

Re: negotiation by failure– list operations

Source: <http://coding.derkeiler.com/Archive/Prolog/comp.lang.prolog/2005-06/msg00072.html>

- *From:* "Nameless" <news.mail@xxxxxxxxx>
 - *Date:* Wed, 15 Jun 2005 05:14:23 +0200
-

"ne0" wrote in message

news:1118600315.819949.155550@xx

> hi, what im trying to do is the following:

>

>

> delete1(X,L1,L2) that uses delete(X,L1,L2) but if X isnt in L1

> it should not throw NO – it should only throw YES

>

> i think its donesimilar to that predicate:

> different(X,Y) :- X=Y, !, true.

> different(X,Y).

>

> but how do i change my delete1 predicate that it throws YES if

> X isnt in L1 ?

>

> Can someone help me ?

Isn't this just a question of supplying a 'catch-all'
clause to delete1/3?

delete1(X,L1,L2) :-

delete(X,L1,L2).

delete1(,_,_).

Now delete1/3 will always succeed, but whether it 'throws'
a YES may well depend on the Prolog system in use.

—

Mail sent to this email address is deleted unread
on the server. Please send replies to the newsgroup.

.

-
- *Follow-Ups:*
 - ◆ **Re: negotiation by failure– list operations**
◇ *From:* Brian Hulley

Re: negotiation by failure– list operations

- **References:**

- ◆ **[negotiation by failure– list operations](#)**

- ◆ *From:* ne0

- Prev by Date: **[Re: cross–module asserting/retracting – is this a good thing?](#)**

- Next by Date: **[Re: cross–module asserting/retracting – is this a good thing?](#)**

- Previous by thread: **[Re: negotiation by failure– list operations](#)**

- Next by thread: **[Re: negotiation by failure– list operations](#)**

- Index(es):

- ◆ **[Date](#)**

- ◆ **[Thread](#)**