CS4215 Programming Language Implementation

Lab task for Week 04
Interpreter for ePL (25 Jan 2017)

1. Download a folder named lab04.zip from IVLE workbin. The folder contains the following files:
   (a) `ePL.ml` which contains type declarations and pretty-printers for ePL expression and eVML instructions. The codes related to eVML virtual machine will be used in Lab05.
   (b) `ePL_parser.ml` which contains a parser for ePL written using Camlp4.
   (c) `ePL_inter.ml` which is supposed to contain a type-checker and an interpreter for ePL.
   (d) `bincomp4.sh` which contains the compilation instruction for building an interpreter for ePL. You can execute it using `sh bincomp4.sh` which will then produce a batch interpreter, called `epli`.
   (e) Some ePL code examples, e.g. `e6.epl`.

2. The interpreter has two components (i) a type checking/inference system, and (ii) an interpreter for ePL expressions. Both components are incomplete and would currently work for only integer expressions. Do ensure that equality `=` operator is made polymorphic to work with both integer and boolean inputs. After compilation, you may test it against an example using for e.g. `./epli e6.epl`.

3. Complete the two components of the ePL interpreter in `ePL_inter.ml`, so that it would work for all valid expressions of the ePL language. Add call debug tracing for `oneStep`. Find more info in readme.txt.

4. **BONUS 10% :** Can you extend ePL interpreter to support an exponentiation operator `x^y` which returns `x` to the power of `y`. This will require significant changes to the parser, abstract syntax tree, type-checker and interpreter.

5. Submit your program (in a zip file) by the deadline of 6th February 2017, (Fri) 6pm through IVLE workbin.