-goal: f-136,f-21
==> Activation 0      propagate-goal: f-136,f-22
FIRE    8 propagate-goal: f-136,f-22
==> f-137   (goal is phylum)
==> Activation 0      propagate-goal: f-137,f-6
==> Activation 0      propagate-goal: f-137,f-7
==> Activation 0      propagate-goal: f-137,f-9
==> Activation 0      propagate-goal: f-137,f-10
FIRE    9 propagate-goal: f-137,f-10
==> f-138   (goal is superphylum)
==> Activation 0      propagate-goal: f-138,f-3
==> Activation 0      propagate-goal: f-138,f-4
FIRE   10 propagate-goal: f-138,f-4
==> f-139   (goal is backbone)
==> Activation 10     ask-question-legalvalues: f-2,f-139,f-5
FIRE   11 ask-question-legalvalues: f-2,f-139,f-5
<== f-139   (goal is backbone)
Does your animal have a backbone? (yes no) |
```

根据 ask—question—legalvalues 代码，f—139 和 f—5 的事实索引约束到变量?f1 和?f2 中，?variable＝

backbone，?text = Does your animal have a backbone?并打印输出。用户输入的单字段会被约束到变量?reply 中。

1.4 回答问题 Does your animal have a backbone?

以 yes 为例：



```
Does your animal have a backbone? (yes no) yes
==> f-140    (variable backbone yes)
==> Activation 20      rule-satisfied: f-140,f-3
==> Activation 20      remove-rule-no-match: f-140,f-4
<== f-5      (question backbone is "Does your animal have a backbone?")
FIRE    12 remove-rule-no-match: f-140,f-4
<== f-4      (rule (if backbone is no) (then superphylum is jellyback))
FIRE    13 rule-satisfied: f-140,f-3
<== f-3      (rule (if backbone is yes) (then superphylum is backbone))
<== Activation 0       propagate-goal: f-138,f-3
==> f-141    (variable superphylum backbone)
==> Activation 20      modify-rule-match: f-141,f-6
==> Activation 20      modify-rule-match: f-141,f-7
==> Activation 20      remove-rule-no-match: f-141,f-9
==> Activation 20      remove-rule-no-match: f-141,f-10
FIRE    14 remove-rule-no-match: f-141,f-10
<== f-10     (rule (if superphylum is jellyback and live.prime.in.soil is no) (then phylum is elsewhere))
FIRE    15 remove-rule-no-match: f-141,f-9
<== f-9      (rule (if superphylum is jellyback and live.prime.in.soil is yes) (then phylum is soil))
<== Activation 0       propagate-goal: f-137,f-9
FIRE    16 modify-rule-match: f-141,f-7
<== f-7      (rule (if superphylum is backbone and warm.blooded is no) (then phylum is cold))
<== Activation 0       propagate-goal: f-137,f-7
==> f-142    (rule (if warm.blooded is no) (then phylum is cold))
==> Activation 0       propagate-goal: f-137,f-142
FIRE    17 modify-rule-match: f-141,f-6
<== f-6      (rule (if superphylum is backbone and warm.blooded is yes) (then phylum is warm))
<== Activation 0       propagate-goal: f-137,f-6
==> f-143    (rule (if warm.blooded is yes) (then phylum is warm))
==> Activation 0       propagate-goal: f-137,f-143
FIRE    18 propagate-goal: f-137,f-143
==> f-144    (goal is warm.blooded)
==> Activation 10      ask-question-legalvalues: f-2,f-144,f-8
FIRE    19 ask-question-legalvalues: f-2,f-144,f-8
<== f-144    (goal is warm.blooded)
Is the animal warm blooded? (yes no) |
```

根据 ask-question-legalvalues 代码，经过 member 关键字和 if-then 语句的判断，将在综合数据库中插入新的事实 f-140 (variable backbone yes)，并撤销事实 f-5。

综合数据库更新，CLIPS 系统进行新一轮匹配，rule-satisfied：f-140，f-3 和 remove-rule-no-match：f-140，f-4 被激活。

规则 remove-rule-no-match 移除原综合数据库中不符合选项的事实 f-4 (rule (if backbone is no) (then superphylum is jellyback))。因为没有新的事实增加，继续执行议程中激活的规则。

规则 rule-satisfied 撤销原综合数据库中的事实 f-3，并加入 f-141(variable superphylum backbone)。因为 f-3 的撤销，议程中已激活的规则 propagate-goal：f-138，f-3 失效，移出议程。

新事实 f-141 的插入使得专家系统进行新的匹配。从而导致新的规则被激活，并压入议程：remove-rule-no-match：f-141，f-10 和 remove-rule-no-match：f-141，f-9。同时，因事实的撤销，原激活规则 propagate-goal：f-137，f-9 被移除。

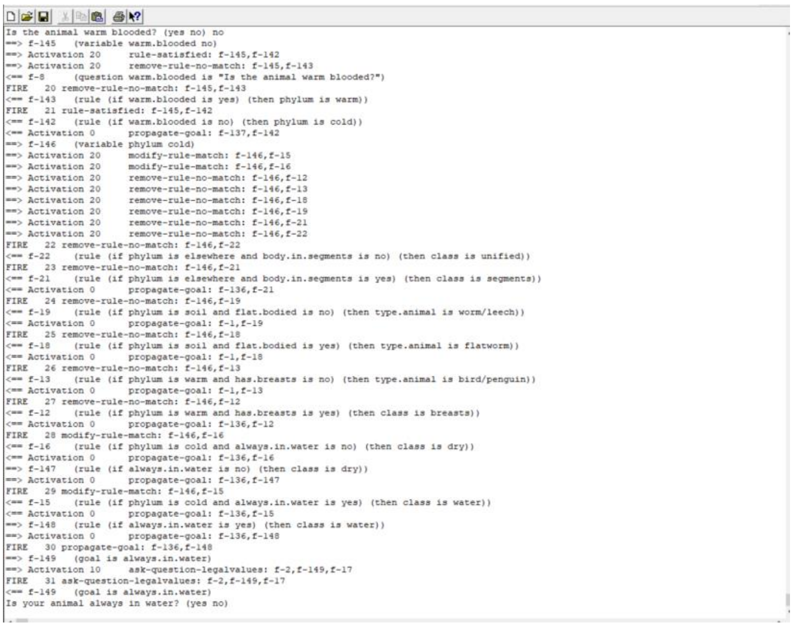
规则 modify-rule-match：f-141，f-7 将 f-7 撤销，并加入新的事实 f-142 (rule (if warm.blooded is no) (then phylum is cold))。在此过程中，原激活规则 propagate-goal：f-137，f-7 被移除。

规则 modify-rule-match：f-141，f-6 将 f-6 撤销，并加入新的事实 f-143 (rule (if warm.blooded is yes) (then phylum is warm))。在此过程中，原激活规则 propagate-goal：f-137，f-6 被移除。

因为新的事实加入，规则 propagate-goal：f-137，f-143 被激活并执行，直到新的事实 f-144 (goal is warm.blooded) 插入，从而激活规则 ask-question-legalvalues：f-2，f-144，f-8。

1.5 回答问题 Is the animal warm blooded?

