Exercise Set 9: Strategy Writing and Computational Reflection

The PVS file `exercises/strat.pvs` and the Lisp file `exercises/pvs-strategies` support these exercises.

1. Modify the strategy `bflt3` in `exercises/pvs-strategies` so that it can prove sequents of the form

   ```
   {-1}  a <= k
   {-2}  b <= k
   {-3}  c <= k
   {-4}  a ^ 3 + b ^ 3 = c ^ 3
     |-------
   ```

   where `a`, `b`, and `c` are `posnat` and `k` is an arbitrary constant given as a mandatory first parameter to the strategy.

   Hint: The parameter `k` should appear before the keyword `&optional`, e.g.,

   ```
   (defstep bflt3 (k &optional ...) ...)
   ```

   Do not forget to use the parameter `k` in the recursive calls.

2. Use the modified strategy `bflt3` to prove the following lemmas in `exercises/strat.pvs`.

   ```
   bounded_FLT3_2 : LEMMA
     a <= 2 AND b <= 2 and c <= 2 IMPLIES a^3+b^3 /= c^3

   bounded_FLT3_3 : LEMMA
     a <= 3 AND b <= 3 and c <= 3 IMPLIES a^3+b^3 /= c^3

   bounded_FLT3_4 : LEMMA
     a <= 4 AND b <= 4 and c <= 4 IMPLIES a^3+b^3 /= c^3

   bounded_FLT3_5 : LEMMA
     a <= 5 AND b <= 5 and c <= 5 IMPLIES a^3+b^3 /= c^3
   ```

   Hint: Start the proofs with `(skeep)` and then apply `bflt3` with an appropriate parameter.[1] What did you notice when proving `bounded_FLT3_5`? Do you think that `bflt3` is appropriate to prove the following lemma?

   ```
   bounded_FLT3_10 : LEMMA
     a <= 10 AND b <= 10 and c <= 10 IMPLIES a^3+b^3 /= c^3
   ```

[1] If a proof takes too long you can kill it by typing `Control-C` twice and then `(restore)`.

3. Assume the following definition and theorem

```
bfltn(k,n:nat) : bool =
  FORALL (a,b,c:subrange(1,k)) : a^n+b^n /= c^n

bfltn_sound : THEOREM
  FORALL (a,b,c:posnat,k,n:nat) :
    a <= k AND b <= k AND c <= k AND a^n+b^n = c^n IMPLIES
      NOT bfltn(k,n)
```

Define the strategy `bfltn`, with no parameters, that uses computational reflection to discharge the following lemmas.

```
bounded_FLT3_10 : LEMMA
  a <= 10 AND b <= 10 and c <= 10 IMPLIES a^3+b^3 /= c^3

bounded_FLT4_100 : LEMMA
  a <= 100 AND b <= 100 and c <= 100 IMPLIES a^4+b^4 /= c^4
```

Hint: Before writing the strategy, try to prove lemma `bounded_FLT3_10` by hand using the theorem `bfltn_sound`. Note that the function `bfltn` is defined using a bounded universal quantifier on natural numbers. Hence, it can be ground evaluated with `eval-formula`. The proof involves the proof rules `skeep`, `inst?`, `lemma`, `assert`, and `eval-formula`. The body of the strategy has the form `(then ...)`.

4. Write a strategy `quadratic`, with no parameters, to automatically prove lemmas of the form:

```
quadratic_a_b_c: LEMMA
  EXISTS (x:real): a*x^2 + b*x + c = 0
```

where a, b, and c are fixed real numbers and $a > 0$. Upon completion, you should be able to prove the following two lemmas by entering only one command into the PVS prover

```
quadratic_3_4_1: LEMMA
  EXISTS (x:real): 3*x^2 + 4*x + 1 = 0

quadratic_2p5_n1_n0p7: LEMMA
  EXISTS (x:real): 2.5*x^2 + (-1)*x + (-0.7) = 0
```

Assume the following axiom. You will have to call it in the strategy.

```
quadratic_solvable: AXIOM
  FORALL (a:posreal,b,c:real):
  (EXISTS (x:real): a*x^2 + b*x + c = 0)
  IFF
  b^2 - 4*a*c >=0
```

Finally, use the strategy to prove that

```
quadratic_bignumbers: LEMMA
  EXISTS (x:real): 23451234134*x^2 + 2^700*x+3434532453245^30=0
```

Hint: As in the previous example, before writing the strategy try to prove one of the lemmas by hand using the axiom `quadratic_solvable`. The proof involves the proof rules `lemma`, `inst?`, `replace`, and `eval-formula`. The body of the strategy has the form `(then ...)`.