

## Re: removing the largest item from the list

**Source:** <http://coding.derkeiler.com/Archive/Prolog/comp.lang.prolog/2004-03/0152.html>

---

**From:** anders t ([anthu\\_001\\_at\\_No€SPaM€\\_hotmail.com](mailto:anthu_001_at_No€SPaM€_hotmail.com))

**Date:** 03/15/04

Date: Mon, 15 Mar 2004 17:05:10 +0100

Quoting Matthew Huntbach in comp.lang.prolog:

>So are you suggesting that the "thousands and thousands of students" who are  
>learning Prolog should swamp this newsgroup with trivial questions?

No.

>Or are you suggesting that people who teach Prolog do so in a way different  
>from people who teach other sorts of programming? If so, I don't see any  
>evidence of that. My experience of teaching any sort of programming is that  
>students need to be encouraged to think through it themselves and that  
>sometimes if they are show answers too early they never properly learn how  
>to program.

>

>Right now I am teaching a course in programming (not in Prolog however) and  
>as an experiment I interviewed those stuents on the course who did badly on  
>the mid-term test. I found a common feature of these students was an  
>over-readiness to jump to model answers or to seek help without thinking it  
>through first. When I asked these students to demonstrate their lab work,  
>they showed me model answers they'd copied, and sometimes trivially modified  
>and said "Oh, I looked through the answers and now I understand it". They  
>didn't understand it – these are people who after a term and a half of  
>learning to program (in Java actually) couldn't string together half a  
>dozen lines of syntactically correct code that resembled a correct solution.

Generally, the guys who have responded to my various posts here, present thoughtful arguments and insights for which I thank.

While I, also in general, certainly agree that practical assignments is must in order to really learn, I must still highlight that the problem is that Prolog really is something new to anyone who has never dealt with declarative programming or, for that matter, heavy list programming.

We don't seem to get the time to let things mature. Especially in a combined AI/Prolog course we get 3–6 hours to learn how Prolog (I guess generally, declarative programming) works and then we are supposed to be off with solving (simpler) AI problems.

comp.lang.prolog: Re: removing the largest item from the list

How many students start solving (simpler) AI problems or equivalents in OO programming after three hours of learning Java as a first OO language? There will of course always be geniuses who handle this perfectly, but geniuses alone can't support the Software market with the manpower needed.

Perhaps our brains are indeed destroyed by functional programming, but that's just another incentive to be more gentle with us students, in order to not repel us from Prolog/Declarative programming.

Also, it isn't helping us students that Prolog is diversified between (competing(??) implementations.

Noone has still not really answered my now (in)famous question (a bit rephrased):

Is there a true interest among Prolog theoreticians and practitioners to popularize Prolog and perhaps make it a mainstream language, like Java, C++, C#, or is there a desire to protect a territory? (I'm not necessarily saying that the latter would be something bad, I just want to establish an understanding.)

--

So, son of logic,  
Boy, born of light,  
Is the picture drawn, is the harvest one,  
Step away