以 no 为例：



```
Is the animal warm blooded? (yes no) no
==> f-145    (variable warm.blooded no)
==> Activation 20     rule-satisfied: f-145,f-142
==> Activation 20     remove-rule-no-match: f-145,f-143
<== f-8      (question warm.blooded is "Is the animal warm blooded?")
FIRE   20 remove-rule-no-match: f-145,f-143
<== f-143    (rule (if warm.blooded is yes) (then phylum is warm))
FIRE   21 rule-satisfied: f-145,f-142
<== f-142    (rule (if warm.blooded is no) (then phylum is cold))
<== Activation 0      propagate-goal: f-137,f-142
==> f-146    (variable phylum cold)
==> Activation 20     modify-rule-match: f-146,f-15
==> Activation 20     modify-rule-match: f-146,f-16
==> Activation 20     remove-rule-no-match: f-146,f-12
==> Activation 20     remove-rule-no-match: f-146,f-13
==> Activation 20     remove-rule-no-match: f-146,f-18
==> Activation 20     remove-rule-no-match: f-146,f-19
==> Activation 20     remove-rule-no-match: f-146,f-21
==> Activation 20     remove-rule-no-match: f-146,f-22
FIRE   22 remove-rule-no-match: f-146,f-22
<== f-22     (rule (if phylum is elsewhere and body.in.segments is no) (then class is unified))
FIRE   23 remove-rule-no-match: f-146,f-21
<== f-21     (rule (if phylum is elsewhere and body.in.segments is yes) (then class is segments))
<== Activation 0      propagate-goal: f-136,f-21
FIRE   24 remove-rule-no-match: f-146,f-19
<== f-19     (rule (if phylum is soil and flat.bodied is no) (then type.animal is worm/leech))
<== Activation 0      propagate-goal: f-1,f-19
FIRE   25 remove-rule-no-match: f-146,f-18
<== f-18     (rule (if phylum is soil and flat.bodied is yes) (then type.animal is flatworm))
<== Activation 0      propagate-goal: f-1,f-18
FIRE   26 remove-rule-no-match: f-146,f-13
<== f-13     (rule (if phylum is warm and has.breasts is no) (then type.animal is bird/penguin))
<== Activation 0      propagate-goal: f-1,f-13
FIRE   27 remove-rule-no-match: f-146,f-12
<== f-12     (rule (if phylum is warm and has.breasts is yes) (then class is breasts))
<== Activation 0      propagate-goal: f-136,f-12
FIRE   28 modify-rule-match: f-146,f-16
<== f-16     (rule (if phylum is cold and always.in.water is no) (then class is dry))
<== Activation 0      propagate-goal: f-136,f-16
==> f-147    (rule (if always.in.water is no) (then class is dry))
==> Activation 0      propagate-goal: f-136,f-147
FIRE   29 modify-rule-match: f-146,f-15
<== f-15     (rule (if phylum is cold and always.in.water is yes) (then class is water))
<== Activation 0      propagate-goal: f-136,f-15
==> f-148    (rule (if always.in.water is yes) (then class is water))
==> Activation 0      propagate-goal: f-136,f-148
FIRE   30 propagate-goal: f-136,f-148
==> f-149    (goal is always.in.water)
==> Activation 10     ask-question-legalvalues: f-2,f-149,f-17
FIRE   31 ask-question-legalvalues: f-2,f-149,f-17
<== f-149    (goal is always.in.water)
Is your animal always in water? (yes no)
```

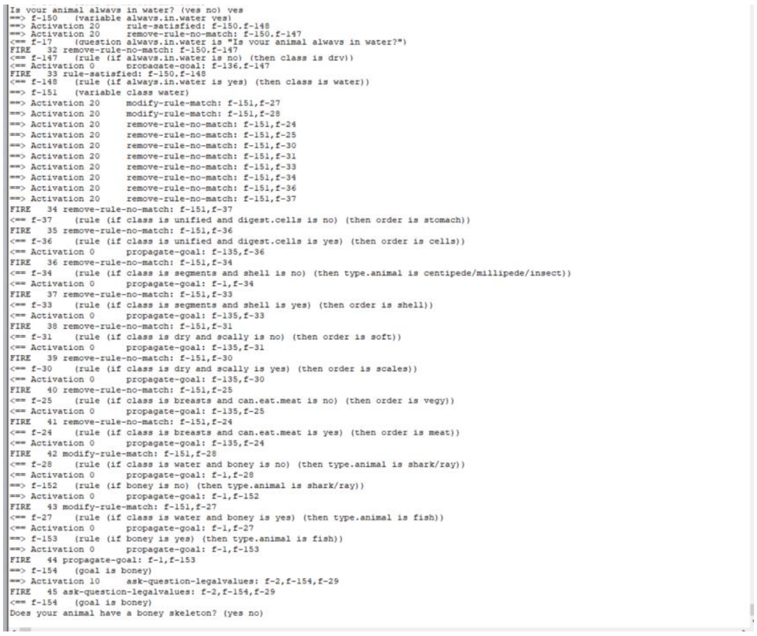
系统运行原理与之前大体相同，因而简要说明：

新的事实 f—145（variable warm.blooded no）加入，从而激活 rule—satisfied：f—145，f—142 和 remove—rule—no—match：f—145，f—143，并顺序执行。同时，撤销不符合要求的事实 f—143，新增事实 f—146（variable phylum cold）。根据 f—146，激活 6 个有关 remove—rule—no—match 的规则，并顺序执行，分别撤销事实 f—12，f—13，f—18，f—19，f—21 和 f—22；激活 2 个有关 modify—rule0match 的规则；撤销包含 phylum cold 字段值的事实 f—15 和 f—16，并取出它们 if 槽值中的字段值生成新的事实 f—147（rule （if always.in.water is no）（then class is dry））和 f—148（rule （if always.in.water is yes）（then class is water））。

系统根据新加入的事实 f—148 和已存在的事实 f—136 激活规则 propagate—goal：f—136，f—148，并生成事实 f—149（goal is always.in.water），从而激活并执行 ask—question—legalvalues：f—2，f—149，f—17。

1.6 回答问题 Is your animal always in water?

