CS101C Homework 5

Due: Wednesday, May 6, 11:59PM (firm)

Collaboration: You are allowed and encouraged to work together and collaborate on all aspects of this homework. However, your submission **must** be your own; you must type in your homework without referring to shared material. For example, suppose you work as part of a group to prove a long, complicated theorem; and suppose you sketch the proof on the board. When you enter the proof into MetaPRL for homework submission, you must not refer to the board — you must recover the proof from your own memory.

Setup

Start this homework by updating MetaPRL to revision **8** (e.g. version "`0.8.1 (CS101 rev 8)`"). Upgrade instructions are available at `http://nogin.org/cs101c/mp-update.html`.

Next, in directory `theories/cs101` of your MetaPRL installation, copy

the files	`cs101_lc.ml`	into	`cs101_lc_`*name*`.ml`	respectively
	`cs101_lc.mli`		`cs101_lc_`*name*`.mli`	
	`cs101_lc.prla`		`cs101_lc_`*name*`.prla`	
	`cs101_int.ml`		`cs101_int_`*name*`.ml`	
	`cs101_int.mli`		`cs101_int_`*name*`.mli`	
	`cs101_int.prla`		`cs101_int_`*name*`.prla`	

(where *name* is your login name). In file `cs101_lc_`*name*`.prla`, replace all occurrences of string `Cs101_lc` with `Cs101_lc_`*name* (use your favorite text editor's "replace all" functionality) and, similarly, in file `cs101_int_`*name*`.prla` replace all occurrences of string `Cs101_int` with `Cs101_int_`*name*. Finally, add `cs101_lc_`*name* and `cs101_int_`*name* to the end of the `MPFILES` variable in the `theories/cs101/Makefile`.

For this homework, you should be working based on the existing `Cs101_lc` and `Cs101_int` theories. You are not allowed to add any new `prim` rules or rewrites to the system (except for the `dep_prod_elim` of problem I.2) and you are not allowed to modify the system in any way, other than extending it with your new `hw4` modules.

Note: after you change the `MPFILES` variable in the `Makefile` or add a new `extends` or `open` directives to a MetaPRL file and before you run `make opt`, you might need to run `make depend` to update the cross-module dependencies.

Part I: cs101_lc_*name*

1. Prove the `pair_type` rule.

2. Add a `dep_prod_elim` rule similar to the `pair_elim` one (which should have been called `prod_elim`) with $p : (x : A \times B[x])$ as the hypothesis being eliminated. Change the `prod_elim` from a `prim` rule into `interactive` one and prove it (using the new `dep_prod_elim` rule).

Part II: `cs101_int_`*name*

Prove all the `interactive` rules that do not have a proof yet (`or_i1`, `or_i2`, `or_e`, `and_i`, `and_e`, `not_i` and `not_e`).

Hint: most of the proofs should be relatively straightforward, except possibly for the `or_e` rule. To prove `or_e`, note that the assumptions provided you with "evidence" for $A \Rightarrow C$ and $B \Rightarrow C$ and once you have $x : (A \vee B)$, $a : (A \Rightarrow C)$ and $b : (B \Rightarrow C)$ you can construct an explicit term providing "evidence" for C.

Submission Instructions

Make sure you `export` all the proofs. Send all the six files (each of the `.ml`, `.mli` and `.prla` for each of the `cs101_lc_`*name* and `cs101_int_`*name*) as text attachments in an email to `cs101-admin@metaprl.org`. Please include "`CS101 HW5`" in the message subject line.