Programming Assignment 03
Deadline: November 24th 11:59pm

Instructions

Your goal in this programming assignment is to build an Expert System (ES) in Prolog. It is up to you to decide how you are going to structure your ES. However, we recommend that you use the framework discussed in Lesson 22 as a starting point. You are also free to choose any ES domain, except *animals* (since we are already using it as an example) . Your ES must be able to "learn" from the user using underline dynamic clauses (both positive and negative ones). Your ES should also have a minimal number of 15 distinct classes. For example, if your chosen domain is *diseases* your program should be able to diagnose at least 15 different *diseases*.

User Interface

Your ES begins when the user types "`begin.`" in the Prolog shell. The system should then welcome the user with some informative text. For example:

```
?- begin.
Welcome to the ES about animals!
I am going to ask questions about animal features.
Please answer yes. or no.
Ready?
```

Answering "`yes.`" initiates a sequence of yes/no questions that ends when the system is able to determine the *animal* or the system exhausted all the questions that it can ask. If the user answers "`no.`" to the "`Ready?`" question the program should terminate with "`Bye!`" Below is a run where the system was able to find an answer.

```
has fur?
|: no.
has feathers?
|: yes.
says quack?
|: yes.
I think your animal is a duck.
```

The system should then ask the user for feedback. Depending on how the user answers, the system should react differently. For example:

```
Did I get it right? |: yes.
Nice!

OR

Did I get it right? |: no.
Not my fault! My designer did not give me enough information
about animals.
```

Below is a run where the system was not able to find an answer after exhausting all possible questions available.

```
has fur?
|: yes.
says woof?
|: no.
says meow?
|: no.
has feathers?
|: no.
Hmmm, I could not figure this one out...
```

At the end, the ES should display a message explaining how to start a new session. For example:

```
To try again just type begin.
```

Please note that each ES session should be independent from previous ones. Therefore, you need to *retract all* of the dynamically added clauses before starting a new session.

Program Submission

You should submit your Prolog source code through Blackboard. If working with a partner, make sure you add both of your names in the comment section in your source code.

Rubric

TBD