以 yes 为例：



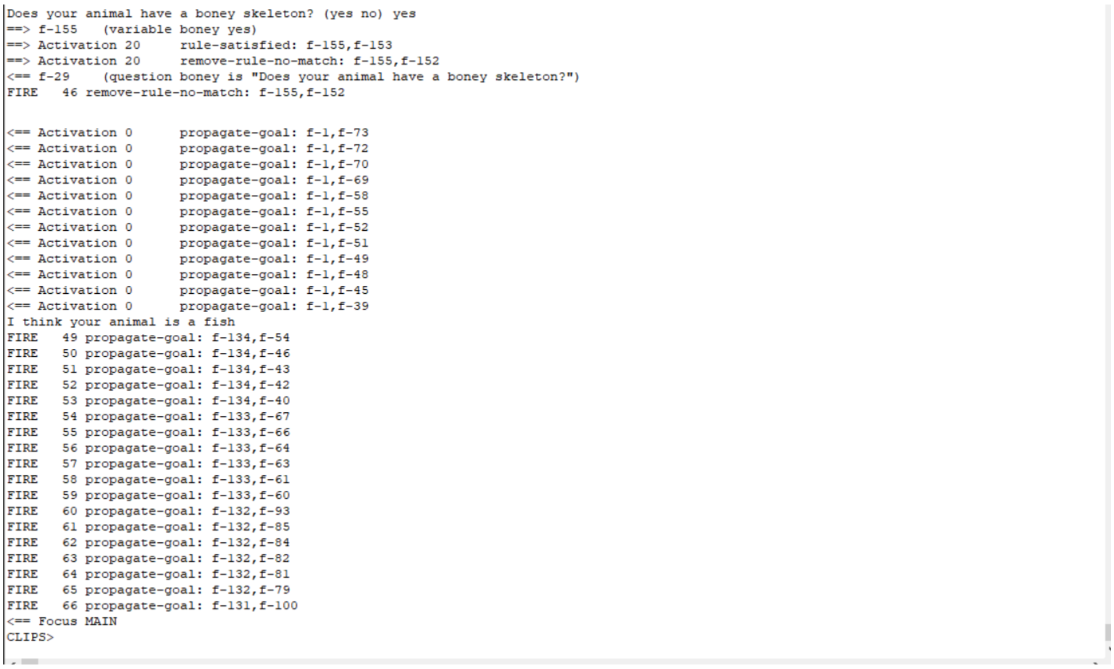
```
Is your animal always in water? (yes no) yes
==> f-150    (variable always.in.water yes)
==> Activation 20     rule-satisfied: f-150,f-148
==> Activation 20     remove-rule-no-match: f-150,f-147
<== f-17     (question always.in.water is "Is your animal always in water?")
