Q&A Session for Programming Languages Lecture 7

Session Number: 1206497878
Date: 2020-9-25
Starting time: 14:23

_____

ANON — 14:48
Q: For question 6, do we count duplicate answers?
Priority: N/A
      Ana L. Milanova — 14:48
      A: You should count what Prolog would count.

_____

ANON — 14:49
Q: How is class going to proceed today with career fair?
Priority: N/A
      Ana L. Milanova — 14:50
      A: As usual with recording. We expect sparser Q&A but if you
have questions, bring them back on Tuesday.

_____

ANON — 15:04
Q: do you have advice on how to become more used to Proglog syntax?
Priority: N/A
      Konstantin Kuzmin — 15:06
      A: I would say just by trying solve as many exercises (from
simple to harder) as you can. It's like swimming, you just need to
dive in. ;)

_____

ANON — 15:08
Q: Can you show the pl file side by side with the terminal?
Priority: N/A
      Konstantin Kuzmin — 15:16
      A: We can make the pl file available, so that you can replay
the recording of this Q&A session and have the source code side-by-
side.
      Ana L. Milanova — 16:26
      A: I didn't get your request in real time while presenting,
but yes, you can split the window and show them next to each other. I
usually switch between windows when coding in Python or Prolgog to
have a larger screen but it's just preference.

_____

ANON — 15:12
Q: What does "[debug]." do in Prolog?
Priority: N/A
      Steven Haussmann — 15:13

A: It disables some optimizations that can cause it to lose information about errors (amongst other things, probably!). It shouldn't affect what your queries do.

_____

ANON — 15:14
Q: Just to be clear, I streamed all lecture videos and also attended lecture events this week. My checkin is complete correct, no need to visit office hours?
Priority: N/A
        Konstantin Kuzmin — 15:16
        A: Correct!

_____

ANON — 15:25
Q: could we overcome the failure of snowy(troy) by adding a new variable Y, s.t. snowy(X) :- rainy(Y), !, cold(Y), Y=X. ?
Priority: N/A
        Ana L. Milanova — 15:29
        A: No, I don't think this will work. The ! succeeds, which will prevent the interpreter from trying the other choice at the parent, the other choice being snowy(troy).

_____

ANON — 15:29
Q: If I attended live lecture, do I need to send in an email saying I attended? or will you just look at the automated return emails from webex?
Priority: N/A
        Konstantin Kuzmin — 15:30
        A: No need to send an email in this case. We get logs of students who attended live lectures directly from WebEx.

_____

ANON — 15:31
Q: if we add snowy(troy) to the end of example 4 or 5, both of them cannot output C = troy right?
Priority: N/A
        Ana L. Milanova — 15:34
        A: Correct.

_____

ANON — 15:31
Q: for this last example would there be no other answers printed (assuming there are other possible answers) since it's commited at the end now?
Priority: N/A
        Ana L. Milanova — 15:34
        A: Correct again.

_____

ANON — 15:32
Q: Is there a certain amount of time you have to present for the live lecture to count? I had to join a few minutes late because of the career fair but will have attended the majority of lecture.
Priority: N/A
        Konstantin Kuzmin — 15:36
        A: WebEx registers every time you join or leave, even if it is multiple times during the Event. For the purposes of attendance credit, it is sufficient to have at least one live lecture attendance record on WebEx, even if it is just one minute in duration.
_____

ANON — 15:35
Q: so essencially, once the program backtracks to an "!" the program terminates?
Priority: N/A
        Ana L. Milanova — 15:36
        A: No, not necessarily. It just commits the interpreter to the bindings that the program made since the unification of the parent of the cut (!).
        Ana L. Milanova — 15:38
        A: We may have a tree above the goal that has the Cut and Prolog will continue to search there, even as it has committed to those bindings due to the cut.
_____

ANON — 15:49
Q: Can I attend office hours after this lecture to get attendance?
Priority: N/A
        Ana L. Milanova — 15:51
        A: You are getting attendance by attending the lecture. Attendence is an OR: you can do either one of lecture attendance, streaming videos on Mediasite, or checking in in office hours.
_____

ANON — 15:52
Q: what is meant by negation is not the same in prolog as in logic? Isn't prolog logical by design?
Priority: N/A
        Ana L. Milanova — 15:53
        A: It does not implement full first order logic, because if it did inference would be undecidable. So it has restrictions and one of the restrictions is that it does not allow us to make negative assiertions.
        Ana L. Milanova — 15:53
        A: For example, we cannot assert in the database of facts something like not(food(jane)).
_____

ANON — 15:53
Q: is "_" considered a variable?
Priority: N/A
        Konstantin Kuzmin — 15:53
        A: In Prolog, "a single underscore (_) denotes an anonymous
variable and means "any term"" (from Wikipedia).
        Steven Haussmann — 16:03
        A: If you have a variable that's never actually used, Prolog
will warn you about it being a "singleton" —— i.e., it only shows up
once. That's where you want to use _; it takes up space, but has no
meaning.
        Ana L. Milanova — 16:05
        A: Yes. For example, if you write member(A,[A|B]). Prolog will
warn you about B being a singleton variable. To avoid that, you can
write member(A,[A|_]). meaning that we don't care what the tail of the
list is. The way we had it in n-Queens.
_____

ANON — 16:12
Q: can you show the liast slide on the parser again?
Priority: N/A
_____

ANON — 16:15
Q: So the closest thing to normal programming Prolog Functions are is
the "Recursive functions only challange?"
Priority: N/A
        Steven Haussmann — 16:16
        A: That does get you closer to Prolog's style, yes —— so those
familiar with functional languages will probably find it more
comfortable
_____

ANON — 16:23
Q: Can you explain a little more about invertibility?
Priority: N/A
        Ana L. Milanova — 16:30
        A: Yes, you can think of giving the result of the computation,
and prolog computing the arguments. For example, if you write
append(l1,l2) in other languages, it works in the forward way, i.e.,
it expects l1 and l2 to be fully bound values, and comutes
        Ana L. Milanova — 16:30
        A: the result of appending l2 onto l1. But we cannot give
append the result, and have it compute l1 and l2. In Prolog we can do
just that!
_____

ANON — 16:24
Q: Can I simply think of it as having variables on the rhs, which is
not allowed since variables can't self bind?

Priority: N/A

Ana L. Milanova — 16:32

A: I think yes, you can think of having this result (or out) argument/variable, which we can't give on the rhs, but must give as argument.

Ana L. Milanova — 16:33

A: And prolog computes the arguments based off of a fully isntantiated result variable. (Of course, not all predicates are invertible!)