FIRE   32 remove-rule-no-match: f-150,f-147
<== f-147    (rule (if always.in.water is no) (then class is dry))
<== Activation 0      propagate-goal: f-136,f-147
FIRE   33 rule-satisfied: f-150,f-148
<== f-148    (rule (if always.in.water is yes) (then class is water))
==> f-151    (variable class water)
==> Activation 20     modify-rule-match: f-151,f-27
==> Activation 20     modify-rule-match: f-151,f-28
==> Activation 20     remove-rule-no-match: f-151,f-24
==> Activation 20     remove-rule-no-match: f-151,f-25
==> Activation 20     remove-rule-no-match: f-151,f-30
==> Activation 20     remove-rule-no-match: f-151,f-31
==> Activation 20     remove-rule-no-match: f-151,f-33
==> Activation 20     remove-rule-no-match: f-151,f-34
==> Activation 20     remove-rule-no-match: f-151,f-36
==> Activation 20     remove-rule-no-match: f-151,f-37
FIRE   34 remove-rule-no-match: f-151,f-37
<== f-37     (rule (if class is unified and digest.cells is no) (then order is stomach))
FIRE   35 remove-rule-no-match: f-151,f-36
<== f-36     (rule (if class is unified and digest.cells is yes) (then order is cells))
<== Activation 0      propagate-goal: f-135,f-36
FIRE   36 remove-rule-no-match: f-151,f-34
<== f-34     (rule (if class is segments and shell is no) (then type.animal is centipede/millipede/insect))
<== Activation 0      propagate-goal: f-1,f-34
FIRE   37 remove-rule-no-match: f-151,f-33
<== f-33     (rule (if class is segments and shell is yes) (then order is shell))
<== Activation 0      propagate-goal: f-135,f-33
FIRE   38 remove-rule-no-match: f-151,f-31
<== f-31     (rule (if class is dry and scally is no) (then order is soft))
<== Activation 0      propagate-goal: f-135,f-31
FIRE   39 remove-rule-no-match: f-151,f-30
<== f-30     (rule (if class is dry and scally is yes) (then order is scales))
<== Activation 0      propagate-goal: f-135,f-30
FIRE   40 remove-rule-no-match: f-151,f-25
<== f-25     (rule (if class is breasts and can.eat.meat is no) (then order is vegy))
<== Activation 0      propagate-goal: f-135,f-25
FIRE   41 remove-rule-no-match: f-151,f-24
<== f-24     (rule (if class is breasts and can.eat.meat is yes) (then order is meat))
<== Activation 0      propagate-goal: f-135,f-24
FIRE   42 modify-rule-match: f-151,f-28
<== f-28     (rule (if class is water and boney is no) (then type.animal is shark/ray))
<== Activation 0      propagate-goal: f-1,f-28
==> f-152    (rule (if boney is no) (then type.animal is shark/ray))
==> Activation 0      propagate-goal: f-1,f-152
FIRE   43 modify-rule-match: f-151,f-27
<== f-27     (rule (if class is water and boney is yes) (then type.animal is fish))
<== Activation 0      propagate-goal: f-1,f-27
==> f-153    (rule (if boney is yes) (then type.animal is fish))
==> Activation 0      propagate-goal: f-1,f-153
FIRE   44 propagate-goal: f-1,f-153
==> f-154    (goal is boney)
==> Activation 10     ask-question-legalvalues: f-2,f-154,f-29
FIRE   45 ask-question-legalvalues: f-2,f-154,f-29
<== f-154    (goal is boney)
Does your animal have a boney skeleton? (yes no)
```

新的事实 f-150（variable always.in.water yes）加入，从而激活 rule-satisfied：f-150，f-148 和 remove-rule-no-match：f-150，f-147，并顺序执行。同时，撤销不符合要求的事实 f-147，新增事实 f-151（variable class water）。根据 f-151 激活 8 个有关 remove-rule-no-match 的规则，并顺序执行，分别撤销事实 f-24，f-25，f-30，f-31，f-33，f-34，f-36 和 f-37；激活 2 个有关 modify-rule-match 的规则；撤销包含 class water 字段值的事实 f-27 和 f-28，并取出它们 if 槽值中的字段值生成新的事实 f-153 (rule (if boney is no) (then type.animal is fish))。

系统根据新加入的事实 f-153 和已存在的 f-1，激活规则 propagate-goal：f-153，f-1，并生成事实 f-154（goal is boney），从而激活并执行 ask-question-legalvalues：f-2，f-154，f-29。

1.7 回答问题 Does your animal have a boney skeleton?

以 yes 为例：



```
Does your animal have a boney skeleton? (yes no) yes
==> f-155    (variable boney yes)
==> Activation 20     rule-satisfied: f-155,f-153
==> Activation 20     remove-rule-no-match: f-155,f-152
<== f-29     (question boney is "Does your animal have a boney skeleton?")
FIRE   46 remove-rule-no-match: f-155,f-152


<== Activation 0      propagate-goal: f-1,f-73
<== Activation 0      propagate-goal: f-1,f-72
<== Activation 0      propagate-goal: f-1,f-70
<== Activation 0      propagate-goal: f-1,f-69
<== Activation 0      propagate-goal: f-1,f-58
<== Activation 0      propagate-goal: f-1,f-55
<== Activation 0      propagate-goal: f-1,f-52
<== Activation 0      propagate-goal: f-1,f-51
<== Activation 0      propagate-goal: f-1,f-49
<== Activation 0      propagate-goal: f-1,f-48
<== Activation 0      propagate-goal: f-1,f-45
<== Activation 0      propagate-goal: f-1,f-39
I think your animal is a fish
FIRE   49 propagate-goal: f-134,f-54
FIRE   50 propagate-goal: f-134,f-46
FIRE   51 propagate-goal: f-134,f-43
FIRE   52 propagate-goal: f-134,f-42
FIRE   53 propagate-goal: f-134,f-40
FIRE   54 propagate-goal: f-133,f-67
FIRE   55 propagate-goal: f-133,f-66
FIRE   56 propagate-goal: f-133,f-64
FIRE   57 propagate-goal: f-133,f-63
FIRE   58 propagate-goal: f-133,f-61
FIRE   59 propagate-goal: f-133,f-60
FIRE   60 propagate-goal: f-132,f-93
FIRE   61 propagate-goal: f-132,f-85
FIRE   62 propagate-goal: f-132,f-84
FIRE   63 propagate-goal: f-132,f-82
FIRE   64 propagate-goal: f-132,f-81
FIRE   65 propagate-goal: f-132,f-79
FIRE   66 propagate-goal: f-131,f-100
<== Focus MAIN
CLIPS>
```

新的事实 f-155（variable boney yes）加入，从而激活 rule-satisfied：f-155，f-153 和 remove-rule-no-match：f-155，f-152，并顺序执行。同时，撤销不符合要求的事实 f-152，新增事实 f-156（variable type.animal fish）。根据 f-156 激活 goal-satisfied：f-1，f-156，f-129，根据 goal-satisfied 的代码，系统撤销了 f-1 及原议程中包含 f-1 的所有激活规则，并给出结论 I think your animal is a fish。之后，议程中其他激活规则会依次执行，但因所生成的事实与原综合数据库中的重复，不再插入。至此，推理完成。

2. 实验心得

在本次实验中，遇到的主要问题有以下三点：

1） CLIPS 环境。双击可执行文件后，对于该系统如何运行一头雾水。经老师演示和查阅资料，熟悉了 CLIPS 的操作，并通过设置多个监控条项，辅助理解推理分析过程；

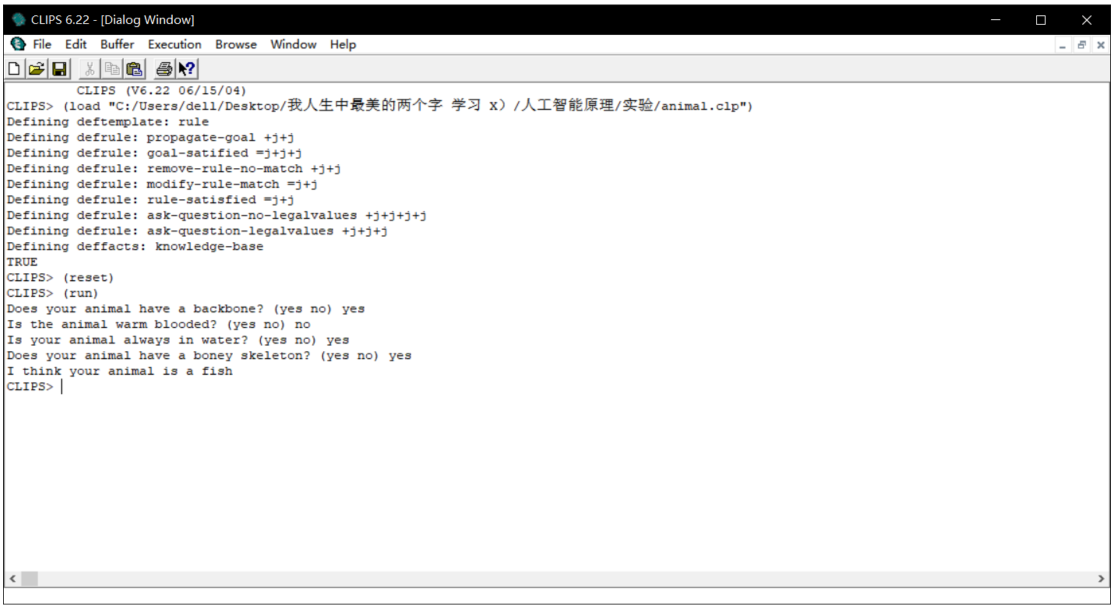
2） 符号含义。仔细阅读网上教程，了解了众多符号及关键词的含义，如=>、<=、FIRE 等，从而对该系统形成初步认识；

3） 知识库的局限性。该动物识别专家系统受知识数量的限制，所能推理出的动物种类极少；

实践出真知，本次 CLIPS 专家系统实验是对课堂和书本所学知识的补充，陌生的环境及规则和知识表达方式均增加了该实验的难度。通过自己逐步对推理过程进行分析，加深了对专家系统的理解和记忆，收获颇丰。

三、 附录

1. 完整运行界面


CLIPS 6.22 - [Dialog Window]
File Edit Buffer Execution Browse Window Help

```
         CLIPS (V6.22 06/15/04)
CLIPS> (load "C:/Users/dell/Desktop/我人生中最美的两个字 学习 x) /人工智能原理/实验/animal.clp")
Defining deftemplate: rule
Defining defrule: propagate-goal +j+j
Defining defrule: goal-satified =j+j+j
Defining defrule: remove-rule-no-match +j+j
Defining defrule: modify-rule-match =j+j
Defining defrule: rule-satisfied =j+j
Defining defrule: ask-question-no-legalvalues +j+j+j+j
Defining defrule: ask-question-legalvalues +j+j+j
Defining deffacts: knowledge-base
TRUE
CLIPS> (reset)
CLIPS> (run)
Does your animal have a backbone? (yes no) yes
Is the animal warm blooded? (yes no) no
Is your animal always in water? (yes no) yes
Does your animal have a boney skeleton? (yes no) yes
I think your animal is a fish
CLIPS> 
```

2. Reset 后，FACTS 中的事实清单

```
f-0     (initial-fact)
f-1     (goal is type.animal)
f-2     (legalanswers are yes no)
f-3     (rule (if backbone is yes) (then superphylum is backbone))
f-4     (rule (if backbone is no) (then superphylum is jellyback))
f-5     (question backbone is "Does your animal have a backbone?")
f-6     (rule (if superphylum is backbone and warm.blooded is yes) (then phylum is warm))
f-7     (rule (if superphylum is backbone and warm.blooded is no) (then phylum is cold))
f-8     (question warm.blooded is "Is the animal warm blooded?")
f-9     (rule (if superphylum is jellyback and live.prime.in.soil is yes) (then phylum is soil))
f-10    (rule (if superphylum is jellyback and live.prime.in.soil is no) (then phylum is elsewhere))
f-11    (question live.prime.in.soil is "Does your animal live primarily in soil?")
f-12    (rule (if phylum is warm and has.breasts is yes) (then class is breasts))
f-13    (rule (if phylum is warm and has.breasts is no) (then type.animal is bird/penguin))
f-14    (question has.breasts is "Normally, does the female of your animal nurse its young with
```

milk?")

f–15 (rule (if phylum is cold and always.in.water is yes) (then class is water))

f–16 (rule (if phylum is cold and always.in.water is no) (then class is dry))

f–17 (question always.in.water is "Is your animal always in water?")

f–18 (rule (if phylum is soil and flat.bodied is yes) (then type.animal is flatworm))

f–19 (rule (if phylum is soil and flat.bodied is no) (then type.animal is worm/leech))

f–20 (question flat.bodied is "Does your animal have a flat body?")

f–21 (rule (if phylum is elsewhere and body.in.segments is yes) (then class is segments))

f–22 (rule (if phylum is elsewhere and body.in.segments is no) (then class is unified))

f–23 (question body.in.segments is "Is the animals body in segments?")

f–24 (rule (if class is breasts and can.eat.meat is yes) (then order is meat))

f–25 (rule (if class is breasts and can.eat.meat is no) (then order is vegy))

f–26 (question can.eat.meat is "Does your animal eat red meat?")

f–27 (rule (if class is water and boney is yes) (then type.animal is fish))

f–28 (rule (if class is water and boney is no) (then type.animal is shark/ray))

f–29 (question boney is "Does your animal have a boney skeleton?")

f–30 (rule (if class is dry and scally is yes) (then order is scales))

f–31 (rule (if class is dry and scally is no) (then order is soft))

f–32 (question scally is "Is your animal covered with scaled skin?")

f–33 (rule (if class is segments and shell is yes) (then order is shell))

f–34 (rule (if class is segments and shell is no) (then type.animal is centipede/millipede/insect))

f–35 (question shell is "Does your animal have a shell?")

f–36 (rule (if class is unified and digest.cells is yes) (then order is cells))

f–37 (rule (if class is unified and digest.cells is no) (then order is stomach))

f–38 (question digest.cells is "Does your animal use many cells to digest it's food instead of a stomach?")

f–39 (rule (if order is meat and fly is yes) (then type.animal is bat))

f–40 (rule (if order is meat and fly is no) (then family is nowings))

f–41 (question fly is "Can your animal fly?")

f–42 (rule (if order is vegy and hooves is yes) (then family is hooves))

f–43 (rule (if order is vegy and hooves is no) (then family is feet))

f–44 (question hooves is "Does your animal have hooves?")

f–45 (rule (if order is scales and rounded.shell is yes) (then type.animal is turtle))

f–46 (rule (if order is scales and rounded.shell is no) (then family is noshell))

f–47 (question rounded.shell is "Does the animal have a rounded shell?")

f–48 (rule (if order is soft and jump is yes) (then type.animal is frog))

f–49 (rule (if order is soft and jump is no) (then type.animal is salamander))

f–50 (question jump is "Does your animal jump?")

f–51 (rule (if order is shell and tail is yes) (then type.animal is lobster))

f–52 (rule (if order is shell and tail is no) (then type.animal is crab))

f–53 (question tail is "Does your animal have a tail?")

f–54 (rule (if order is cells and stationary is yes) (then family is stationary))

f–55 (rule (if order is cells and stationary is no) (then type.animal is jellyfish))

f–56 (question stationary is "Is your animal attached permanently to an object?")

```
f-57    (rule (if order is stomach and multicelled is yes) (then family is multicelled))
f-58    (rule (if order is stomach and multicelled is no) (then type.animal is protozoa))
f-59    (question multicelled is "Is your animal made up of more than one cell?")
f-60    (rule (if family is nowings and opposing.thumb is yes) (then genus is thumb))
f-61    (rule (if family is nowings and opposing.thumb is no) (then genus is nothumb))
f-62    (question opposing.thumb is "Does your animal have an opposing thumb?")
f-63    (rule (if family is hooves and two.toes is yes) (then genus is twotoes))
f-64    (rule (if family is hooves and two.toes is no) (then genus is onetoe))
f-65    (question two.toes is "Does your animal stand on two toes/hooves per foot?")
f-66    (rule (if family is feet and live.in.water is yes) (then genus is water))
f-67    (rule (if family is feet and live.in.water is no) (then genus is dry))
f-68    (question live.in.water is "Does your animal live in water?")
f-69    (rule (if family is noshell and limbs is yes) (then type.animal is crocodile/alligator))
f-70    (rule (if family is noshell and limbs is no) (then type.animal is snake))
f-71    (question limbs is "Does your animal have limbs?")
f-72    (rule (if family is stationary and spikes is yes) (then type.animal is sea.anemone))
f-73    (rule (if family is stationary and spikes is no) (then type.animal is coral/sponge))
f-74    (question spikes is "Does your animal normally have spikes radiating from it's body?")
f-75    (rule (if family is multicelled and spiral.shell is yes) (then type.animal is snail))
f-76    (rule (if family is multicelled and spiral.shell is no) (then genus is noshell))
f-77    (question spiral.shell is "Does your animal have a spiral-shaped shell?")
f-78    (rule (if genus is thumb and prehensile.tail is yes) (then type.animal is monkey))
f-79    (rule (if genus is thumb and prehensile.tail is no) (then species is notail))
f-80    (question prehensile.tail is "Does your animal have a prehensile tail?")
f-81    (rule (if genus is nothumb and over.400 is yes) (then species is 400))
f-82    (rule (if genus is nothumb and over.400 is no) (then species is under400))
f-83    (question over.400 is "Does an adult normally weigh over 400 pounds?")
f-84    (rule (if genus is twotoes and horns is yes) (then species is horns))
f-85    (rule (if genus is twotoes and horns is no) (then species is nohorns))
f-86    (question horns is "Does your animal have horns?")
f-87    (rule (if genus is onetoe and plating is yes) (then type.animal is rhinoceros))
f-88    (rule (if genus is onetoe and plating is no) (then type.animal is horse/zebra))
f-89    (question plating is "Is your animal covered with a protective plating?")
f-90    (rule (if genus is water and hunted is yes) (then type.animal is whale))
f-91    (rule (if genus is water and hunted is no) (then type.animal is dolphin/porpoise))
f-92    (question hunted is "Is your animal, unfortunately, commercially hunted?")
f-93    (rule (if genus is dry and front.teeth is yes) (then species is teeth))
f-94    (rule (if genus is dry and front.teeth is no) (then species is noteeth))
f-95    (question front.teeth is "Does your animal have large front teeth?")
f-96    (rule (if genus is noshell and bivalve is yes) (then type.animal is clam/oyster))
f-97    (rule (if genus is noshell and bivalve is no) (then type.animal is squid/octopus))
f-98    (question bivalve is "Is your animal protected by two half-shells?")
f-99    (rule (if species is notail and nearly.hairless is yes) (then type.animal is man))
f-100   (rule (if species is notail and nearly.hairless is no) (then subspecies is hair))
```

f-101 (question nearly.hairless is "Is your animal nearly hairless?")

f-102 (rule (if species is 400 and land.based is yes) (then type.animal is bear/tiger/lion))

f-103 (rule (if species is 400 and land.based is no) (then type.animal is walrus))

f-104 (question land.based is "Is your animal land based?")

f-105 (rule (if species is under400 and thintail is yes) (then type.animal is cat))

f-106 (rule (if species is under400 and thintail is no) (then type.animal is coyote/wolf/fox/dog))

f-107 (question thintail is "Does your animal have a thin tail?")

f-108 (rule (if species is horns and one.horn is yes) (then type.animal is hippopotamus))

f-109 (rule (if species is horns and one.horn is no) (then subspecies is nohorn))

f-110 (question one.horn is "Does your animal have one horn?")

f-111 (rule (if species is nohorns and lives.in.desert is yes) (then type.animal is camel))

f-112 (rule (if species is nohorns and lives.in.desert is no) (then type.animal is giraffe))

f-113 (question lives.in.desert is "Does your animal normally live in the desert?")

f-114 (rule (if species is teeth and large.ears is yes) (then type.animal is rabbit))

f-115 (rule (if species is teeth and large.ears is no the type.animal is rat/mouse/squirrel/beaver/porcupine) (then))

f-116 (question large.ears is "Does your animal have large ears?")

f-117 (rule (if species is noteeth and pouch is yes) (then type.animal is "kangaroo/koala bear"))

f-118 (rule (if species is noteeth and pouch is no) (then type.animal is mole/shrew/elephant))

f-119 (question pouch is "Does your animal have a pouch?")

f-120 (rule (if subspecies is hair and long.powerful.arms is yes) (then type.animal is orangutan/gorilla/chimpanzie))

f-121 (rule (if subspecies is hair and long.powerful.arms is no) (then type.animal is baboon))

f-122 (question long.powerful.arms is "Does your animal have long, powerful arms?")

f-123 (rule (if subspecies is nohorn and fleece is yes) (then type.animal is sheep/goat))

f-124 (rule (if subspecies is nohorn and fleece is no) (then subsubspecies is nofleece))

f-125 (question fleece is "Does your animal have fleece?")

f-126 (rule (if subsubspecies is nofleece and domesticated is yes) (then type.animal is cow))

f-127 (rule (if subsubspecies is nofleece and domesticated is no) (then type.animal is deer/moose/antelope))

f-128 (question domesticated is "Is your animal domesticated?")

f-129 (answer is "I think your animal is a " type.animal)

For a total of 130 facts.

3．得出结论后，FACTS 中的事实清单

f-0 (initial-fact)

f-2 (legalanswers are yes no)

f-11 (question live.prime.in.soil is "Does your animal live primarily in soil?")

f-14 (question has.breasts is "Normally, does the female of your animal nurse its young with milk?")

f-20 (question flat.bodied is "Does your animal have a flat body?")

```
f-23    (question body.in.segments is "Is the animals body in segments?")
f-26    (question can.eat.meat is "Does your animal eat red meat?")
f-32    (question scally is "Is your animal covered with scaled skin?")
f-35    (question shell is "Does your animal have a shell?")
f-38    (question digest.cells is "Does your animal use many cells to digest it's food instead of a stomach?")
f-39    (rule (if order is meat and fly is yes) (then type.animal is bat))
f-40    (rule (if order is meat and fly is no) (then family is nowings))
f-41    (question fly is "Can your animal fly?")
f-42    (rule (if order is vegy and hooves is yes) (then family is hooves))
f-43    (rule (if order is vegy and hooves is no) (then family is feet))
f-44    (question hooves is "Does your animal have hooves?")
f-45    (rule (if order is scales and rounded.shell is yes) (then type.animal is turtle))
f-46    (rule (if order is scales and rounded.shell is no) (then family is noshell))
f-47    (question rounded.shell is "Does the animal have a rounded shell?")
f-48    (rule (if order is soft and jump is yes) (then type.animal is frog))
f-49    (rule (if order is soft and jump is no) (then type.animal is salamander))
f-50    (question jump is "Does your animal jump?")
f-51    (rule (if order is shell and tail is yes) (then type.animal is lobster))
f-52    (rule (if order is shell and tail is no) (then type.animal is crab))
f-53    (question tail is "Does your animal have a tail?")
f-54    (rule (if order is cells and stationary is yes) (then family is stationary))
f-55    (rule (if order is cells and stationary is no) (then type.animal is jellyfish))
f-56    (question stationary is "Is your animal attached permanently to an object?")
f-57    (rule (if order is stomach and multicelled is yes) (then family is multicelled))
f-58    (rule (if order is stomach and multicelled is no) (then type.animal is protozoa))
f-59    (question multicelled is "Is your animal made up of more than one cell?")
f-60    (rule (if family is nowings and opposing.thumb is yes) (then genus is thumb))
f-61    (rule (if family is nowings and opposing.thumb is no) (then genus is nothumb))
f-62    (question opposing.thumb is "Does your animal have an opposing thumb?")
f-63    (rule (if family is hooves and two.toes is yes) (then genus is twotoes))
f-64    (rule (if family is hooves and two.toes is no) (then genus is onetoe))
f-65    (question two.toes is "Does your animal stand on two toes/hooves per foot?")
f-66    (rule (if family is feet and live.in.water is yes) (then genus is water))
f-67    (rule (if family is feet and live.in.water is no) (then genus is dry))
f-68    (question live.in.water is "Does your animal live in water?")
f-69    (rule (if family is noshell and limbs is yes) (then type.animal is crocodile/alligator))
f-70    (rule (if family is noshell and limbs is no) (then type.animal is snake))
f-71    (question limbs is "Does your animal have limbs?")
f-72    (rule (if family is stationary and spikes is yes) (then type.animal is sea.anemone))
f-73    (rule (if family is stationary and spikes is no) (then type.animal is coral/sponge))
f-74    (question spikes is "Does your animal normally have spikes radiating from it's body?")
f-75    (rule (if family is multicelled and spiral.shell is yes) (then type.animal is snail))
f-76    (rule (if family is multicelled and spiral.shell is no) (then genus is noshell))
```

```
f-77    (question spiral.shell is "Does your animal have a spiral-shaped shell?")
f-78    (rule (if genus is thumb and prehensile.tail is yes) (then type.animal is monkey))
f-79    (rule (if genus is thumb and prehensile.tail is no) (then species is notail))
f-80    (question prehensile.tail is "Does your animal have a prehensile tail?")
f-81    (rule (if genus is nothumb and over.400 is yes) (then species is 400))
f-82    (rule (if genus is nothumb and over.400 is no) (then species is under400))
f-83    (question over.400 is "Does an adult normally weigh over 400 pounds?")
f-84    (rule (if genus is twotoes and horns is yes) (then species is horns))
f-85    (rule (if genus is twotoes and horns is no) (then species is nohorns))
f-86    (question horns is "Does your animal have horns?")
f-87    (rule (if genus is onetoe and plating is yes) (then type.animal is rhinoceros))
f-88    (rule (if genus is onetoe and plating is no) (then type.animal is horse/zebra))
f-89    (question plating is "Is your animal covered with a protective plating?")
f-90    (rule (if genus is water and hunted is yes) (then type.animal is whale))
f-91    (rule (if genus is water and hunted is no) (then type.animal is dolphin/porpoise))
f-92    (question hunted is "Is your animal, unfortunately, commercially hunted?")
f-93    (rule (if genus is dry and front.teeth is yes) (then species is teeth))
f-94    (rule (if genus is dry and front.teeth is no) (then species is noteeth))
f-95    (question front.teeth is "Does your animal have large front teeth?")
f-96    (rule (if genus is noshell and bivalve is yes) (then type.animal is clam/oyster))
f-97    (rule (if genus is noshell and bivalve is no) (then type.animal is squid/octopus))
f-98    (question bivalve is "Is your animal protected by two half-shells?")
f-99    (rule (if species is notail and nearly.hairless is yes) (then type.animal is man))
f-100   (rule (if species is notail and nearly.hairless is no) (then subspecies is hair))
f-101   (question nearly.hairless is "Is your animal nearly hairless?")
f-102   (rule (if species is 400 and land.based is yes) (then type.animal is bear/tiger/lion))
f-103   (rule (if species is 400 and land.based is no) (then type.animal is walrus))
f-104   (question land.based is "Is your animal land based?")
f-105   (rule (if species is under400 and thintail is yes) (then type.animal is cat))
f-106   (rule (if species is under400 and thintail is no) (then type.animal is coyote/wolf
/fox/dog))
f-107   (question thintail is "Does your animal have a thin tail?")
f-108   (rule (if species is horns and one.horn is yes) (then type.animal is hippopotamus))
f-109   (rule (if species is horns and one.horn is no) (then subspecies is nohorn))
f-110   (question one.horn is "Does your animal have one horn?")
f-111   (rule (if species is nohorns and lives.in.desert is yes) (then type.animal is camel))
f-112   (rule (if species is nohorns and lives.in.desert is no) (then type.animal is giraffe))
f-113   (question lives.in.desert is "Does your animal normally live in the desert?")
f-114   (rule (if species is teeth and large.ears is yes) (then type.animal is rabbit))
f-115   (rule (if species is teeth and large.ears is no the type.animal is rat/mouse/squirrel
/beaver/porcupine) (then))
f-116   (question large.ears is "Does your animal have large ears?")
f-117   (rule (if species is noteeth and pouch is yes) (then type.animal is "kangaroo/koala bear"))
f-118   (rule (if species is noteeth and pouch is no) (then type.animal is mole/shrew/elephant))
```

```
f-119   (question pouch is "Does your animal have a pouch?")
f-120   (rule (if subspecies is hair and long.powerful.arms is yes) (then type.animal is orangutan
/gorilla/chimpanzie))
f-121   (rule (if subspecies is hair and long.powerful.arms is no) (then type.animal is baboon))
f-122   (question long.powerful.arms is "Does your animal have long, powerful arms?")
f-123   (rule (if subspecies is nohorn and fleece is yes) (then type.animal is sheep/goat))
f-124   (rule (if subspecies is nohorn and fleece is no) (then subsubspecies is nofleece))
f-125   (question fleece is "Does your animal have fleece?")
f-126   (rule (if subsubspecies is nofleece and domesticated is yes) (then type.animal is cow))
f-127   (rule (if subsubspecies is nofleece and domesticated is no) (then type.animal is deer
/moose/antelope))
f-128   (question domesticated is "Is your animal domesticated?")
f-129   (answer is "I think your animal is a " type.animal)
f-130   (goal is subsubspecies)
f-131   (goal is subspecies)
f-132   (goal is species)
f-133   (goal is genus)
f-134   (goal is family)
f-135   (goal is order)
f-136   (goal is class)
f-137   (goal is phylum)
f-138   (goal is superphylum)
f-140   (variable backbone yes)
f-141   (variable superphylum backbone)
f-145   (variable warm.blooded no)
f-146   (variable phylum cold)
f-150   (variable always.in.water yes)
f-151   (variable class water)
f-155   (variable boney yes)
f-156   (variable type.animal fish)
For a total of 118 facts.
```